MCMC-Based Bayesian Change Point Detection: a Comparison between the Gibbs and Metropolis-within-Gibbs Samplers

Naitong Chen

March 16, 2021

1 Introduction

Change point detection solves the class of problems where one would like to identify times when abrupt changes in the underlying data generating process of a time series occurs. In practice, this setting is widely seen in areas such as stock analysis and climate studies [CG97; ENJ04]. In the literature, a wide variety of approaches, both frequentist and Bayesian, have been developed to efficiently solve the change point problems [KFE12; CGS92]. In the Bayesian approaches, the problem is often framed as estimating the posterior distribution of the change point locations. The common advantages of the Bayesian approach to statistical inference apply in the change point setting. Compared to the point estimates of the change point locations, the posterior distribution of the change point locations of a given sequence provides a more flexible and holistic understanding of the time series. For example, having access to the approximate posterior distribution of the change point locations allows us to estimate the probability that a change has occured at some given time.

One of the earliest Bayesian approaches to the change point problem is developed in [CGS92] through Markov Chain Monte Carlo (MCMC). Particularly, a Gibbs sampler is used to obtain samples for estimating the posterior distribution of the change point locations. It is worth noting that, although only the detection of a single change point is discussed in [CGS92], there is a natural extension to the multiple change point setting when the number of change points is known. Specifically, instead of sampling a single change point from the distribution of change points conditioned on all other parameters in the model, a set of change points can be sampled from the conditional distribution over all possible configurations of the change point locations. It is then obvious that the number of possible configurations of the change point locations grows linearly in the length of the time series and exponentially in the number of change points to be detected. As a result, the computational complexity of the Gibbs sampler becomes exponentially more expensive.

The scalability issue described above is present in virtually all change point detection methods. In fact, many of the more recent developments in MCMC-based approaches can be viewed as trying to bypass this computational bottleneck either through clever reformulation of the problem [Ste94; LL01] or the use of Metropolis-Hastings (MH) and potentially well-designed proposal distributions [Gre95; AL08]. The approach in [AL08] is one of the easiest and most intuitive attempts at circumventing the computational challenge posed in [CGS92]. Instead of computing the full conditional distribution of the change point locations, a uniform proposal over all possible configurations of change point locations is accepted/rejected based on the MH ratio. However, while the expensive full conditional distribution no longer needs to be evaluated, exploring this potentially enormous space of change point locations using a uniform proposal may not be the most efficient, thus posing a different challenge.

In this report, we compare the Bayesian change point detection problems proposed in [CGS92] and [AL08]. Using a piecewise constant model, we would like to explore, under different settings, whether there is a trade-off between evaluating the expensive full conditional distribution and searching over a potentially large space using a cheaper sampler that risks low acceptance rates. Through a number of experiments, we find that although the approach in [CGS92] is more favourable in all models that are tested, the approach in [AL08] offers very competitive performance. However, we note that only a uniform proposal distribution is used.

By designing more informative proposal distribution, as already been explored in [BF+18], proposal-based MCMC methods applied to the change point problems would be a very attractive choice when the number of possible change point configurations is too large to be handled by a Gibbs sampler.

2 Problem Setup

Below, we start by specifying the piecewise constant model and outlining the procedures of the two MCMC methods before presenting the discussing the experimental results.

2.1 Pairwise Constant Model

Consider a sequence of random variables $(Y_1, \cdots, Y_n)$, where $\forall i \in \{1, \cdots, n\}, Y_i \in \mathbb{R}$, such that

$$Y_i \sim \begin{cases} N(\mu_1, 1) & 1 \leq i \leq r_1 \\ N(\mu_2, 1) & r_1 < i \leq r_2 \\ \quad \vdots & \\ N(\mu_{k+1}, 1) & r_k < i \leq n \end{cases},$$

where k is known, $\boldsymbol{r} = (r_1, \cdots, r_k)$ denotes the set of integer valued change point locations, and $\boldsymbol{\mu} = (\mu_1, \cdots, \mu_{k+1})$ denotes the underlying means of each segment. Note that the jth segment is defined to be $\boldsymbol{Y}_{r_{j-1}+1:r_j} = (Y_{r_{j-1}+1}, \cdots, Y_{r_j})$, with $r_0 = 0, r_{k+1} = n$ by convention. In addition, we assume that $\boldsymbol{r}$ and $\boldsymbol{\mu}$ are independent and that the random variables are i.i.d. within each segment.

Given a sequence of realized observations $\boldsymbol{y} = (y_1, \cdots, y_n)$ from the above data generating process, its likelihood is defined by

$$L(\boldsymbol{r}, \boldsymbol{\mu} \mid \boldsymbol{y}) = \prod_{i=1}^{k+1} \prod_{j=r_{i-1}+1}^{r_i} f(\mu_i; y_j).$$

Note $f(\mu_i; y_j)$ is the likelihood of μ_i being the mean of a normal distribution with variance 1 given the observation y_j.

Then once a set of prior distributions are defined on the parameters $\boldsymbol{r}$ and $\boldsymbol{\mu}$, the goal of the Bayesian change point detection methods is to estimate the posterior distributions $p(\boldsymbol{r} \mid \boldsymbol{y})$ and $p(\boldsymbol{\mu} \mid \boldsymbol{y})$. We focus on the posterior distribution of the change point locations $p(\boldsymbol{r} \mid \boldsymbol{y})$.

For simplicity, we assume the priors on the segment means to be

$$\mu_i \sim N(m, 1), \forall i = 1, \cdots, k+1, \text{and } m \in \mathbb{R}. \tag{1}$$

As a prior on the change point locations, we assume that $\boldsymbol{r}$ follows a discrete uniform distribution over all possible configurations of the change point locations. Specifically, for any gien possible configuration $\boldsymbol{r}_c$,

$$p_{\boldsymbol{r}}(\boldsymbol{r}_c) = \frac{1}{\binom{n-1}{k}}.$$

Note that by our specification, it is not possible for the last observation in the sequence to be a change point.

2.2 MCMC Simulation Procedures

Under the above specification, to obtain a set of samples for $\boldsymbol{r}$ and $\boldsymbol{\mu}$, given $\left(\boldsymbol{r}^i, \boldsymbol{\mu}^i\right)$, the Gibbs sampler described in [CGS92] generates the next sample $\left(\boldsymbol{r}^{i+1}, \boldsymbol{\mu}^{i+1}\right)$ through the following steps:

1. generate $\boldsymbol{r}^{i+1}$ from the probability mass function

$$p\left(\boldsymbol{r} \mid \boldsymbol{y}, \boldsymbol{\mu}^{\boldsymbol{i}}\right) = \frac{L\left(\boldsymbol{r}, \boldsymbol{\mu}^i \mid \boldsymbol{y}\right)}{\sum_{j=1}^{\binom{n-1}{k}} L\left(\boldsymbol{r}^j, \boldsymbol{\mu}^i \mid \boldsymbol{y}\right)},$$

   where $\boldsymbol{r}^j$ in the denominator denotes the jth configuration from the permutation of all possible configurations;

2. generate $\boldsymbol{\mu}^{i+1}$ by

$$\mu_j^{i+1} \sim N\left(\frac{m + \sum_{l=r_{j-1}+1}^{r_j^{i+1}} y_l}{r_j^{i+1} - r_{j-1}^{i+1}}, \left(1 + r_j - r_{j-1}\right)^{-1}\right), \forall j \in \{1, \cdots, k+1\}. \tag{2}$$

Similarly, given $\left(\boldsymbol{r}^i, \boldsymbol{\mu}^i\right)$, the Metropolis-within-Gibbs (MWG) sampler described in [AL08] generates the next sample $\left(\boldsymbol{r}^{i+1}, \boldsymbol{\mu}^{i+1}\right)$ through the following steps:

1. generate $\boldsymbol{r'}$ from the discrete uniform prior $p_{\boldsymbol{r}}$;

2. accept $\boldsymbol{r'}$ as $\boldsymbol{r}^{i+1}$ with probability $\min\left(1, \beta\left(\boldsymbol{\mu}^i, \boldsymbol{r}^i, \boldsymbol{r'}\right)\right)$, where

$$\beta\left(\boldsymbol{\mu}^i, \boldsymbol{r}^i, \boldsymbol{r'}\right) = \frac{L\left(\boldsymbol{r'}, \boldsymbol{\mu}^i \mid \boldsymbol{y}\right)}{L\left(\boldsymbol{r}^i, \boldsymbol{\mu}^i \mid \boldsymbol{y}\right)};$$

3. otherwise set $\boldsymbol{r}^{i+1} = \boldsymbol{r}^i$;

4. generate μ^{i+1} following the last step of the Gibbs sampler.

