How Heuristics Expedite Markov Chain Search?

–Hitting-time Analysis of the Independence Metropolis Sampler

Romeo Maciuca and Song-Chun Zhu

Abstract

Solving vision problems often entails searching a solution space for optimal state(s) that has maximum Bayesian posterior probability or minimum energy. When the volume of the space is huge, exhaustive search becomes infeasible. Generic stochastic search (e.g. Markov chain Monte Carlo) could be even worse than exhaustive search as it may visit a state repeatedly. To experdite the Markov chain search, one may use heuristics as *proposal probability* to guide the search in promising portions of the space. Empirically the recent data-driven Markov chain Monte Carlo (DDMCMC) scheme[14, 12, 2], achieve fast search in a number of vision tasks, attributed by two observations. (i). The posterior probabilities in vision tasks often have very low entropy and thus narrowly focused on a tight portion of the state space. (ii). The proposal probability computed in bottom-up method can approximate the posterior well. In this paper, we study a independent Metropolis sampler which is simple case used as components in designing complex MCMC algorithm. We obtain analytic formula for the expected times to first hit a certain state ("first hitting-time"), as well as very tight lower and upper bounds which depends on the total variation between the target posterior and the heuristic probabilities. These results show, though in humble cases, that one can indeed reach optimal solutions in very few steps with good proposal probabilities regardless of the size of the original search space. This result is different from previous analysis on the Markov chain convergence rate which are bounded by the second largest eigen-value (modulus) and often correspond to the worst case in the entire search space. In comparison our analysis bears more relevance to the optimization tasks in vision.

Keywords Markov Chain Monte Carlo, First Hitting Time, Convergence Rate, Independence Metropolis Sampler.

The authors are with the Department of Statistics, 8130 Math. Science Bldg, Box 951554, University of California, Los Angeles, Los Angeles, CA 90095.
emails: {rmaciuca,sczhu}@stat.ucla.edu

1 Introduction

Solving vision problems often entails searching a solution space Ω for optimal state(s) $x^* \in \Omega$ that has maximum Bayesian posterior probability $p(x^*)$ (or minimum energy). When the volume of the space is huge, for example, a typical computer vision problem may have $O(10^3)$ variables, then exhaustive search becomes infeasible. Furthermore a typical space Ω contains a large number of subspaces of varying dimensions, this will rule out greedy algorithm and variational methods (PDEs) as they do not realize structural changes and cannot "jump" between subspaces of different dimensions, though these algorithms can be used as components in the general search.

In the recent years there has been growing interest in applying stochastic methods, such as Markov chain Monte Carlo (MCMC) search, to general vision problems in the hope that the Markov chains can find nearly optimal solutions by sampling a tiny portion of the state space. However, generic stochastic search guided by uniform proposal probabilities could be even worse than exhaustive search as it may visit a state repeatedly. One option to experdite the MCMC search is to use heuristics, expressed as *importance proposal probability* $q(x), x \in \Omega$, to guide the Markov chain search. This concept has been used in the recent data-driven Markov chain Monte Carlo (DDMCMC) scheme, and empirically it achieves fast search in a number of vision tasks [14, 12, 2].

In experiments, we observe that designing search algorithms in vision tasks has two advantages in comparison to other MCMC applications in physics and biology. That is, an input image has a large amount of pixels and provides abundant information.

1. An image usually has a unique or very few possible interpretations agreed among human observers. Thus the posterior probability p must be focused on a tight portion of the state space with the mode (x^*) dominating the probability. In other words $p(x^*) \sim O(1)$ is close to one if we discretize the space. The effective state space for each p has a volume equivalent to $\exp\{-\text{entropy}(p)\}$ and is often very small. The rest of the state space is irrelevant to the optimization problem.

2. The vision community has developed abundant bottom-up methods that can compute probability $q()$ as approximations to $p()$. Usually $p()$ is of complex form and we cannot

draw samples from it directly. Instead we can sample $q()$ and use these samples to test p or its ratio. In this way, $q()$'s are used as heuristics to guide the search in a tiny portion of the state space.

Despite the empirical success, unfortunately there is no analysis in the literature: for example, why can heuristics can experdite the Markov chain search? To what extent does the speed-up depend on the divergence between the heuristics q anf the target p?

In this paper, we study a Metropolized Independence Sampler or Independent Metropolis Sampler(IMS), which is a simple case used as components in designing complex MCMC algorithms[12]. Our analysis shows that one can indeed find the optimal states in very few steps if the two observations above hold true: (1) p is peaky and (2). q is close to p. Though our results are derived from a humble case, they are very different from conventional analysis on the convergence rate bounded. As we shall show later that the conventional bounds are based on worst state performance and they are less relevant to the optimization problems in vision.

The rest of this paper is organized as follows. In Section 2 we first overview our main results to provide the intuitive ideas. Then in Section 3, we present the previous results on the convergence rate to set up the background and to compare them with our analysis. In Section 4, we show our analysis and proofs in details. In Section 5 We discuss some remaining problems for future research.

Considering that the materials may not be familiar to general vision audience, we present the results in a tutorial style and provides discussions for clarity.

2 Overview of our results

As the analysis in later sections engages long mathematical deductions, we summarize the main results in this section and illustrate them with a simple example in Figure 1.

Let $\Omega = \{1, 2, ..., N\}$ be a finite state space. $p(x)$ and $q(x), x = 1, 2, ..., N$ are respectively the target and proposal probabilities. Our goal is to search for a state x^* with maximum probability

$$x^* = \arg\max_{x \in \Omega} p(x).$$

The Markov chain starts with a state x_0 sampled from $q(x)$ and it visits a sequence of state over time,

$$x_0, X_1, X_2, ..., X_n, ... \quad x_o \sim q(x).$$

The transition from X_n to X_{n+1} is decided by conditional probability – the transition kernel which is an $N \times N$ matrix,

$$\mathbf{K}(x,y) = \begin{cases} q(y)\alpha(x,y) & y \neq x, \\ 1 - \sum_{y \neq x} \mathbf{K}(x,y) & y = x. \end{cases} \tag{1}$$

At each step, it proposes a new state y according to $y \sim q(y)$ and accepted it with probability

$$\alpha(x,y) = \min(1, \frac{q(x)p(y)}{q(y)p(x)}).$$

It stays at x if the proposal is rejected. This is the Metropolis-Hastings method with the proposal independent of the current state x. It is called the Independence Metropolis Sampler (IMS).

As it shall become clear later that a key quantity to the analysis is the probability ratio

$$\omega(x) = \frac{q(x)}{p(x)}.$$

It measures how much knowledge heuristics $q(x)$ has about $p(x)$. Therefore we define the following concepts.

Definition 1 *As state x is said to be* over-informed *if $q(x) > p(x)$ and x is* under-informed *if $q(x) < p(x)$.*

There are three special states defined below.

