

Lecture 40: The Hastings–Metropolis Sampler; Stochastic Sampling of MRFs

Charles A. Bouman

Purdue University

Wednesday, December 2, 2026

Outline

- 1 Why the Metropolis algorithm can be slow to converge
- 2 The Hastings–Metropolis acceptance probability and transitions
- 3 Proof of reversibility: the three cases
- 4 The special case $q(w|x) = p(x)$
- 5 Example: sampling a bimodal density two ways
- 6 Section 15.4: coordinatewise sampling of an MRF
- 7 Nonhomogeneity, subsampling at full sweeps, and convergence

Reading: Chapter 15.3–15.4, *Foundations of Computational Imaging*.

Today's Big Ideas

- The Metropolis algorithm can be slow to converge. Rejected proposals leave the state unchanged, and samples do not become independent as soon as burn-in has ended.
- The Hastings–Metropolis sampler removes the requirement that the proposal distribution be symmetric. It corrects for the asymmetry inside the acceptance probability, and the chain is still reversible.
- A proposal closer to the desired distribution is accepted more often. A proposal equal to the desired distribution is always accepted, and the sampler converges in one step.
- When the state is a Markov random field, a practical proposal changes one pixel at a time, and the pixels are visited in a fixed repeating order.
- The one-pixel-at-a-time chain is not homogeneous. Looking at it once per full sweep of the image gives a homogeneous chain that converges to the desired Gibbs distribution.

The theorem guarantees convergence, but says nothing about speed. The proposal distribution is what makes the sampler practical.

Why slow convergence matters

While the Metropolis algorithm is a very powerful method, it can be slow to converge to the desired distribution. This is a problem for two important reasons.

- First, it means that one might need to generate many samples before the ergodic distribution is reached. The **burn-in time** is the time required for the Markov chain to approximately converge to its ergodic distribution.
- Second, and equally important, many samples must be generated before the random variables, X^k , become approximately independent.

A common misconception

Naive users sometimes assume that once burn-in has occurred, subsequent samples of the Markov chain will be independent. However, this is not true. So if the objective is to produce independent stochastic samples from the Markov chain's ergodic distribution, then a slowly converging Markov chain will require many more samples between approximately independent samples.

The symmetry restriction is severe

A crucial limitation in the Metropolis algorithm is the requirement that the proposal distribution be symmetric. That is,

$$q(x^{(k+1)}|x^{(k)}) = q(x^{(k)}|x^{(k+1)}) .$$

- Intuitively, it is best if the proposal distribution closely approximates the desired ergodic distribution.
- However, this restriction to a symmetric distribution is severe because it is often the case that we might want to “bias” the proposals toward ones that more closely match the desired distribution and are therefore more likely to be accepted.
- Intuitively, proposals that are rejected do not lead to any change in the state, so they are substantially wasted, and one would expect the associated Markov chain’s convergence to be slowed.

Fortunately, the **Hastings–Metropolis sampler** is a generalization of the Metropolis sampler that allows for the use of proposal distributions that are not symmetric. It was discovered by Hastings and Peskun.

The Hastings–Metropolis acceptance probability

Any valid distribution, $q(x^{(k)}|x^{(k-1)})$, can be chosen, but in order to correct for the bias due to this asymmetric proposal density, the probability of acceptance and rejection must be appropriately adjusted. The acceptance probability is given by

$$\begin{aligned}\alpha(w|x^{(k)}) &= \min \left\{ 1, \frac{q(x^{(k)}|w)}{q(w|x^{(k)})} \exp \left\{ -[u(w) - u(x^{(k)})] \right\} \right\} \\ &= \min \left\{ 1, \frac{q(x^{(k)}|w)}{q(w|x^{(k)})} \frac{p(w)}{p(x^{(k)})} \right\} .\end{aligned}$$