Despite the use of conjugate priors when sampling the means of each segment, the trade-off between evaluating the expensive full conditional distribution of the change point locations and searching over a potentially large space using a cheaper sampler that risks low acceptance rates is still present. Specifically, the cost of computing $p\left(\boldsymbol{r} \mid \boldsymbol{y}, \boldsymbol{\mu}^i\right)$ from the Gibbs sampler grows linearly in the length of the time series and exponentially in the number of change points to be detected. The search space size of all configurations of change point locations from the MWG sampler also grows in the same manner.

Before presenting the experiment results, it is worth noting that the posterior distribution

$$\begin{aligned}
p(\boldsymbol{r} \mid \boldsymbol{y}) &= \int_{\mu_1} \cdots \int_{\mu_{k+1}} p(\boldsymbol{y} \mid \boldsymbol{\mu}, \boldsymbol{r}) p_{\boldsymbol{r}}(\boldsymbol{r}) p(\mu_1) \cdots p(\mu_{k+1}) d\mu_1 \cdots d\mu_{k+1} \\
&= p_{\boldsymbol{r}}(\boldsymbol{r}) \left(\int_{\mu_1} p(\boldsymbol{y}_{1:r_1} \mid \mu_1) p(\mu_1) d\mu_1\right) \cdots \left(\int_{\mu_{k+1}} p(\boldsymbol{y}_{r_k+1:n} \mid \mu_{k+1}) p(\mu_{k+1}) d\mu_{k+1}\right) \\
&:= p_{\boldsymbol{r}}(\boldsymbol{r}) Q(1, r_1, \mu_1) \cdots Q(r_k + 1, n, \mu_{k+1}).
\end{aligned}$$

can be fully evaluated. Note that $p(\mu_i)$ denotes the prior distribution of the ith segment mean, and

$$p(\boldsymbol{y}_{r_{i-1}+1:r_i} \mid \mu_i) = \prod_{j=r_{i-1}+1}^{r_i} f(\mu_i; y_j)$$

denotes the likelihood of μ_i given all observations from the ith segment of the sequence.

This posterior distribution can be evaluated using Baye's rule because the $Q(\cdot)$'s are the normalizing constants of the known posterior distributions of the segment means defined in Eq. (2). This allows us to more directly compare the quality of the samples generated from the two MCMC methods in the next section.

3 Experiment

As mentioned in the previous sections, both the length of the entire sequence and the number of change points to be detected have an impact on the number of possible configurations of change point locations and thus the computational cost of evaluating the conditional distribution of the change point locations. Therefore, to evaluate the performance of both MCMC-based Bayesian change point detection methods, inference on five sequences of varying lengths and number of true change points are conducted using both the Gibbs and MWG approaches. The number of possible change point configurations range from 49 to 32509. The detailed specification and visualization of the five sequences are shown in Appendix A.

In this set of experiments, the m value in Eq. (1) is set to be the average of the true segment means of each sequence specified in Appendix A. Then inference on the change point locations of each sequence is run five times using both methods to ensure the results that we obtain are not by chance. For each of these runs, the same initial values are used for both the Gibbs sampler and the MWG sampler, although these initial values are not stored in the final output. Specifically, the initial change point locations are uniformly sampled and the initial segment means are the empirical means of the segments based on the initial change points. The Markov chains are run for one minute on the first and smallest time series, and for three minutes on all of the other four time series. Note that the relative time is measured using `time.perf_counter` provided in Python with the time spent on storing each output sample excluded. It is worth noting that the `numba` package is used for both methods to speed up computations by translating and pre-compiling some of the `numpy` functions. The experiments are run with an Intel i5 7200U processor and 8GB of memory. Code is available at `https://github.com/NaitongChen/STAT520A_Project`.

3.1 Discussion

Firstly, by checking the plots shown in Appendix B, we confirm that the number of samples produced by MWG is indeed orders of magnitude larger than that of Gibbs except for the first sequence. However, this is justified by that evaluating the conditional distribution of the change point locations is relatively cheap as there are only 49 possible locations of the change point.

To further study the actual values of the samples produced, we take a look at the trace plots of each change point location. Appendix C lists the trace plots of each change point, grouped by the particular sequences, across all five runs using both methods. After visually assessing these plots, the samples generated before the Makov chains have converges are removed, and the trace plots after removing the burin-in periods are shown in Appendix D. It is worth noting that this visual assessment is possible because we have knowledge of the ground truth for each sequence. For example, looking at Appendix C without knowing the true change point locations may lead to the conclusion that the corresponding Markov chain has converged since the beginning of the simulation.

From these trace plots, we see that the Markov chains in both methods converge almost immediately under the first two sequences. However, for the latter three sequences, MWG more often than not requires more time than Gibbs before reaching the high density regions. This is particularly evident for the sequence with 60 observations and 3 change points because the shorter segments make the segment means less stable. Even after the corresponding chains have converged, it is clear under the models with a larger search space of change point locations, MWG moves between states much less frenquently than Gibbs does.

To better understand the behaviour of the two approaches, we take a closer look at the last round of simulation on the sequence with 60 observations and 3 change points (Figs. 7e and 12e). Using the same initialization, initially both MWG and Gibbs completely miss the last change point in the sequence. Indeed, putting a change point at location 30 corresponds to a local mode in the posterior distribution of the change point locations, regardless of whether it is the corresponding change point that is given this value. As a result, with a uniform proposal on the change point locations, MWG takes a very long time to generate a sample of higher mass from the more than 30000 possible configurations. Even after the Markov chain of MWG has converged to the region around the global mode, a similar problem is present. Since any set of change points around the global mode has a relatively high mass already, again with the uniform proposal, MWG fails at efficiently exploring this space as the probability of proposing another set of change point locations of comparable mass is low. This is clearly reflected in Fig. 12e.

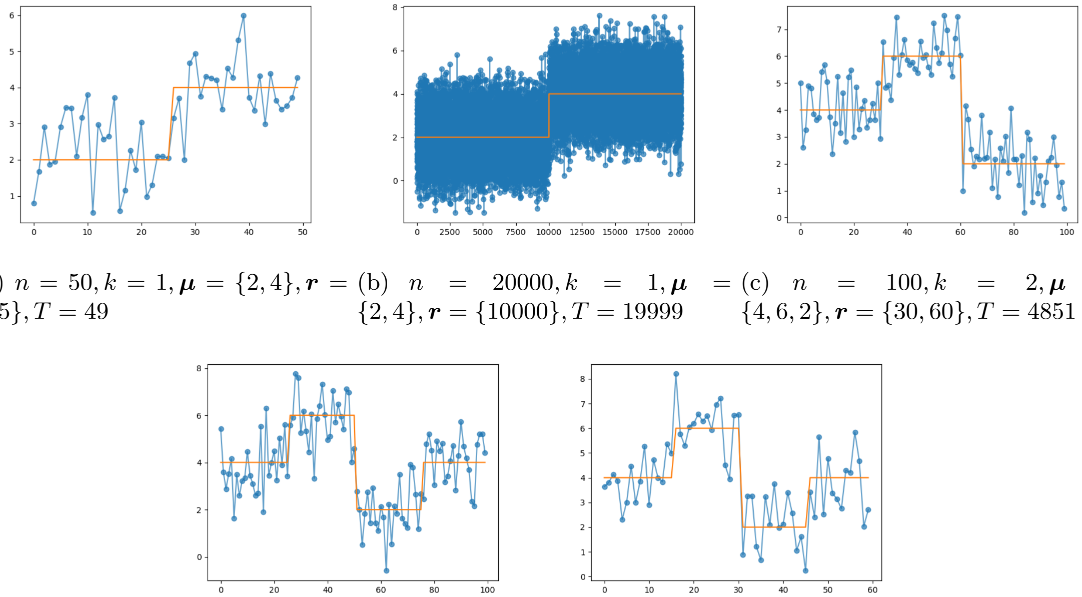
In comparison, the Gibbs sampler gets out of the local mode much faster than MWG does. And even after the Gibbs sampler has reached the region of the global mode of the posterior, it seems to explore this high mass region a lot more thoroughly. This is because the Gibbs sampler utilizes the full conditional distribution of the change point locations conditioning on all other parameters (segment means). In this particular case, despite missing the last change point completely, it very quickly identifies the first and third change point using the more information conditional distribution of the change point locations. Once two of the three change points have been identified, the segment means are immediately updated to be much closer to the true segment means, which then informs the Gibbs sampler to ultimately get out of the local mode.