Definition 2 *A state x is* exactly-informed *if $q(x) = p(x)$. A state x is* most-informed *(or* least-informed*) if it have the highest (or lowest) ratio $\omega(x)$,*

$$x_{\max} = \arg\max_{x \in \Omega}\{ \frac{q(x)}{p(x)} \} \quad x_{\min} = \arg\min_{x \in \Omega}\{ \frac{q(x)}{p(x)} \}$$

For each state x, we measure the “first hitting-time” defined below.

Definition 3 *The* first-hitting-time *for a state x is the number of steps for reaching x at the first time in the Markov chain sequence,*

$$\tau(x) = \min\{n \geq 1 : \ X_n = x\}. \tag{2}$$

$E[\tau(x)]$ is the expected first hitting time for the Markov chain governed by $\mathbf{K}$.

$E[\tau(x)]$ is a good measure for the speed of search in general. As a special case we need to know $E[\tau(x^*)]$ for the optimal state.

Our main conclusion comes as an analytical solution below,

$$E[\tau(x)] = \frac{1}{p(x)(1-\lambda_x)}. \tag{3}$$

In the above equation, $0 \leq \lambda_x \leq \omega_{\min} < 1$ is the eigen-value of the kernel matrix $\mathbf{K}$ corresponding to state x, and $\omega_{\min}$ is the ratio at the least-informed state.

$$\omega_{\min} = \min_{x \in \Omega} \{ \ \frac{q(x)}{p(x)} \ \}. \tag{4}$$

In fact, the eigen-values can be solved analytically[7]. Furthermore, we obtain very tight lower and upper bounds for $E[\tau(x)]$,

$$\frac{1}{\min\{p(x), q(x)\}} \leq E[\tau(x)] \leq \frac{1}{\min\{p(x), q(x)\}} \cdot \frac{1}{1 - ||p-q||_{TV}}, \tag{5}$$

where $||p-q||_{TV}$ is the total variation between q and p

$$||p-q||_{TV} = \frac{1}{2} \sum_{x \in \Omega} |p(x) - q(x)|.$$

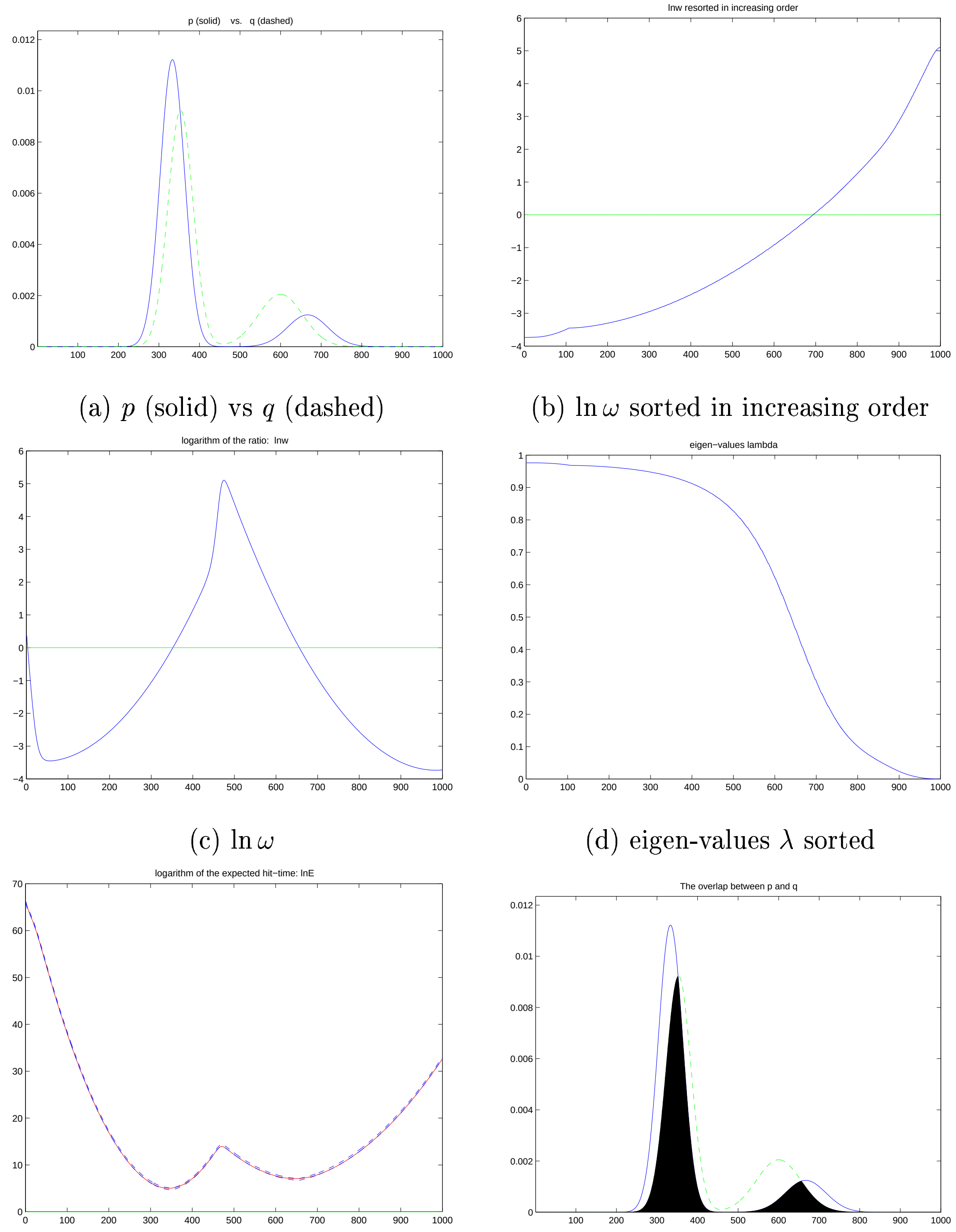
$1 - ||p-q||_{TV}$ is the overlap between p and q.

Therefore, if $p(x^*)$ dominates the probability and q is close to p, then the algorithm can hit x^* in very few steps regardless of the original volume of the state space.

The bound is very tight as Figure 1.e shows and equalities are achieved at the three special states.

1. $E[\tau(x)] = 1/p(x)$ at the most-informed state $x_{\max}$.

2. $E[\tau(x)] = 1/q(x)$ at the least-informed state $x_{\min}$.



(a) p (solid) vs q (dashed)

(b) $\ln \omega$ sorted in increasing order

(c) $\ln \omega$

(d) eigen-values λ sorted

(e) $\ln E[\tau(i)]$ (solid) and bounds (dashed)

(f) p and q overlap (shadow)

Figure 1: An example of illustration.

3. $E[\tau(x)] = 1/p(x) \cdot 1/(1 - ||p - q||_{TV})$ at the exactly-informed state.