So the resulting transition probability is then given by

$$p(x^{(k+1)}|x^{(k)}) = \begin{cases} q(x^{(k+1)}|x^{(k)})\alpha(x^{(k+1)}|x^{(k)}) & \text{if } x^{(k+1)} \neq x^{(k)} , \\ 1 - \sum_{w \neq x^{(k)}} q(w|x^{(k)}) \alpha(w|x^{(k)}) & \text{if } x^{(k+1)} = x^{(k)} \end{cases} .$$

Pseudocode: the Hastings–Metropolis algorithm

Hastings–Metropolis Algorithm:

Initialize $X^{(0)}$

For $k = 0$ to $K - 1$ {

 // Generate proposal

$W \sim q(w|X^{(k)})$ // Use proposal distribution to generate W

 // Accept or reject proposal

$\alpha \leftarrow \min \left\{ 1, \frac{q(X^{(k)}|W)}{q(W|X^{(k)})} \exp \left\{ -[u(W) - u(X^{(k)})] \right\} \right\}$ // Acceptance prob.

$X^{(k+1)} \leftarrow \begin{cases} W & \text{with probability } \alpha \\ X^{(k)} & \text{with probability } 1 - \alpha \end{cases}$

}

The method is basically the same as for the Metropolis algorithm, except that the acceptance probability incorporates the ratio $\frac{q(X^{(k)}|W)}{q(W|X^{(k)})}$.

Proving reversibility: three cases

As with the basic Metropolis algorithm, our approach is to prove that the Markov chain resulting from the Hastings–Metropolis algorithm is reversible. To prove this, we must again show that the Gibbs distribution solves the detailed-balance equation.

Our proof is broken into three cases:

Case 1: $x' \neq x$ and $\frac{q(x|x')}{q(x'|x)} \exp \{-[u(x') - u(x)]\} \geq 1$;

Case 2: $x' \neq x$ and $\frac{q(x|x')}{q(x'|x)} \exp \{-[u(x') - u(x)]\} \leq 1$;

Case 3: $x' = x$.

We again start with Case 1, in which the proposed move is accepted with probability one.

Case 1, line by line

$$\begin{aligned} p(x'|x) &= q(x'|x) \alpha(x'|x) \\ &= q(x'|x) \\ &= q(x|x') \frac{q(x'|x)}{q(x|x')} \\ &= q(x|x') \frac{q(x'|x)}{q(x|x')} \exp\{-[u(x) - u(x')]\} \exp\{-[u(x') - u(x)]\} \\ &= q(x|x') \alpha(x|x') \exp\{-[u(x') - u(x)]\} \\ &= p(x|x') \exp\{-[u(x') - u(x)]\} . \end{aligned}$$

Line 2 uses $\alpha(x'|x) = 1$ in Case 1; line 4 multiplies by two exponentials whose product is one; line 5 recognizes the definition of $\alpha(x|x')$ in Case 1.

Detailed balance, and Cases 2 and 3

From this we again get the detailed-balance equation

$$p(x'|x) \frac{1}{Z} e^{-u(x)} = p(x|x') \frac{1}{Z} e^{-u(x')} .$$

- Once again, Case 2 results directly from the symmetry of this equation.
- Case 3 is immediate by substituting $x' = x$ into it.

So the Gibbs distribution $p(x) = \frac{1}{Z} e^{-u(x)}$ solves the detailed-balance equation for the Hastings–Metropolis chain, for any valid proposal distribution q .

Special case: the proposal is the target distribution

Another special case arises when the proposal distribution is *exactly* the true Gibbs distribution. In this case, we have that $q(w|x) = p(x)$, where $p(x)$ is the desired distribution, and then

$$\begin{aligned}\alpha(w|x) &= \min \left\{ 1, \frac{q(x|w) p(w)}{q(w|x) p(x)} \right\} \\ &= \min \left\{ 1, \frac{p(x) p(w)}{p(w) p(x)} \right\} = 1 .\end{aligned}$$

So in this case, the acceptance probability is always one, and the algorithm converges to the Gibbs distribution in one step.