From above, it seems like the more informative conditional distribution of the change point locations puts the Gibbs sampler at an advantage. However, we remind ourselves that the trace plots are compared iteration wise. In other words, even though MWG requires more samples to reach the region around the global mode, samples from MWG are generated much faster than those from Gibbs. We now check whether this faster sampling rate from MWG makes up for the difference in performance mentioned above.

4 Conclusion

A Description and Visualization of Datasets

See below for the visualization of the five sequences whose change points are to be detected using MCMC-based Bayesian change point detection methods. The blue dots are the observed values and the orange lines show the underlying means of each segment. The number of observation (n), number of change points (k), true segment means ($\boldsymbol{\mu}$), true change point locations ($\boldsymbol{r}$), and the number of possible configurations of change points (T) are shown in the plots below. Note that the x-axis indicate the index or order in time, and the y-axis are the values of the observations.



(a) $n = 50, k = 1, \boldsymbol{\mu} = \{2, 4\}, \boldsymbol{r} = \{25\}, T = 49$

(b) $n = 20000, k = 1, \boldsymbol{\mu} = \{2, 4\}, \boldsymbol{r} = \{10000\}, T = 19999$

(c) $n = 100, k = 2, \boldsymbol{\mu} = \{4, 6, 2\}, \boldsymbol{r} = \{30, 60\}, T = 4851$

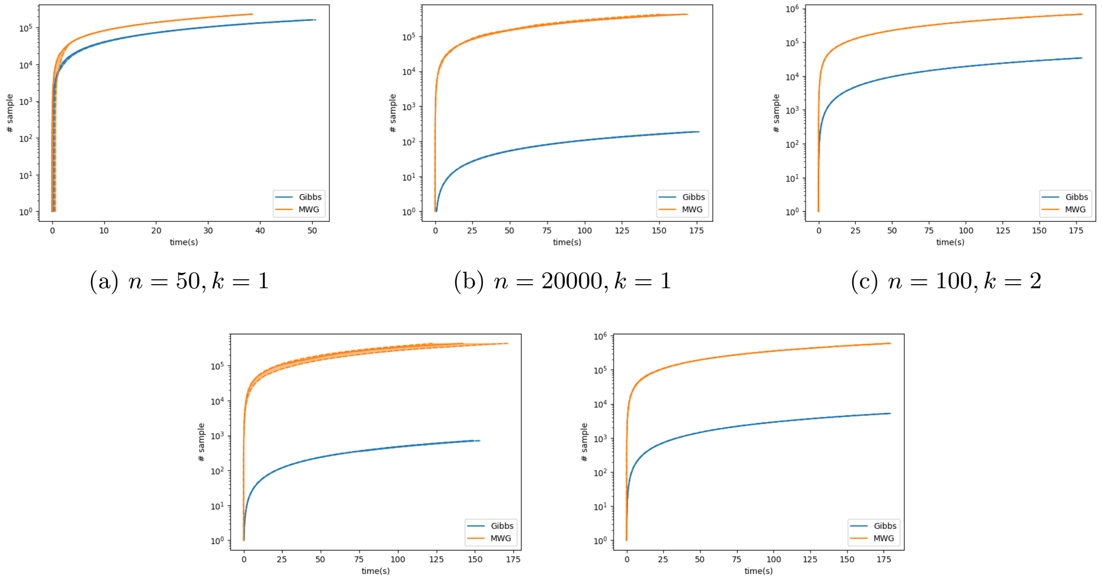
(d) $n = 100, k = 3, \boldsymbol{\mu} = \{4, 6, 2, 4\}, \boldsymbol{r} = \{25, 50, 75\}, T = 156849$

(e) $n = 60, k = 3, \boldsymbol{\mu} = \{4, 6, 2, 4\}, \boldsymbol{r} = \{15, 30, 45\}, T = 32509$

Figure 1: Visualization of sequences with change points to be detected.

B Sample Size Comparison

See below for a comparison of the size of the samples generated by each MCMC sampler for each of the five sequences. For each model, the medians (solid lines), the 25th and the 75th percentiles (dashed lines) of the number of samples generated by each MCMC sampler are plotted across the amount of time elapsed since the start of the simulation. Note that in this particular plot, for each sequence, the samples of each MCMC sampler is trimmed to the minimum sample sizes across all five runs.



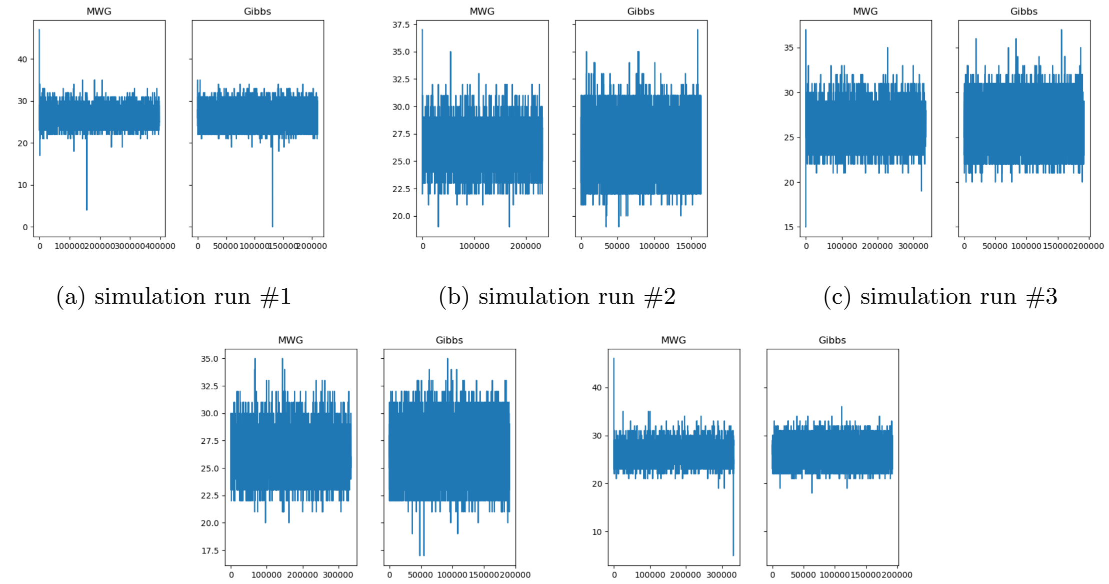
(a) $n = 50, k = 1$ (b) $n = 20000, k = 1$ (c) $n = 100, k = 2$

(d) $n = 100, k = 3$ (e) $n = 60, k = 3$

Figure 2: Number of samples generated by each MCMC sampler for each of the five sequences.