Figure 1 illustrates the results in a simple example. We choose a space with $N = 1000$ states. p and q are mixtures of two Gaussians discretized with tails truncated and then normalized to one. They are plotted as solid (p) and dashed (q) curves in Fig.1.a. Fig.1.c plots $\ln \omega(x), x = 1, 2, ..., N$. The states with $ln\omega(x)$ below the horizontal line are under-informed, otherwise they are over-informed. The minimum, zero-crossing, and maximum points in the figure correspond respectively to the least-, exactly-, and most-informed states. Fig.1.e plots the expected first hitting-time $\ln E[\tau(x)]$. The lower and upper bounds in equation (5) are plotted in logarithm as dashed curves which almost coincide with the hitting-time plot. This implies the bounds are in the same order as the expected hitting-time. As we can see that the mode $x^* = 333$ has $p(x^*) \approx 0.012$ and is hit in $E[\tau_{x^*}] \approx 162$ times on average. This is much smaller than $N = 1000$. We define the ratio below to measure how much fold a Markov chain with heuristics improves over exhaustive search.

Definition 4 *The search efficiency of a Markov chain for target p with heuristics q is measured by*

$$\text{efficient ratio}: \ \mathrm{S(p, q)} = \frac{|\Omega|}{E[\tau(x^*)]}$$

To reveal more information, we sort the states in increasing order of $\omega(x)$ and re-plot $\ln \omega$ in Fig. 1.b. The corresponding eigen-values are sorted accordingly and are plotted in Fig. 1.d. It is clear that $x_{\max}$ (right endpoint in Fig. 1.d) corresponds to the smallest eigen-value $\lambda_{x_{\max}} = 0$ thus it is sampled literally according to $1/p(x_{\max})$. In contrast, the least-informed state $x_{\min}$ corresponds to the largest eigen-value 0.96 (left end point). Fig. 1.f plots the overlap $1 - ||p - q||_{TV}$ by the shadow area. The bigger the overlap is, the fast the search will be.

Our result on the first hitting time analysis is different from previous analysis on the Markov chain convergence rate which are bounded by the second largest eigen-value and often correspond to the worst case in the entire search space. To be more concrete, for the independence Metropolis sampler, this largest eigen-value corresponds to the least-informed state, which is the most severe bottleneck in the sampling process. In comparison the first hitting-time analysis is not limited by the worst case and instead it is affected by the informed-ness at each state and the total variation between p and q. So it bears more relevance to the optimization tasks in vision. For comparison, we review the previous results on IMS convergence.

3 Previous results on the convergence rate of IMS

Suppose the Markov chain starts at x_o, after n steps its state X_n follows $\mathbf{K}^n(x_o, x)$ – a probability for visiting x from x_o in n steps. The convergence of the Markov chain is measured by the total variation,

$$||\mathbf{K}^n(x_o, \cdot) - p||_{TV} = \frac{1}{2} \sum_{x \in \Omega} |\mathbf{K}^n(x_o, x) - p(x)|.$$

The objective is to have the Markov chain simulating the target probability p so that X_n is a fair sample from p. Recall that this is a viable way to draw fair samples from p when p is complicated and cannot be sampled in one step.

The independent Metropolis sampler is a special case and clean results have been derived for its convergence rate in both finite and infinite spaces.

3.1 Convergence rate of IMS in continuous space

The conclusions in this section are true for Ω being continuous space.

A Markov chain is denoted by a triplet $< \mu_o, p, \mathbf{K} >$ with probability μ_o for its initial state, p the target probability, and $\mathbf{K}$ the transition kernel. It is desirable to have the total variation decreasing steadily. This is the concept of being "uniformly ergodic".

Definition 5 *A Markov chain* $< \mu_o, p, \mathbf{K} >$ *is said to be* uniformly ergodic*, if there exists a sequence* $r_n \to 0$ *as* $n \to \infty$ *such that*

$$||\mathbf{K}^n(x_o, \cdot) - p||_{TV} \leq r_n, \quad \forall x_o \in \Omega.$$

We re-state a theorem for the convergence rate (Meyn and Tweedie, 1993, chapter 16).

Theorem 1 (Geometric convergence) *A Markov chain is uniformly ergodic, if there exists a constant $\delta > 0$ and some probability $\nu(\cdot)$ such that*

$$\mathbf{K}^m(x_o, y) \geq \delta \nu(y), \quad \forall y, x_o \in \Omega.$$

If the condition holds, the Markov chain converges at geometric rate

$$||\mathbf{K}^n(x_o, \cdot) - p||_{TV} \leq (1-\delta)^{[n/m]}$$

$[n/m]$ is the integer of n/m.

Obviously, as δ is close to zero, then the Markov chain converges very slowly. For the IMS case, one can compute δ explicitly[8],

$$\delta = \min_{x \in \Omega} \{\frac{q(x)}{p(x)}\} = \omega_{\min}.$$

That is the ratio at the least-informed state.

Theorem 2 (Mengersen and Tweedie, 94). *In the above notation, an IMS Markov chain is uniformly ergodic if there exists $\omega_{\min} > 0$, and*

$$||\mathbf{K}^n(x_o, \cdot) - p||_{TV} \leq (1-\omega_{\min})^n.$$

Conversely if $\omega_{\min} = 0$, then the algorithm is not uniformly ergodic, i.e. it does not have a geometric rate of convergence.

The proof is rather simple, we put it below to illustrate the ideas.

[Proof]. For any states $x, y \in \Omega$,

$$\begin{aligned} \mathbf{K}(x,y) &= q(y) \min\{\ 1, \frac{p(y)q(x)}{p(x)q(y)}\ \} \\ &= \min\{\ \frac{q(y)}{p(y)}, \frac{q(x)}{p(x)}\}p(y) \\ &\geq \min_{x \in \Omega}\{\frac{q(x)}{p(x)}\}p(y) = \omega_{\min} p(y). \end{aligned}$$

Then it follows from theorem 1 with $m = 1$, $\nu = p$, and $\delta = \omega_{rmmin}$. To save space, we don't show the converse conclusion for $\omega_{\min} = 0$. □.

This theorem depicts a rather dismay picture for the IMS Markov chain, for its convergence rate is decided by the least-informed state in the entire state space. For example, suppose we have two Gaussians with identical variances and slightly shifted centers $\mu_1 \neq \mu_2$.

$$p(x) = N(\mu_1, \sigma^2), \text{ and } q(x) = N(\mu_2, \sigma^2).$$

Then it is trivial to show that

$$\omega_{\min} = \min_{x \in \Omega}\{\frac{q(x)}{p(x)}\} = 0.$$

The least-informed case occurs at infinity, then the Markov chain has not even a geometric rate of convergence! It becomes uniformly ergodic when q has a bigger variance than p, i.e. the proposal probability is broader. Such worst case analysis also shows up in the discrete finite space.