Example 15.1: a bimodal target, sampled by Metropolis

We use both the Metropolis and the Hastings–Metropolis algorithms to generate samples from the distribution

$$p(x) = \frac{1}{Z} \exp \{-u(x)\} \quad , \quad \text{where} \quad u(x) = 4(x-1)^2(x+1)^2 + \frac{x}{2} \quad .$$

We will generate proposals, W , for the Metropolis algorithm as $W = X^{(k)} + Z$, where Z is additive Gaussian noise with $Z \sim N(0, (1/2)^2)$. From this we know that the proposal distribution has the form

$$q(w|x^{(k)}) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{1}{2\sigma^2} \left(w - x^{(k)} \right)^2 \right\} \quad ,$$

where $\sigma = 1/2$. Therefore, the proposal density has the symmetry property required for the Metropolis algorithm, $q(w|x^{(k)}) = q(x^{(k)}|w)$, and the acceptance probability is

$$\alpha = \min \left\{ 1, \exp \{ -[u(w) - u(x^{(k)})] \} \right\} = \min \left\{ 1, \frac{p(w)}{p(x^{(k)})} \right\} \quad .$$

Example 15.1: a mixture proposal for Hastings–Metropolis

For the Hastings–Metropolis algorithm, we are not limited to symmetric proposal densities, so we may very roughly approximate the desired distribution. We do this rough approximation with a mixture of two Gaussian distributions, one distributed as $N(1, (1/4)^2)$ and a second distributed as $N(-1, (1/4)^2)$:

$$q(w) = \frac{\pi_1}{\sqrt{2\pi\sigma_w^2}} \exp\left\{-\frac{1}{2\sigma_w^2}(w-1)^2\right\} + \frac{\pi_2}{\sqrt{2\pi\sigma_w^2}} \exp\left\{-\frac{1}{2\sigma_w^2}(w+1)^2\right\},$$

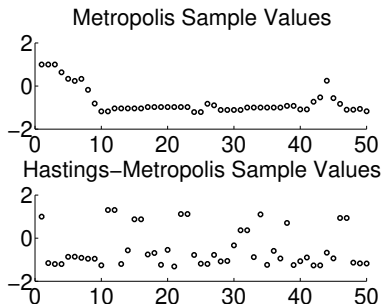
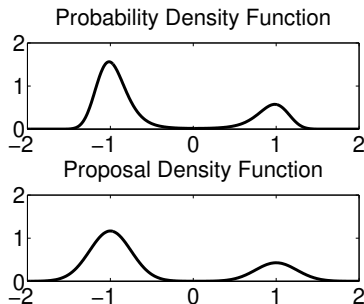
where $\sigma_w = 1/4$, $\pi_1 = \frac{\exp(-1)}{1+\exp(-1)}$, and $\pi_2 = \frac{1}{1+\exp(-1)}$.

This density reasonably approximates the true density, so we would expect that fewer rejections will be needed. The acceptance probability is then

$$\alpha = \min\left\{1, \frac{q(x^{(k)})}{q(w)} \frac{p(w)}{p(x^{(k)})}\right\}.$$

So if $q(x) \approx p(x)$, then the acceptance probability should be ≈ 1 .

Example 15.1: the two simulations



Upper left: the desired density function. Upper right: time sequence of sample values generated by the Metropolis algorithm. Lower left: the proposal density formed by a mixture of two Gaussians. Lower right: time sequence of sample values generated by the Hastings–Metropolis algorithm.

- Adjacent samples generated by the Metropolis algorithm are often exactly the same, because many of the samples are rejected. Intuitively, this high rejection rate slows down convergence of the Markov chain.
- The Hastings–Metropolis samples are much less correlated in time, with many fewer repeated samples. The chain converges more quickly, samples become approximately independent with fewer time steps, and fewer samples need to be generated to compute accurate estimates of time averages.

Section 15.4: when X is an MRF

So far we have been a bit vague about exactly what X represents. In fact, our theory does not assume a specific form for X , but our examples emphasize the case when X is a random variable.