C Trace plots of entire sequences

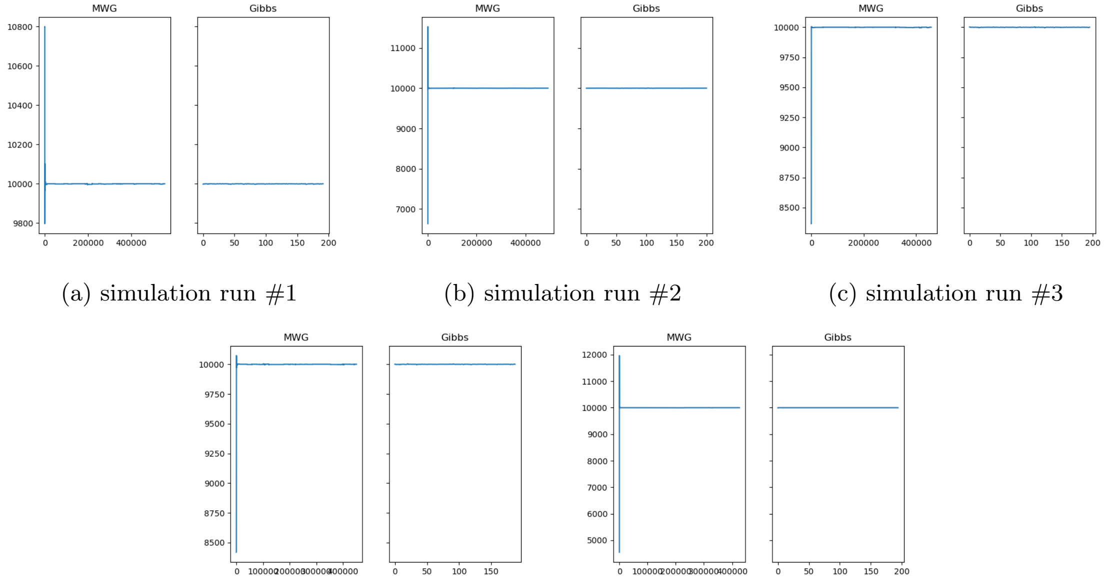
Each of the five figures below show the trace plots of the change points across the five runs for each sequence. Note that the trace plots include all samples generated by both MCMC samplers. Each color represents a distinct change point. For each plot, the x-axis indicate the number of the sample, and the y-axis represent the values of each sample (locations of change points).



(a) simulation run #1 (b) simulation run #2 (c) simulation run #3

(d) simulation run #4 (e) simulation run #5

Figure 3: Trace plots of entire simulations for the sequence with $n = 50, k = 1, \boldsymbol{\mu} = \{2, 4\}, \boldsymbol{r} = \{25\}, T = 49$.



(a) simulation run #1

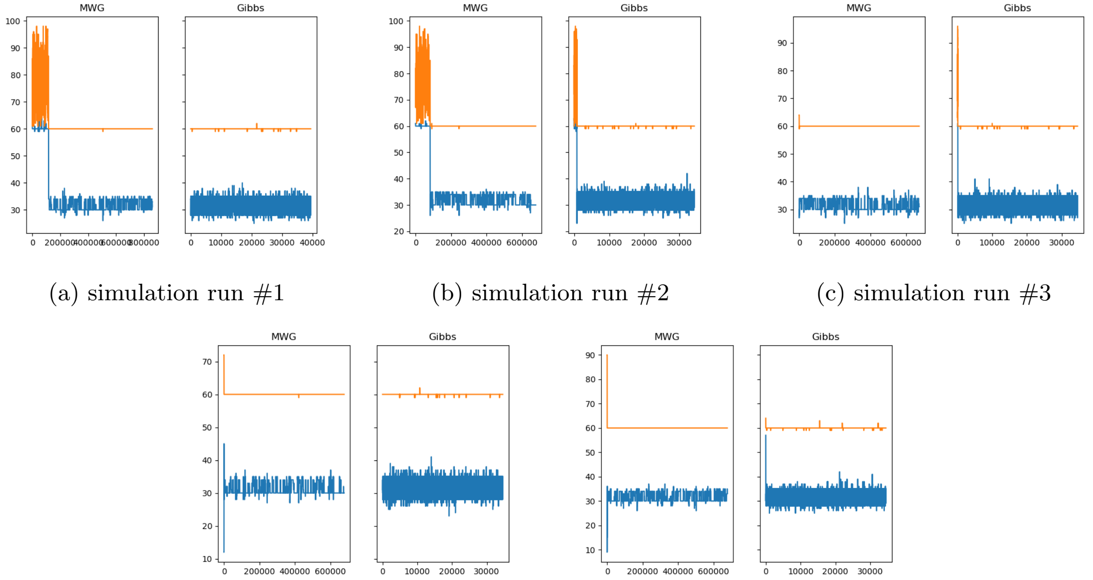
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 4: Trace plots of entire simulations for the sequence with $n = 20000, k = 1, \boldsymbol{\mu} = \{2, 4\}, \boldsymbol{r} = \{10000\}, T = 19999$.



(a) simulation run #1

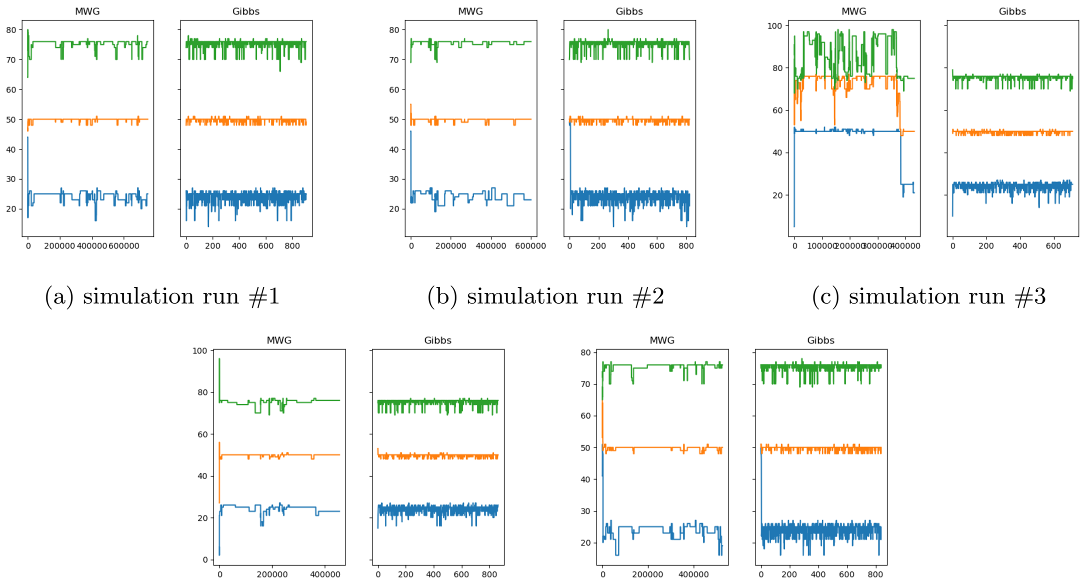
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 5: Trace plots of entire simulations for the sequence with $n = 100, k = 2, \boldsymbol{\mu} = \{4, 6, 2\}, \boldsymbol{r} = \{30, 60\}, T = 4851$.