3.2 Convergence rate of IMS in finite space

We now consider finite space Ω with N states, without loss of generality, we re-order the states and index them in an increasing order of the probability ratio $\omega(x)$,

$$\omega(1) = \frac{q(1)}{p(1)} \leq w(2) = \frac{q(2)}{p(2)} \leq \cdots \leq \omega(N) = \frac{q(N)}{p(N)} \tag{6}$$

The advantage for re-ordering the states is that we can eliminate the min() function in the transition kernel. For $i \neq j$, we have

$$\mathbf{K}(i,j) = q(j)\min\{1, \frac{q(i)p(j)}{q(j)p(i)}\} = q(j)\min\{1, \frac{\omega(i)}{\omega(j)}\}.$$

Therefore we have

$$\mathbf{K}(i,j) = \begin{cases} \omega(i)p(j) & i < j, \\ 1 - \sum_{k<i} q(k) - \omega(i)\sum_{k>i} p(k) & i = j, \\ q(j) = \omega(j)p(j) & i > j. \end{cases}$$

As a result of this neat formulation, its eigen-values can be computed analytically.

Theorem 3 (Jun Liu, 1995-96). *When the states are re-ordered in eqn. 6, the transition matrix* $\mathbf{K}$ *has ordered eigen-values*

$$1 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N \geq 0.$$

Let $\alpha(k) = \sum_{i \geq k} q(i)$ *and* $\beta(k) = \sum_{i \geq k} p(i)$*, then*

$$\lambda_k = \alpha_k - \omega(k) \cdot \beta(k).$$

The eigen-vector for λ_k *is*

$$v_k = (0, \cdots, 0, \beta(k+1), -p(k), \cdots, -p(k))$$

For a special case, $\alpha(1) = \beta(1) = 1$ *and* $\lambda_1 = 1 - \omega(1) = 1 - \frac{q(1)}{p(1)}$.

One can verify the conclusion with feasible efforts! Then, the following two results relate the convergence rate to λ_1 and the ratio of the least-informed state $\omega_{\min} = \omega(1)$.

Theorem 4 (Diaconis and Hanlon, 1992). *For the above IMS Markov kernel* $\mathbf{K}$*, with stationary distribution* p*, for any starting state* i*, after* n *steps,*

$$\|\mathbf{K}^n(i, \cdot) - p\|^2 \leq \frac{\lambda_1^{2n}}{4p(i)}.$$

Theorem 5 (Jun Liu, 1996) *If the initial probability is* q *then*

$$\|\mathbf{K}^n - p\| \leq 2\lambda_1^n = 2(1 - \omega_{\min})^n.$$

The last result is proved by a coupling argument which still holds for a countable state for a countable state space and also for a continuous state space (Liu 1996).

Again, this rate is based on the least-informed case. When $\omega_{\min}$ is close to zero, the Markov chain converges very slowly.

4 The first-hitting-time analysis

Now we return to the first-hitting-time analysis and present a sequence of results in the order that we originally pursued them. The final result was discussed in the introduction section.

To fix notation, we assume the states are re-ordered so that,

$$\omega(1) = \frac{q(1)}{p(1)} \leq \cdots \leq \omega(N) = \frac{q(N)}{p(N)}.$$

The Markov chain state over discrete time steps are

$$x_o, X_1, X_2, ..., X_n, ..., \quad x_o \sim q.$$

We are interested in $\tau(i)$ – the first time that the Markov chain hits (visits) state i, as we define before. Then the average hitting time of i is

$$E[\tau(i)] = \sum_{n=1}^{\infty} nP(\tau(i) = n) = 1 + \sum_{n\geq 2}^{\infty} P(\tau(i) \geq n). \tag{7}$$

4.1 The first lower bound: $E[\tau_i] \geq 1/q(i)$

We start with an ideal case that we have perfect knowledge about p from heuristics, i.e. $q(i) = p(i), i = 1, 2, ..., N$. In other words, we can sample from p directly with acceptance

$$\alpha(i,j) = \min\{1, \frac{q(i)p(j)}{q(j)p(i)}\} = 1, \forall i, j = 1, 2, ..., N.$$

Then we have a quick proposition below.

Proposition 1 *For an ideal IMS with $q(i) = p(i), i = 1, 2, ..., N$, then*

$$E[\tau(i)] = \frac{1}{p(i)} = \frac{1}{q(i)}, \quad \forall i.$$

[Proof] When sample from p independently, the probability of hitting state i is,

$$P(\tau(i) = n) = P(X_1 \neq i, X_2 \neq i, \ldots, X_{n-1} \neq i, X_n = i) = (1 - p(i))^{n-1} p(i).$$

Then

$$E[\tau(i)] = \sum_{n=1}^{\infty} n(1 - p(i))^{n-1} p(i). \tag{8}$$

By a standard treatment, we multiply both sides by $(1 - p(i))$, thus

$$(1 - p(i))E[\tau(i)] = \sum_{n=2}^{\infty} (n-1)(1 - p(i))^{n-1} p(i).. \tag{9}$$

(8) - (9) yields,

$$p(i)E[\tau(i)] = \sum_{n=1}^{\infty}(1-p(i))^{n-1}p(i) = \frac{p(i)}{1-(1-p(i))} = 1.$$

Therefore $E[\tau(i)] = 1/p(i)$. □

This sends us a first encouraging message that IMS could beat exhaustive search if $p(x^*) > 2/N$ that is easily satisfied in computer vision. This "ideal" case leads us to believe that $\frac{1}{p(i)}$ and $\frac{1}{q(i)}$ are the lower bounds of $E[\tau(i)]$ for general cases, for we cannot imagine a "better" proposal than this ideal case where every state is exactly-informed. This guess will prove to be true.

Proposition 2 *For an IMS and notation above,*

$$E[\tau(i)] \geq \frac{1}{q(i)}, \quad \forall i \in \Omega$$

The equality holds at least-informed states i with $\omega(i) = \min_j\{\frac{q(j)}{p(j)}\}$.

[Proof] First we note that

$$\mathbf{K}(j,i) = q(i)\alpha(j,i) \leq q(i), \quad \forall j \neq i. \tag{10}$$

Secondly we show that

$$P(\tau(i) = k) \leq q(i)P(\tau(i) > k-1), \quad \forall k > 1. \tag{11}$$

This is obvious: if it does not hit state i in the first $k-1$ steps, the change to hit i at k-th step is smaller than $q(i)$. To show it formally, we have

$$\begin{aligned}
P(\tau(i) = k) &= P(X_1 \neq i, X_2 \neq i, \ldots, X_k = i) \\
&= \sum_{j \neq i} P(X_k = i | X_{k-1} = j)P(X_{k-1} = j, \ldots, X_2 \neq i, X_1 \neq i) \\
&= \sum_{j \neq i} \mathbf{K}(j,i)P(X_{k-1} = j, \ldots, X_2 \neq i, X_1 \neq i) \\
&\leq q(i)\sum_{j \neq i} P(X_{k-1} = j, \ldots, X_2 \neq i, X_1 \neq i) \quad --\text{use } (10) \\
&= q(i)P(X_{k-1} \neq i, \ldots, X_2 \neq i, X_1 \neq i)
\end{aligned}$$