- Stochastic simulation is more widely used in applications where X is a high-dimensional object, such as an MRF. In this case the simulation method produces samples, $X_s^{(k)}$, which vary in both 1-D time, k , and space, s .
- The methods of Sections 15.2 and 15.3 fully apply when X is an MRF, but the particulars of the implementation bear some additional scrutiny. More specifically, one needs to consider what practical choices exist for the proposal distribution, $q(w|x)$.

The coordinatewise proposal

When X is high dimensional, it is common to restrict the proposals to changes in only one coordinate of the high-dimensional vector. So, for example, the proposal distribution is restricted to have the form

$$q_s(w|x) = g_s(w_s|x) \delta(w_r - x_r; r \neq s) ,$$

where $g_s(w_s|x)$ is the conditional proposal distribution of a single pixel, W_s , given the random field X .

- The term $\delta(w_r - x_r; r \neq s)$ in the conditional probability indicates that all other pixels remain the same. So $W_r = X_r$ for $r \neq s$.
- A **coordinatewise sampler** is one that replaces only a single pixel at a time.
- The pixels in the MRF are first ordered from zero to $N - 1$, and this ordering is denoted by $\{s_i\}_{i=0}^{N-1}$. Then in a single full iteration through the image, the pixels are visited and updated in this repeating sequence.

Pseudocode: coordinatewise Hastings–Metropolis

Coordinatewise Hastings–Metropolis Algorithm:

```
Initialize  $X^{(0)}$ 
Select ordering of pixels,  $\{s_i\}_{i=0}^{N-1}$ 
Initialize  $k \leftarrow 0$ 
For  $l = 0$  to  $L - 1$  {                                     // Repeat for  $L$  full iterations through the MRF
    For  $i = 0$  to  $N - 1$  {
        // Generate proposal
         $W \leftarrow X^{(k)}$ 
         $W_{s_i} \sim g_{s_i}(w|X^{(k)})$                                 // Initialize proposal
                                                                    // Proposal for  $s_i^{th}$  pixel
        // Accept or reject proposal
         $\alpha \leftarrow \min \left\{ 1, \frac{g_{s_i}(X_{s_i}^{(k)}|W)}{g_{s_i}(W_{s_i}|X^{(k)})} \exp \left\{ -[u(W) - u(X^{(k)})] \right\} \right\}$ 
         $X^{(k+1)} \leftarrow \begin{cases} W & \text{with probability } \alpha \\ X^{(k)} & \text{with probability } 1 - \alpha \end{cases}$ 
         $k \leftarrow k + 1$ 
    }
}
```

Acceptance probability, irreducibility, and full balance

For this case, the acceptance probability is then given by

$$\begin{aligned}\alpha &= \min \left\{ 1, \frac{q_{s_i}(X^{(k)}|W)}{q_{s_i}(W|X^{(k)})} \exp \left\{ -[u(W) - u(X^{(k)})] \right\} \right\} \\ &= \min \left\{ 1, \frac{g_{s_i}(X_{s_i}^{(k)}|W)}{g_{s_i}(W_{s_i}|X^{(k)})} \exp \left\{ -[u(W) - u(X^{(k)})] \right\} \right\} .\end{aligned}$$

- If the 1-D density function $g_s(w_s|x) > 0$ for all w_s and x , then on each update it is possible (but perhaps not very likely) to produce any pixel update. Since in one pass through the pixels of X every pixel is updated, in a single update any new image is possible. So the Markov chain is both **irreducible** and **aperiodic**.
- Assuming Hastings–Metropolis is used to accept or reject proposals, the chain is also **reversible**. So with the update of any pixel, s , the detailed-balance equations must be satisfied, and therefore so must the full-balance equations:

$$\frac{1}{z} e^{-u(x^{(k+1)})} = \sum_{x^{(k)} \in \Omega^N} p_s(x^{(k+1)}|x^{(k)}) \frac{1}{z} e^{-u(x^{(k)})} .$$

The chain is not homogeneous — and the fix

The analysis of the previous section assumed that the proposal distribution, $q(x^{(k+1)}|x^{(k)})$, is fixed. But here the proposal distribution depends on s , the pixel being updated. Therefore, the resulting Markov chain is *not homogeneous*, and it is unclear whether it converges to the desired Gibbs distribution.