(a) simulation run #1

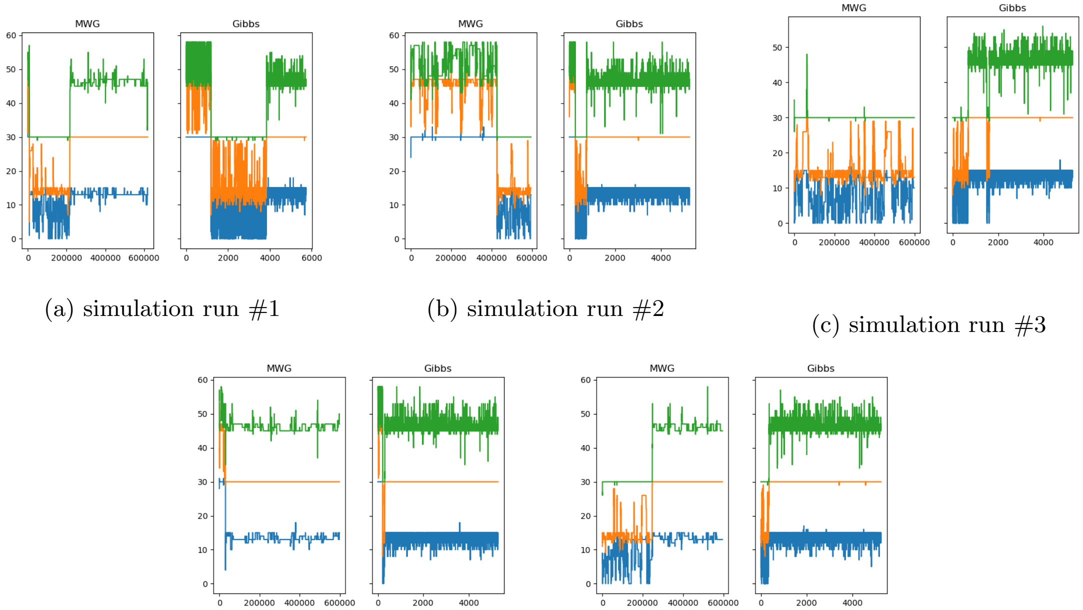
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 6: Trace plots of entire simulations for the sequence with $n = 100, k = 3, \boldsymbol{\mu} = \{4, 6, 2, 4\}, \boldsymbol{r} = \{25, 50, 75\}, T = 156849$.



(a) simulation run #1

(b) simulation run #2

(c) simulation run #3

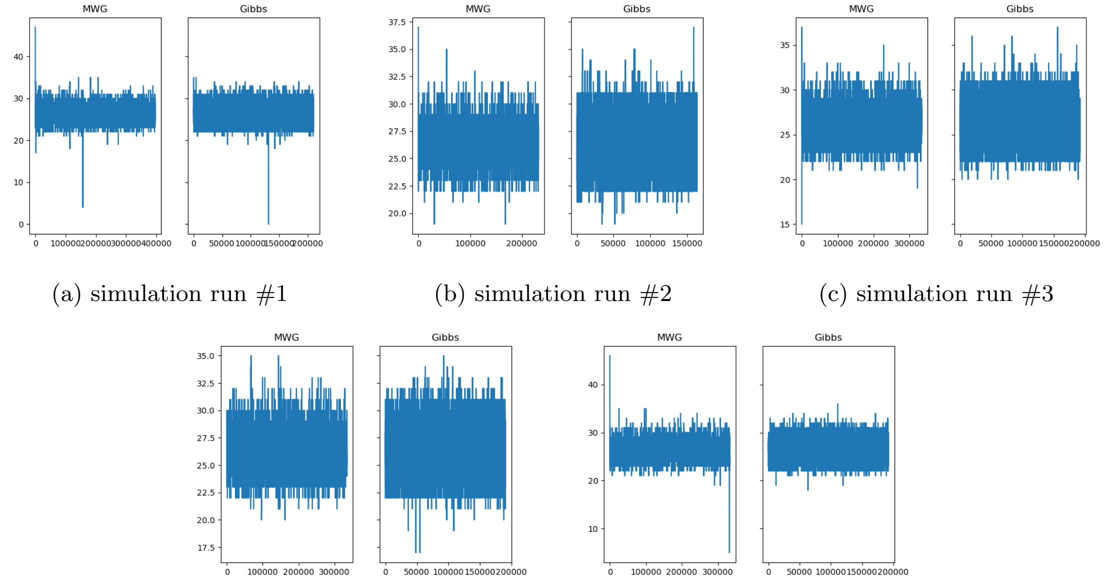
(d) simulation run #4

(e) simulation run #5

Figure 7: Trace plots of entire simulations for the sequence with $n = 60, k = 3, \boldsymbol{\mu} = \{4, 6, 2, 4\}, \boldsymbol{r} = \{15, 30, 45\}, T = 32509$.

D Trace plots after burn in

Each of the five figures below show the trace plots of the change points across the five runs for each sequence. Note that the trace plots exlude samples generated before the corresponding Markov chains have converged. Each color represents a distinct change point. For each plot, the x-axis indicate the number of the sample, and the y-axis represent the values of each sample (locations of change points). Empty plots indicate that the simulation has terminated before the Markov chain converged.



(a) simulation run #1

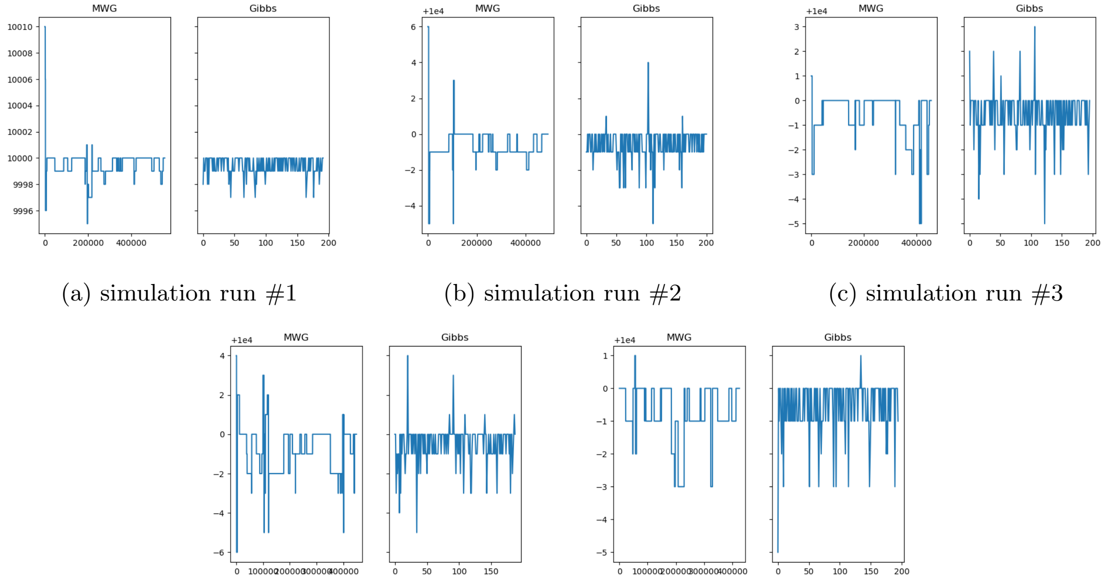
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 8: Trace plots post burn-in for the sequence with $n = 50, k = 1, \boldsymbol{\mu} = \{2, 4\}, \boldsymbol{r} = \{25\}, T = 49$.