The third step is to prove that

$$P(\tau(i) > k) \geq (1 - q(i))P(\tau(i) > k - 1). \tag{12}$$

This follows,

$$\begin{aligned} P(\tau(i) > k) &= P(\tau(i) > k - 1) - P(\tau(i) = k) \\ &\geq P(\tau(i) > k - 1) - q(i)P(\tau(i) > k - 1) \quad --\text{use } (11) \\ &= (1 - q(i))P(\tau(i) > k - 1) \end{aligned}$$

Using inequality (13) repeatedly, we have

$$P(\tau(i) > k) \geq (1 - q(i))^k P(\tau(i) > 0) = (1 - q(i))^k \tag{13}$$

From the formula (7) and inequality (13), we get the lower bound,

$$\begin{aligned} E(\tau(i)) &= 1 + \sum_{k \geq 2} P(\tau(i) \geq k) \\ &= 1 + \sum_{k \geq 1} P(\tau(i) > k) \\ &\geq 1 + \sum_{k \geq 1} (1 - q(i))^k \\ &= \sum_{k \geq 0} (1 - q(i))^k \\ &= \frac{1}{q(i)} \end{aligned}$$

The previous proof reveals that the equality $E(\tau(i)) = 1/q(i)$ holds at the least-informed cases,

$$\mathbf{K}(j, i) = q(i), \forall j \neq i \Longleftrightarrow \alpha(j, i) = 1, \forall j \neq i \Longleftrightarrow w_i \leq w_j, \forall j \Longleftrightarrow \omega(i) = \min_{j \in \Omega}\{\omega(j)\}.$$

This concludes the proof of the proposition. □

Now we are obligated to answer a question arising here. If we simply hit each state by sampling the proposal probability q, then according to proposition 1 we should hit each state at $1/q(i)$ on average which is the lower bound. Why do we need a Metropolis step for rejection?

There are at least two reasons for involving the Metropolis rejection machinery.

Firstly, in practice the Markov chain may be designed by integrating many heuristics. Each $q()$ is used in some sub-dimensions of the solution space. For example, in the DDMCMC scheme[12, 2, 13], we have proposal probabilities in the texture, color, objects subspaces independently. These bottom-up probabilities are computed locally and are conflicting, thus we need to activate the target posterior probability p to coordinate the proposals. Normally p is made of generative models.

Secondly, the IMS Markov chain assumes that it can move freely from a state i to any state j in the space Ω. In practice, this is only true in a component subspace within the whole search space. Due to practical constraints on our operators, such as split-merge, we cannot move freely from one state to an arbitrary other state. For example, it will not jump from one image segmentation state with 5 regions to a state with 10 regions in a single step. Instead it is done in a sequence of small steps. Thus the guidance of $p()$ is necessary along this path.

We cannot provide more concrete answers to the necessity of Markov chain and detailed balance in search, and further research on this topic is needed.

4.2 A formula and second lower bound $E[\tau(i)] \geq 1/p(i)$

The first lower bound leads us to wonder whether the Markov chain can hit an over-informed state with $E[\tau(i)] < 1/p(i)$. Suppose the heuristics q is very narrowly focused around the mode x^*, can the Markov chain visit x^* at fewer steps than $1/p(x^*)$? The answer turns out to be no. Intuitively, the Markov chain cannot fully trust the heuristics. It will be "counter-productive" if the heuristics over-emphasized some states.

To prove the second lower bound, we need to develop an analytical formula for $E[\tau(i)]$.

Recall from eqn (7), we have a formula

$$E[\tau(i)] = 1 + \sum_{n\geq 2}^{\infty} P(\tau(i) \geq n) = 1 + \sum_{n\geq 1}^{\infty} P(\tau(i) > n).$$

To compute $P(\tau(i) > n)$ – the probability that the Markov chain misses state i in the first n trials, we consider a new Markov process IMS2 which is the same as the IMS except that it is prohibited from visiting i. We denote it by four components

$$\text{IMS2} := < \Omega_-, q_-, p_-, \mathbf{K}_- >$$

where i is removed from the space, probabilities and matrix. $\Omega_- = \Omega - \{i\}$, and

$$q_- = (q(1), ..., q(i-1), q(i+1), ..., q(N)), \;\; p_- = (p(1), ..., p(i-1), p(i+1), ..., p(N)).$$

$\mathbf{K}_-$ is a $(N-1) \times (N-1)$ matrix with the i-th column and row removed from $\mathbf{K}$.

$$\mathbf{K}_-(k,j) = \mathbf{K}(k,j), \forall k \neq i, \; j \neq i.$$

Start from arbitrary $x_o \sim q_-$, after n steps the probability for the IMS2 state will be a $(N-1)$-vector

$$p_-^n = q_- \cdot \mathbf{K}_-^{n-1}.$$

Therefore, the sum of this vector is the total probability that IMS misses i in the first n steps.

$$P(\tau(i) > n) = \sum_{j \neq i} p_-^{n-1}(j) = q_- \mathbf{K}_-^{n-1} \mathbf{1}',$$

where $\mathbf{1} = (1, 1, ..., 1)$ is a $(N-1)$-vector with ones. Therefore,

$$\begin{aligned}
E[\tau(i)] &= 1 + \sum_{n \geq 1} P(\tau(i) > n) && (14)\\
&= 1 + \sum_{n \geq 1} q_- \mathbf{K}_-^{n-1} \mathbf{1}' && (15)\\
&= 1 + q_- [\sum_{n \geq 1} \mathbf{K}_-^{n-1}] \mathbf{1}' && (16)\\
&= 1 + q_- (\mathbf{I} - \mathbf{K}_-)^{-1} \mathbf{1}'. && (17)
\end{aligned}$$

In the above equation, $\mathbf{I}$ is an identity matrix.

Proposition 3 *In the above notation,*

$$E[\tau(i)] = 1 + q_- (\mathbf{I} - \mathbf{K}_-)^{-1} \mathbf{1}', \quad \forall i \in \Omega.$$

Using the above formula for $E(\tau(i))$ we can prove the second lower bound.

Proposition 4 *For IMS and notation above,*

$$E[\tau(i)] \geq \frac{1}{p(i)}, \quad \forall i \in \Omega.$$

The equality holds at most-informed states i with $\omega(i) = \max_j \{\frac{q(j)}{p(j)}\}$.