- From the subsampling property of Chapter 13 (Lecture 37), the subsampling of any Markov chain produces another Markov chain.
- Let $X^{(k)}$ be the state after k updates. The subsampled version is $Y^{(k)} = X^{(Nk)}$, where N is the number of pixels. So $Y^{(0)} = X^{(0)}$, $Y^{(1)} = X^{(N)}$ is the state after a single update of each pixel, and $Y^{(k)}$ is the result after k full updates of the state.
- $Y^{(k)}$ forms a **homogeneous** Markov chain because each full update of the pixels follows the same sequence of transition rules. Formally,

$$\tilde{p}(x^{(N)}|x^{(0)}) = \sum_{x^1 \in \Omega^N} \sum_{x^2 \in \Omega^N} \cdots \sum_{x^{N-1} \in \Omega^N} \prod_{i=0}^{N-1} p_{s_i}(x^{(i+1)}|x^{(i)}),$$

which no longer depends on an index of time.

The six ingredients, and the convergence theorem

Since the Gibbs distribution is the stationary distribution of the transitions $p_s(x^{(k+1)}|x^{(k)})$, it must also solve the full-balance equations for $\tilde{p}$. And if $g_s(w_s|x) > 0$, then in one full iteration we can get from any state to any other, so $Y^{(k)}$ is irreducible and aperiodic. So we have all the ingredients required for convergence of the ergodic Markov chain, $Y^{(k)}$, to the Gibbs distribution: 1) The Markov chain is stationary, 2) it has a finite state, 3) it is homogeneous, 4) it is irreducible, 5) it is aperiodic, and 6) the Gibbs distribution solves the full-balance equations of the Markov chain.

Convergence of Coordinatewise Hastings–Metropolis Algorithm

Let $X^{(k)} \in \Omega^N$ be the k^{th} update of a coordinatewise Hastings–Metropolis algorithm sampling from the Gibbs distribution $p(x) = \frac{1}{Z} e^{-u(x)}$, with Ω a finite discrete set. Furthermore, assume that the coordinatewise proposal distribution is strictly positive, that is, for all $w_s \in \Omega$, $g_s(w_s|x) > 0$. Then $X^{(k)}$ is a Markov chain which reaches the stationary distribution

$$\lim_{k \rightarrow \infty} P\{X^{(k)} = x\} = \frac{1}{Z} e^{-u(x)}.$$

Summary

- Metropolis can be slow: burn-in is one cost, and samples do not become independent just because burn-in has occurred. Rejected proposals leave the state unchanged and are substantially wasted.
- Hastings–Metropolis allows any valid proposal q ; the acceptance probability carries the correction $\frac{q(x^{(k)}|w)}{q(w|x^{(k)})}$. The Gibbs distribution still solves detailed balance, so the chain is reversible.
- If $q(w|x) = p(x)$, then $\alpha = 1$ always and the algorithm converges in one step.
- For an MRF, use coordinatewise proposals $q_s(w|x) = g_s(w_s|x)\delta(w_r - x_r; r \neq s)$, visiting pixels in a fixed repeating order.
- That chain is not homogeneous, but the sweep chain $Y^{(k)} = X^{(Nk)}$ is, and it has all six ingredients for convergence. So the coordinatewise Hastings–Metropolis algorithm always converges to $p(x) = \frac{1}{Z}e^{-u(x)}$, although in some cases the convergence is very slow. The basic Metropolis algorithm is a special case, so it converges as well.

Reading: Sections 15.3–15.4. Next lecture: Section 15.5, the Gibbs sampler — another special case that is widely used in practice.