(a) simulation run #1

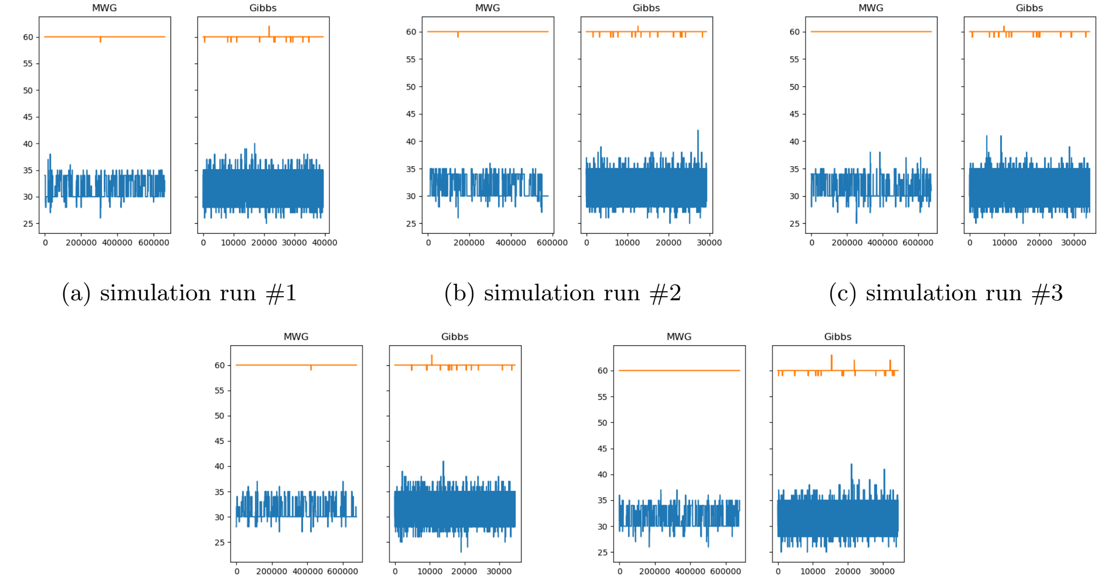
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 9: Trace plots post burn-in for the sequence with $n = 20000, k = 1, \boldsymbol{\mu} = \{2, 4\}, \boldsymbol{r} = \{10000\}, T = 19999$.



(a) simulation run #1

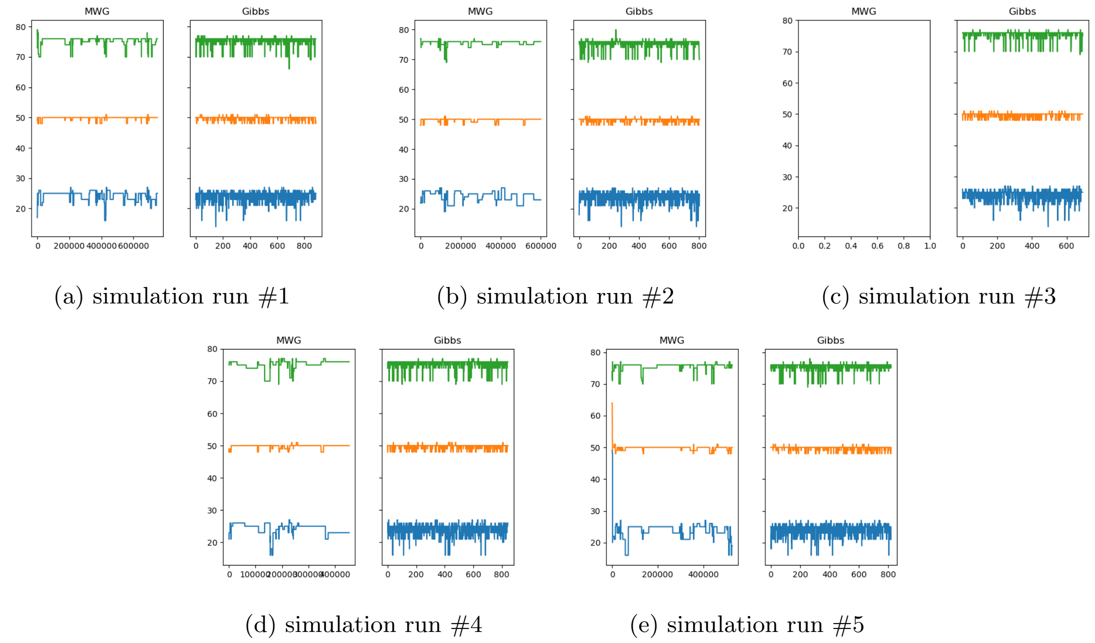
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 10: Trace plots post burn-in for the sequence with $n = 100, k = 2, \boldsymbol{\mu} = \{4, 6, 2\}, \boldsymbol{r} = \{30, 60\}, T = 4851$.



(a) simulation run #1

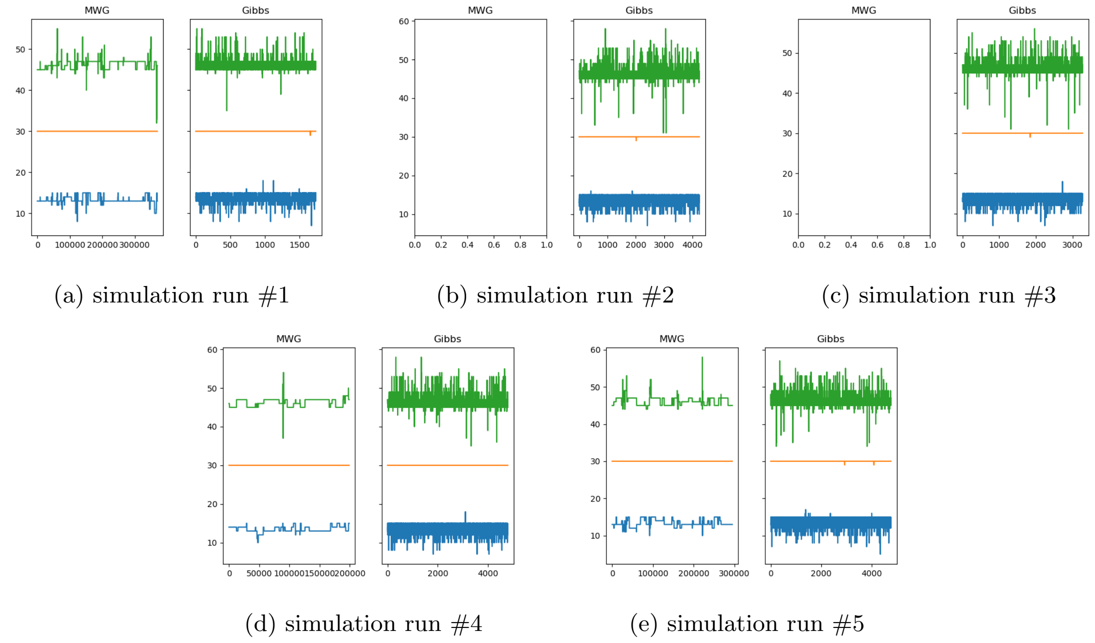
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 11: Trace plots post burn-in for the sequence with $n = 100, k = 3, \boldsymbol{\mu} = \{4, 6, 2, 4\}, \boldsymbol{r} = \{25, 50, 75\}, T = 156849$.



(a) simulation run #1

(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 12: Trace plots post burn-in for the sequence with $n = 60, k = 3, \boldsymbol{\mu} = \{4, 6, 2, 4\}, \boldsymbol{r} = \{15, 30, 45\}, T = 32509$.