[Proof] We start with an IMS property that p is the invariant probability of $\mathbf{K}$. In a matrix form, it is,

$$(p(1), p(2), ..., p(N))\mathbf{K} = (p(1), p(2), ..., p(N)).$$

Let $\mathbf{K}_-(i, \cdot)$ be the i-th row of $\mathbf{K}_-$, we are interested in p_- and divide it in two terms,

$$p_- = (p(1), ..., p(i-1), p(i+1), ..., p(N)) = p_-\mathbf{K}_- + p(i)\mathbf{K}_-(i, \cdot).$$

Using the fact that $\mathbf{K}(i,j) = q(j)\alpha(i,j) \leq q(j)$, we have $\mathbf{K}_-(i, \cdot) \leq q_-$ component-wise. Then

$$p_-(\mathbf{I} - \mathbf{K}_-) = p(i)\mathbf{K}_-(i, \cdot) \leq p(i)q_-.$$

Multiply $\mathbf{1}$ to both sides of the above equation,

$$p_- \cdot \mathbf{1}' \leq p(i)q_-(\mathbf{I} - \mathbf{K}_-)^{-1}\mathbf{1}'.$$

Obviously $p_- \cdot \mathbf{1}' = 1 - p(i)$. Therefore

$$\frac{1 - p(i)}{p(i)} \leq q_-(\mathbf{I} - \mathbf{K}_-)^{-1}\mathbf{1}'.$$

Using the formula in proposition 3, we have

$$E[\tau(i)] \geq \frac{1}{p(i)}, \quad \forall i \in \Omega.$$

The equality holds if and only if $\mathbf{K}(i, \cdot) = q_-$ which implies $\alpha(i,j) = 1,\ \forall j \neq i$. Therefore $\omega(i) \geq \omega(j),\ \forall j$. So it must be a most-informed state. $\square$.

This lower bound tells that the Markov chain cannot hit i early than $1/p(i)$ no matter how much the heuristics stresses the state because it cannot "trust" the heuristics.

4.3 An explicit formula for $E(\tau(i))$

The formula in Proposition 3 involves an inverse matrix, in this subsection we seek a more analytical form. This leads to our main results. The proof is rather long and involves results from previous analysis.

Theorem 6 *Use notation in theorem 3,*

$$E[\tau(i)] = \frac{1}{p(i)(1-\lambda_i)} = \frac{1}{p(i)} \cdot \frac{1}{q(1) + \cdots + q(i-1) + \frac{q(i)}{p(i)}(p(i) + \cdots + p(N))}.$$

[Proof] Following theorem 3, we have the eigen-values of the transition matrix

$$\lambda_i = \alpha_i - \omega(i)\beta_i, \quad \text{with}$$

$$\alpha_i := q(i) + q(i+1) + \ldots + q(N) \qquad \beta_i := p(i) + p(i+1) + \ldots + p(N).$$

Let us denote by $E_j[\tau(i)]$ the expected hitting time of i if starting from state j. We will use a result from (Aldous and Fill, 1999) which states that

$$E_j[\tau(i)] = \frac{Z(i,i) - Z(j,i)}{p(i)} \tag{18}$$

where $Z(i,j) := \sum_{m \geq 0} \mathbf{K}^n(i,j) - p(j), \ \forall i, j$.

The relationship between $E[\tau(i)]$ and $E_j[\tau(i)]$ is given by the following:

$$E[\tau(i)] = 1 + \sum_{j=1}^{n} q(j) E_j[\tau(i)]. \tag{19}$$

The justification is quite simple. First one has to choose an initial state (which accounts for the 1 in the formula) and this initial state is chosen to be j with probability $q(j)$.

From (7) we get:

$$E[\tau(i)] = 1 + \sum_{j=1}^{n} q(j) \frac{Z(i,i) - Z(j,i)}{p(i)} = 1 + \sum_{j=1}^{n} \frac{q(j)}{p(i)} \sum_{m \geq 0} (\mathbf{K}^m(i,i) - \mathbf{K}^m(j,i)) \tag{20}$$

(Smith and Tierney, 1996) computed the exact m-step transition probabilities for the independence Metropolis sampler. With our notation their result is:

$$\mathbf{K}^m(j,i) = \begin{cases} p(i)(1 + \sum_{k=1}^{j-1} \frac{\lambda_k^m p(k)}{\beta_k \beta_{k+1}} - \frac{\lambda_j^m}{\beta_j}) & \text{for } j < i \\ p(i)(1 + \sum_{k=1}^{i-1} \frac{\lambda_k^m p(k)}{\beta_k \beta_{k+1}} - \frac{\lambda_i^m}{\beta_i}) + \lambda_i^m & \text{for } j = i \\ p(i)(1 + \sum_{k=1}^{i-1} \frac{\lambda_k^m p(k)}{\beta_k \beta_{k+1}} - \frac{\lambda_i^m}{\beta_i}) & \text{for } j > i \end{cases} \tag{21}$$

Therefore, we can write:

$$\mathbf{K}^m(i,i) - \mathbf{K}^m(j,i) = \begin{cases} p(i)(\sum_{k=j}^{i-1} \frac{\lambda_k^m p(k)}{\beta_k \beta_{k+1}} - \frac{\lambda_i^m}{\beta_i}) + \frac{\lambda_j^m}{\beta_j})) + \lambda_i^m & \text{for } j < i \\ 0 & \text{for } j = i \\ \lambda_i^m & \text{for } j > i \end{cases} \tag{22}$$

Using (20) in (22) one obtains

$$\begin{aligned} E[\tau(i)] &= 1+\frac{1}{p(i)}\{\sum_{j<i} q(j) \sum_{m\geq 0} [p(i)(\sum_{k=j}^{i-1} \frac{\lambda_k^m p(k)}{\beta_k\beta_{k+1}} - \frac{\lambda_i^m}{\beta_i} + \frac{\lambda_j^m}{\beta_j}) + \lambda_i^m] + \sum_{j>i} q(j) \sum_{m\geq 0} \lambda_i^m\} \\ &= 1+\frac{1}{p(i)}\{\sum_{j<i} q(j)[p(i)(\sum_{k=j}^{i-1} \frac{(\sum_m \lambda_k^m)p(k)}{\beta_k\beta_{k+1}} - \frac{\sum_m \lambda_i^m}{\beta_i} + \frac{\sum_m \lambda_j^m}{\beta_j}) + \sum_m \lambda_i^m] + \sum_{j>i} q(j) \sum_m \lambda_i^m\} \end{aligned}$$

We can simplify the above identity by noting that $\sum_{m\geq 0} \lambda^m = \frac{1}{1-\lambda}, \forall \lambda \in (0,1)$.