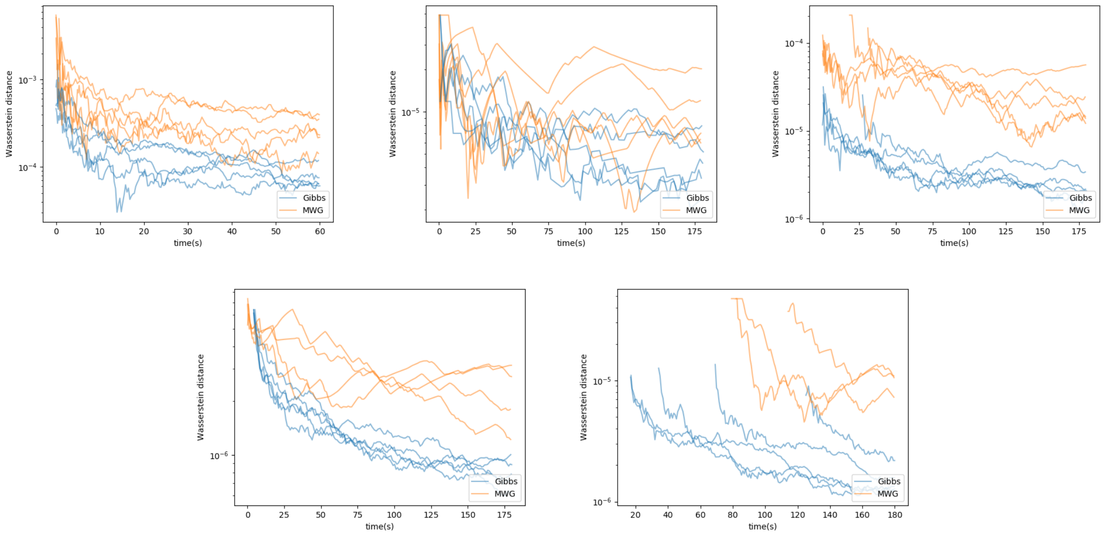
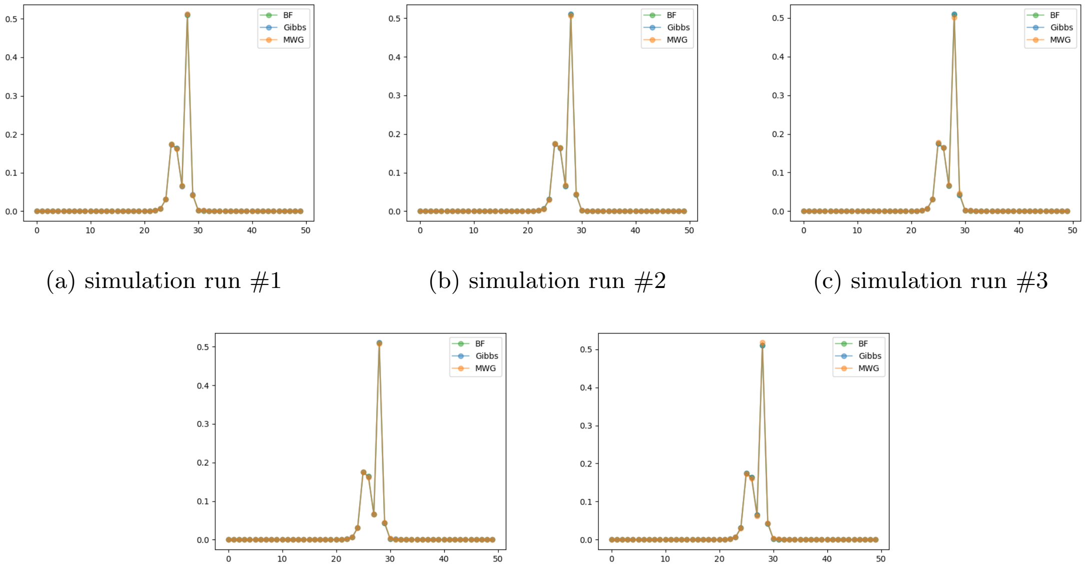


Figure 13: wasserstein

E Wasserstein distance

F Posterior approximation



(a) simulation run #1

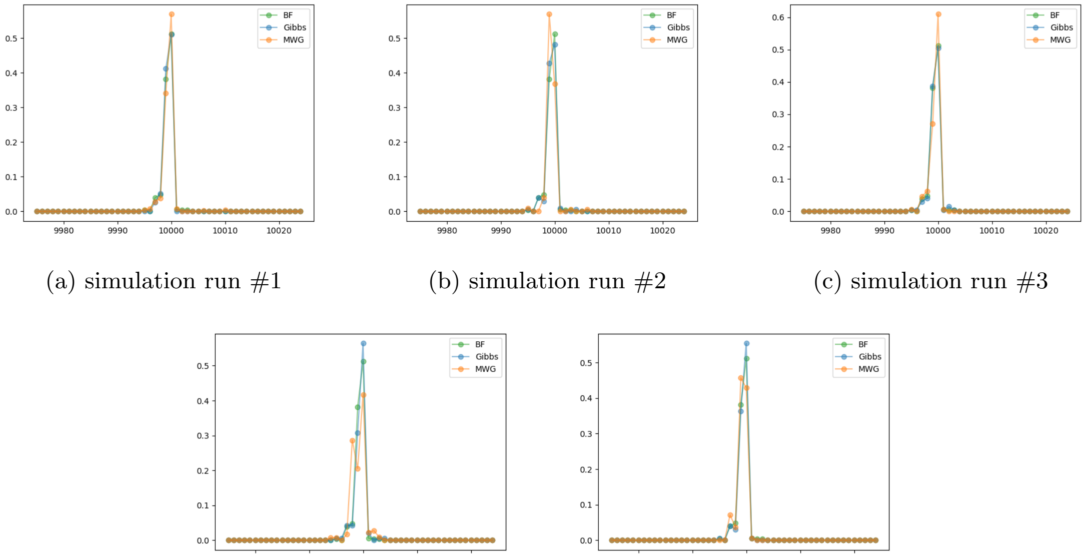
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 14: model1: posterior approximation



(a) simulation run #1

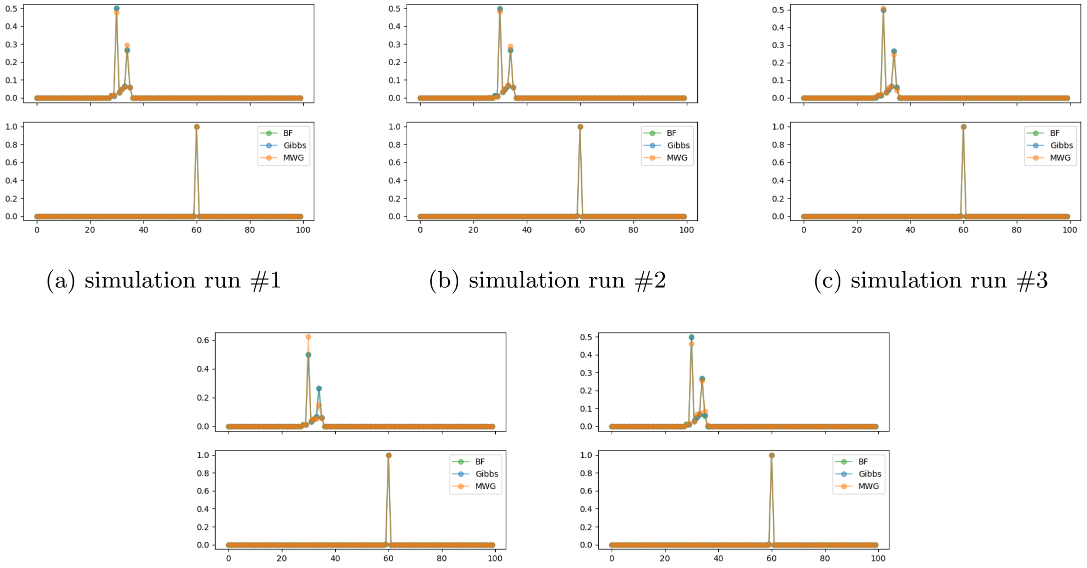
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

Figure 15: model2: posterior approximation



(a) simulation run #1

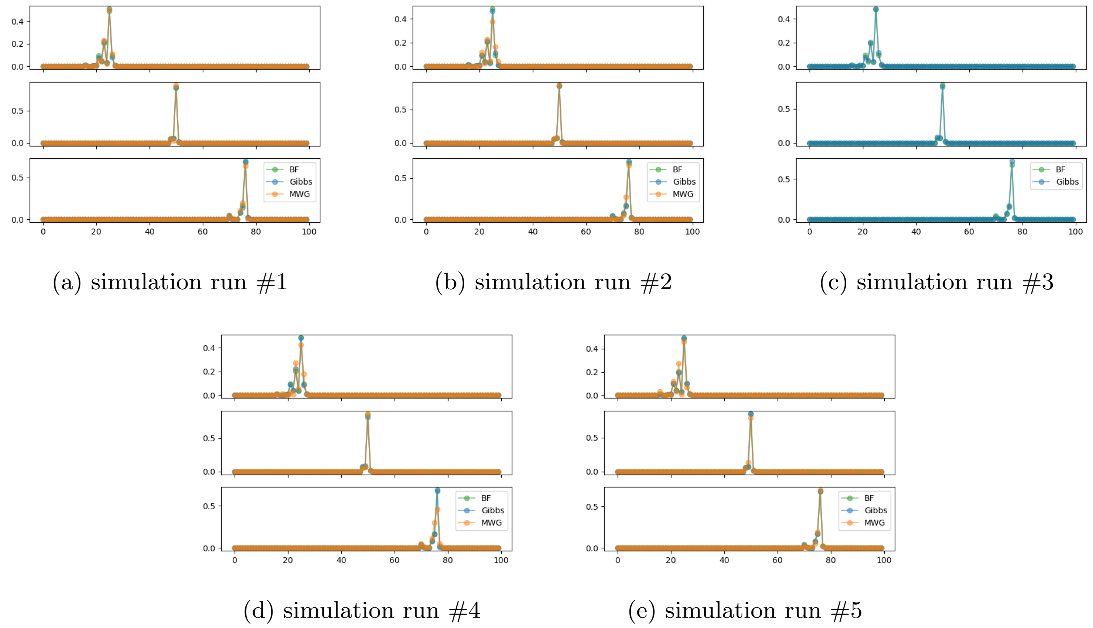
(b) simulation run #2

(c) simulation run #3

(d) simulation run #4

(e) simulation run #5

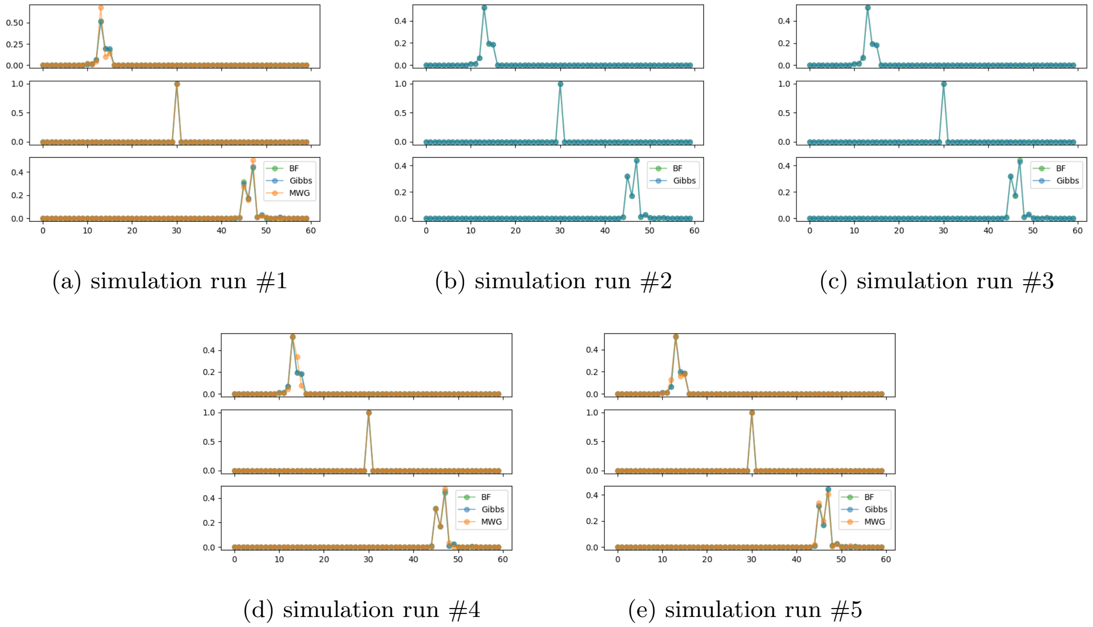
Figure 16: model3: posterior approximation



(a) simulation run #1 (b) simulation run #2 (c) simulation run #3

(d) simulation run #4 (e) simulation run #5

Figure 17: model4: posterior approximation



(a) simulation run #1 (b) simulation run #2 (c) simulation run #3

(d) simulation run #4 (e) simulation run #5

Figure 18: model5: posterior approximation

G ESS and MCSE

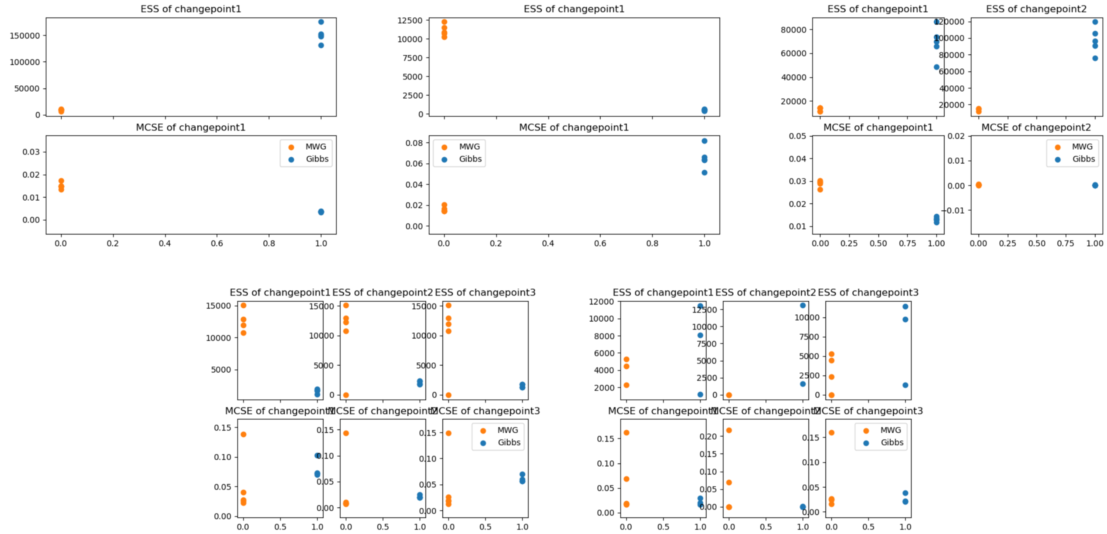


Figure 19: ess and se

References

[AL08] J. Antoch and D. Legát. “Application of MCMC to change point detection”. In: *Applications of Mathematics* 53.4 (2008), pp. 281–296.

[BF+18] A. Benson, N. Friel, et al. “Adaptive MCMC for multiple changepoint analysis with applications to large datasets”. In: *Electronic Journal of Statistics* 12.2 (2018), pp. 3365–3396.

[CGS92] B. P. Carlin, A. E. Gelfand, and A. F. Smith. “Hierarchical Bayesian analysis of changepoint problems”. In: *Journal of the royal statistical society: series C (applied statistics)* 41.2 (1992), pp. 389–405.

[CG97] J. Chen and A. K. Gupta. “Testing and locating variance changepoints with application to stock prices”. In: *Journal of the American Statistical association* 92.438 (1997), pp. 739–747.

[ENJ04] J. B. Elsner, X. Niu, and T. H. Jagger. “Detecting shifts in hurricane rates using a Markov chain Monte Carlo approach”. In: *Journal of climate* 17.13 (2004), pp. 2652–2666.

[Gre95] P. J. Green. “Reversible jump Markov chain Monte Carlo computation and Bayesian model determination”. In: *Biometrika* 82.4 (1995), pp. 711–732.

[KFE12] R. Killick, P. Fearnhead, and I. A. Eckley. “Optimal detection of changepoints with a linear computational cost”. In: *Journal of the American Statistical Association* 107.500 (2012), pp. 1590–1598.

[LL01] M. Lavielle and E Lebarbier. “An application of MCMC methods for the multiple change-points problem”. In: *Signal processing* 81.1 (2001), pp. 39–53.

[Ste94] D. Stephens. “Bayesian retrospective multiple-changepoint identification”. In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 43.1 (1994), pp. 159–178.