$$\begin{aligned} E[\tau(i)] &= 1+\frac{1}{p(i)}\{\sum_{j<i} q(j)[p(i)(\sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} \\ &\quad -\frac{1}{(1-\lambda_i)\beta_i} + \frac{1}{(1-\lambda_j)\beta_j}) + \frac{1}{1-\lambda_i}] + \sum_{j>i} \frac{q(j)}{1-\lambda_i}\} \qquad (23) \\ &= 1+\sum_{j<i} q(j)(\sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} - \frac{1}{(1-\lambda_i)\beta_i} + \frac{1}{(1-\lambda_j)\beta_j}) + \frac{\sum_{j<i} q(j) + \sum_{j>i} q(j)}{p(i)(1-\lambda_i)} \qquad (24) \\ &= 1+\sum_{j<i} q(j)(\sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} - \frac{1}{(1-\lambda_i)\beta_i} + \frac{1}{(1-\lambda_j)\beta_j}) + \frac{1-q(i)}{p(i)(1-\lambda_i)} \qquad (25) \\ &= 1+\sum_{j<i} q(j) \sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} - \frac{\sum_{j=1}^{i-1} q(j)}{(1-\lambda_i)\beta_i} + \sum_{j=1}^{i-1} \frac{q(j)}{(1-\lambda_j)\beta_j} + \frac{1-q(i)}{p(i)(1-\lambda_i)} \qquad (26) \\ &= 1+\sum_{j<i} q(j) \sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} - \frac{1-\alpha_i}{(1-\lambda_i)\beta_i} + \sum_{j=1}^{i-1} \frac{q(j)}{(1-\lambda_j)\beta_j} + \frac{1-q(i)}{p(i)(1-\lambda_i)} \qquad (27) \end{aligned}$$

Lemma (1)

$$\sum_{j<i} q(j) \sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} = \frac{1}{\beta_i} - 1 - \sum_{j=1}^{i-1} \frac{q(j)}{(1-\lambda_j)\beta_j} \qquad (28)$$

Proof

By changing the summation order in the left hand side term we get

$$\begin{aligned} \sum_{j<i} q(j) \sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} &= \sum_{k=1}^{i-1} (\sum_{j=1}^{k} q(j)) \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} \qquad (29) \\ &= \sum_{k=1}^{i-1} \frac{(1-\alpha_{k+1})p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}}. \qquad (30) \end{aligned}$$

It is easy to see that

$$\lambda_k = \alpha_k - \omega(k)\beta_k$$

$$\begin{aligned}
&= q(k) + \alpha_{k+1} - \omega(k)(p(k) + \beta_{k+1}) \\
&= \alpha_{k+1} - \omega(k)\beta_{k+1} + q(k) - \omega(k)p(k) \\
&= \alpha_{k+1} - \omega(k)\beta_{k+1} \qquad (31)
\end{aligned}$$

Therefore, $1 - \alpha_{k+1} = 1 - \lambda_k - \omega(k)\beta_{k+1}$ so we can rewrite (30) as

$$\begin{aligned}
\sum_{j<i} q(j) \sum_{k=j}^{i-1} \frac{p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} &= \sum_{k=1}^{i-1} \frac{(1-\lambda_k-\frac{\beta_{k+1}}{w_k})p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} \qquad (32) \\
&= \sum_{k=1}^{i-1} \frac{(1-\lambda_k)p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} - \sum_{k=1}^{i-1} \frac{\frac{\beta_{k+1}}{w_k}p(k)}{(1-\lambda_k)\beta_k\beta_{k+1}} \qquad (33) \\
&= \sum_{k=1}^{i-1} \frac{p(k)}{\beta_k\beta_{k+1}} - \sum_{k=1}^{i-1} \frac{q(k)}{(1-\lambda_k)\beta_k} \qquad (34)
\end{aligned}$$

Let us note that the following simple identity holds:

$$\sum_{k=1}^{i-1} \frac{p(k)}{\beta_k\beta_{k+1}} = \frac{1}{\beta_i} - 1. \qquad (35)$$

This is because $p(k) = \beta_k - \beta_{k+1} \Longrightarrow \frac{p(k)}{\beta_k\beta_{k+1}} = \frac{\beta_k-\beta_{k+1}}{\beta_k\beta_{k+1}} = \frac{1}{\beta_{k+1}} - \frac{1}{\beta_k}$. By summation over k from 1 to $i-1$ we get the identity.

Now by using (34) in (35) and changing the summation index in the second sum from k to j we notice that we got the lemma. □

Lemma (2)

$$\frac{1-\alpha_i}{(1-\lambda_i)\beta_i} = \frac{1}{\beta_i} - \frac{q(i)}{(1-\lambda_i)p(i)} \qquad (36)$$

Proof As before, $1 - \alpha_i = 1 - \lambda_i - \omega(i)\beta_i$ so

$$\frac{1-\alpha_i}{(1-\lambda_i)\beta_i} = \frac{1-\lambda_i}{(1-\lambda_i)p(i)} - \frac{\omega(i)\beta_i}{(1-\lambda_i)\beta_i} = \frac{1}{\beta_i} - \frac{\omega(i)}{(1-\lambda_i)} = \frac{1}{\beta_i} - \frac{q_i}{(1-\lambda_i)p(i)} \qquad (37)$$

□

Now,we can go back to (27) and use the results of the two lemmas. It will follow that

$$E[\tau(i)] = 1 + (\frac{1}{\beta_i} - 1 - \sum_{j=1}^{i-1} \frac{q(j)}{(1-\lambda_j)\beta_j}) - (\frac{1}{\beta_i} - \frac{q_i}{(1-\lambda_i)p(i)}) + \sum_{j=1}^{i-1} \frac{q(j)}{(1-\lambda_j)\beta_j} + \frac{1-q_i}{(1-\lambda_i)p(i)} \qquad (38)$$

After cancelations we remain with

$$E[\tau(i)] = \frac{q(i)}{(1-\lambda_i)p(i)} + \frac{1-q(i)}{(1-\lambda_i)p(i)} = \frac{1}{p(i)(1-\lambda_i)}. \qquad (39)$$

This concludes the proof of the theorem. □

Using this result we can easily get another proofs for both for the lower bounds in the previous sections.

4.4 Obtaining the upper bound

With the neat formula in the previous subsection, we can obtain the upper bound. Together with the lower bounds, we have the theorem below.

Theorem 7 *In the above notation,*

$$\frac{1}{\min\{p(i),q(i)\}} \leq E[\tau(i)] \leq \frac{1}{\min\{p(i),q(i)\}} \frac{1}{1-\|p-q\|_{TV}} \tag{40}$$

The upper bound is reached at the exactly-informed states.

[Proof] First, let us get a more tractable form for $\|p-q\|_{TV}$. We partition the state space into two sets: under-informed and over-informed with the exactly-informed states in either set.

$$\Omega = \Omega_{\text{under}} \cup \Omega_{\text{over}}.$$

As the states are sorted, so we have k being the dividing point

$$\Omega_{\text{under}} = \{i \leq k : q(i) \leq p(i)\}, \quad \Omega_{\text{over}} = \{i > k : q(i) > p(i)\}.$$

Recalling $\sum_{i\in\Omega}(p(i)-q(i)) = 0$, we have

$$\begin{aligned} \|p-q\|_{TV} &= \frac{1}{2}\sum_{i\in\Omega}|p(i)-q(i)| = \frac{1}{2}[\sum_{i\in\Omega_{\text{under}}}(p(i)-q(i)) + \sum_{i\in\Omega_{\text{over}}}(q(i)-p(i))] \\ &= \sum_{i\in\Omega_{\text{over}}} x(q(i)-p(i)) = \alpha_{k+1} - \beta_{k+1}. \end{aligned}$$

We prove the upper bound for the under-informed and over-formed sets respectively.

Case I: upper bound for under-informed states $i \leq k$.

For under-informed states, $q(i) = \min\{p(i), q(i)\}$.

From (31), we know $\lambda_i = \alpha_{i+1} - \frac{q(i)}{p(i)}\beta_{i+1}$. So

$$p(i)(1-\lambda_i) = p(i)(1-\alpha_{i+1}) + q(i)\beta_{i+1} \geq q(i)(1-\alpha_{i+1}+\beta_{i+1}).$$

Using the form of the total variation, we have for the right side

$$\min\{p(i), q(i)\}(1 - \|p - q\|_{TV}) = q(i)(1 - \alpha_{k+1} + \beta_{k+1}).$$

Thus we only need to show

$$\beta_{i+1} - \beta_{k+1} \geq \alpha_{i+1} - \alpha_{k+1}.$$

By definition, this is equivalent to

$$p_{i+1} + p_{i+2} + \ldots + p_k \geq q_{i+1} + q_{i+2} + \ldots q(k)$$

This is obviously true because states $i + 1, ..., k$ are under-informed.

The equality is attained, as noticed from the proof, when $p(i) = q_i, \forall j \in [i, k]$, which is the exactly-informed states.

Case II: upper bound for over-informed states $i > k$.

As $\min\{p(i), q(i)\} = p(i)$, it suffices to show

$$p(i)(1 - \lambda_i) \geq p(i)(1 - \alpha_{k+1} + \beta_{k+1})$$

Or,

$$\lambda_i \leq \alpha_{k+1} - \beta_{k+1} \tag{41}$$

As $\lambda_i \leq \lambda_{k+1}$, it suffices to show that

$$\begin{aligned} & \lambda_{k+1} \leq \alpha_{k+1} - \beta_{k+1} \\ \text{or} \quad & \alpha_{k+1} - \omega(k+1)\beta_{k+1} \leq \alpha_{k+1} - \beta_{k+1} \\ \text{or} \quad & \beta_{k+1}(1 - \omega(k+1)) \leq 0 \end{aligned}$$

The last step becomes trivial as $\omega(k + 1) \geq 1$ for over-informed states.

Equality in this case is obtained if $\lambda_i = \lambda_{i-1} = \ldots = \lambda_{k+1}$ and $\omega(k+1) = 1$. That is the exactly-informed state. This concludes our proof of the upper bound. □.

5 Future work

In a typical computer vision task, such as image parsing, the objective is to parse an image into its constituent patterns. The patterns include regions of coherent color, texture, or illumination, curves structures, and objects, such as human faces, vehicles, buildings, road etc. The number of components, their locations and models are all unknown. In more general problems, we need to deal with motion. The solution space is very big and consists of many subspaces of different dimensions. But we can decompose it into a number of "atomic"-spaces as the basic units, such as texture, color, motion, etc. Therefore a number of independent Metropolis samplers are used in each atomic space with its own heuristics. The Markov chain kernel is a mixture of many M sub-kernels

$$\mathbf{K}(x,y) = \sum_{\ell=1}^{M} w_\ell \mathbf{K}_\ell(x,y), \quad w_1 + \cdots + w_M = 1.$$

Unlike the IMS simple case, the states in the space are not fully-connected, i.e. $\mathbf{K}(x,y) = 0$ for most x, y. Thus the Markov chain cannot move freely from one state to the other in one step.

To analyze its behavior, we shall solve three problems in future.

1. Analyzing hitting-time by combining multiple IMS at different dimensions.

2. In some space, we must use the heuristics to form a composite proposal. Such as the Swendsen-Wang cut[2] in the partition space. Therefore the proposal probability is composed of many atomic heuristics. Analyzing the composite proposals such as Swendsen-Wang cut is also important.

3. We should also search the space in a coarse-to-fine strategy. We should discretize the search space coarsely and then refine the states over time. How to compute the hitting time for such strategy is also unknown.

Acknowledgment

This work is supported by a Sloan fellowship and an NSF grant IIS-02-44-763. The authors thank Zhuowen Tu, Yingnian Wu, and Alan Yuille for stimulating discussions, and thank Persi Diaconis and Jun Liu for insightful information.

References

[1] D. Aldous and J. Fill, Monograph on Markov chain analysis, (chapter 2). Available from David Aldous's web page at UC Berkley Statistics Department.

[2] A. Barbu and S. Zhu, "Graph Partition by Swendsen-Wang Cuts", *Proc. of ICCV*, 2003.

[3] P. Bremaud, "Markov Chains: Gibbs Fields, Monte Carlo Simulation and Queues," Springer, New-York, 1999.

[4] P. Diaconis and P. Hanlon, "Eigen analysis for some examples of the Metropolis algorithm", *Contemporary Mathematics*, **138**, 99-117, 1992.

[5] P. Diaconis and L. Saloff-Coste, "What do we know about the Metropolis algorithm?", *Journal of Computer and System Sciences*, 57, 20-36, 1998.

[6] W.K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications", *Biometrika*, 57, 97-109, 1970.

[7] J.S. Liu, "Metropolized independence sampling with comparisons to rejection sampling and importance sampling", *Statistics and Computing*, 6:113-119, 1995.

[8] K.L. Mengersen and R.L. Tweedie, "Rates of Convergence of the Hastings and Metropolis Algorithms," *Annals of Statistics*, 24:101-121, 1994.

[9] S.P. Meyn and R.L. Tweedie, *Markov chains and stochastic stability*, Springer-Verlag, London, 1993.

[10] N. Metropolis, M.N. Rosenbluth, A.W. Rosenbluth, A.H. Teller, and E. Teller, "Equations of State Calculations by Fast Computing Machines", *J. Chem. Phys.* 21, 1087-92, 1953.

[11] R.L. Smith and L. Tierney, "Exact transition probabilities for Metropolized independence sampling", *Technical Report*, Dept. of Statistics, Univ. of North Carolina, 1994.

[12] Z.W, Tu Tu and S. C. Zhu, "Image Segmentation by Data-Driven Markov Chain Monte Carlo," *IEEE Trans. PAMI*, Vol. 24, No.5, May, 2002.

[13] Z.W. Tu, X.R. Chen, A.L. Yuille, and S.C. Zhu, "Image parsing: segmentation, detection and recognition", *Proc. of ICCV*, Nice, France, 2003.

[14] S.C. Zhu, R. Zhang, and Z.W. Tu, "Integrating top-down/bottom-up for object recognition by Data-Driven Markov Chain Monte Carlo", *Proc. of Computer Vision and Pattern Recognition*, 2000.