Exact MAP states and expectations from perfect sampling: Greig, Porteous and Seheult revisited

C. Fox* and G. K. Nicholls*

*Mathematics Department, The University of Auckland, PB 92019, Auckland, New Zealand

Abstract. In 1989 Greig, Porteous and Seheult showed that the maximum *a posteriori* (MAP) state can be exactly calculated for degraded binary images. Their interest was in assessing the performance of algorithms used to find the MAP state, such as simulated annealing. A secondary conclusion was that the MAP state, at least in the restricted setting of two-colour images, does not provide a robust reconstruction of the true image. That result has been interpreted by some as indicating that the Ising MRF used by GPS is not a good prior model for such images. We show that such a judgement is premature as the MAP state does not well summarize the information in the posterior distribution in this case. In particular, the deviation of the MAP state from the mean, particularly at larger smoothing parameters, shows that the MAP state is not representative of the bulk of feasible reconstructions. We calculate other summary statistics that interpret and display the information in the posterior by implementing full Bayesian inference using a MRF prior and perfect sampling.

INTRODUCTION

Bayesian methods are now well established as a philosophical and computational route to image recovery and analysis. Specifically, it is well recognized that Bayes' rule provides an efficient way to combine the outcomes of multiple experiments and that the resulting posterior distribution is a complete statement of the resulting state of knowledge about the image. However, statistics that summarize the posterior distribution do not seem well developed particularly in the image-recovery community. There, the long-recognized need to publish a single image is typically met by finding the mode of the posterior, i.e. the maximum *a posteriori* (MAP) state. In continuous-variable inverse problems with Gaussian noise and using a Gaussian prior, the posterior distribution is unimodal symmetric and so the mode is the mean and it is clear that the MAP is unlikely to be improved upon as a single estimate. This observation, along with the computational tools available when Bayesian image recovery was being developed, has established the MAP as the estimate of choice to the point where many authors no longer question its use.

However, in any inverse problem where the forward map is non-linear, or using general prior distributions, or where image space is discrete, the posterior distribution is often skewed and modes of the posterior do not necessarily make optimal reconstructions. For high-dimensional problems, such as image recovery, there is then no reason that the mode should be representative of the bulk of probability which is dominated by metric factors rather than the value of the posterior. Instead, summary statistics are required that are based on the bulk of probability in the posterior distribution.

The advent of Markov-chain Monte Carlo (MCMC) algorithms which draw samples from the posterior has enabled statisticians to actually compute the quantities required in a genuine Bayesian analysis. A satisfying consequence (to attendees at this conference) has been the wholesale conversion of statistics departments from frequentist to Bayesian. Some very effective summary statistics have been constructed [1, 2], however those developments do not seem to have been taken up by the image-recovery community. On the other hand, the physically-based priors and likelihoods that are commonplace to researchers in inverse problems are unfortunately absent in many statistical analysis papers.

In this paper we seek to clarify the issues, for this audience, by treating a simple image recovery problem and comparing the MAP estimator to other estimates based on sampling the posterior, and to the mean which provides information about the bulk of posterior probability mass. We have chosen the problem of recovering a binary image from a pixel-wise degraded image and using an Ising Markov random field (MRF) prior model for the image. For that problem, Greig Porteous and Seheult[3] demonstrated that the MAP state can be exactly calculated via an equivalent minimum cut problem in a certain capacitated network, thereby removing any question of convergence when using iterative algorithms such as simulated annealing. We implement that calculation along with a MCMC algorithm that perfectly samples the posterior, i.e. that *actually* draws i.i.d. samples from the posterior distribution and hence for which there is no question of burn in as occurs for standard algorithms. Thus we are able to compare the MAP state with exact samples and actual estimates over the posterior distribution, there being no issues of algorithmic convergence in our results.

Our revisiting of Greig Porteous and Seheult has a somewhat different purpose to their study. Their interest was in comparing the MAP state for degraded binary images with the output of iterative algorithms, such as simulated annealing, to test convergence of those algorithms. Our interest is using their algorithm for calculating the MAP state so that we can compare with other statistics summarizing the posterior, to evaluate the usefulness of the MAP state as a robust estimator for the true image. This relates to the broader impact of Greig Porteous and Seheult which, based on their observation that the MAP state is not robust for the problem studied, was the common misconception that the failing with the MAP estimator must have been in specifying an Ising Markov random field for the prior. That inference followed from the belief, held widely at that time, that the MAP sate *must* be a good estimator. We show that the failing of the MAP estimator actually results from it being unrepresentative of the bulk of posterior probability, particularly for large smoothing parameters where individual states drawn from the posterior are themselves reasonable reconstructions. In these cases the MAP state is actually an extreme outlier.

BAYESIAN RECOVERY OF BINARY IMAGES

Consider a given $N \times N$ binary image $f = (f_1, f_2 \dots f_{N^2})$ where each $f_i \in \{0,1\}$. We measure a pixel-wise corrupted version of this image $g = (g_1, g_2 \dots g_{N^2})$ where $g_i = f_i + \epsilon_i$ with $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$ independently for each $i = 1, 2 \dots N^2$, and σ a known constant.

The likelihood for an image x given the measurements g, $\mathcal{L}(x) = \Pr\{g|x\}$, may be written using conditional independence, and that each $x_i \in \{0,1\}$, as

$$\begin{aligned}\mathcal{L}(x) &= \prod_{i=1}^{N^2} \Pr\{g_i|x_i\} = \prod_{i=1}^{N^2} (\Pr\{g_i|1\})^{x_i} (\Pr\{g_i|0\})^{1-x_i} \\ &= \prod_{i=1}^{N^2} \left(\frac{\Pr\{g_i|1\}}{\Pr\{g_i|0\}}\right)^{x_i} \prod_{i=1}^{N^2} \Pr\{g_i|0\}.\end{aligned}$$

For a given set of measurements the product $\prod_{i=1}^{N^2} \Pr\{g_i|0\}$ is a constant. In terms of the log likelihood ratios $\lambda_i = \log(\Pr\{g_i|1\} / \Pr\{g_i|0\})$ we can therefore write the likelihood as $\mathcal{L}(x) \propto \exp\left(\sum_{i=1}^{N^2} \lambda_i x_i\right)$. Since $\Pr\{g_i|x_i\} \propto \exp\left(-(x_i - g_i)^2 / 2\sigma^2\right)$ we have the simple expression

$$\lambda_i = \frac{2g_i - 1}{2\sigma^2}.$$

We specify a prior distribution by modelling x on the pixel lattice as an Ising Markov random field, with distribution $\Pr(x) \propto \exp\left(\theta \sum_{i=1}^{N^2} \sum_{j\sim i} \delta_{x_i,x_j}\right)$ where the sum over $j \sim i$ is a sum over pixel neighbours and $\delta_{a,b}$ is the indicator function for the event $a = b$. For the binary images defined here, the prior is explicitly

$$\Pr(x) \propto \exp\left(-\theta \sum_{i=1}^{N^2} \sum_{j\sim i} (2x_i - 1)(2x_j - 1)\right).$$

The joint posterior distribution for an image x given measurements g is given by Bayes' rule as

$$\Pr\{x|g\} \propto \mathcal{L}(x) \Pr(x) \propto \exp\left(\sum_{i=1}^{N^2} \lambda_i x_i - \theta \sum_{i=1}^{N^2} \sum_{j\sim i} (2x_i - 1)(2x_j - 1)\right).$$

We seek to recover the true image from this distribution.

NETWORK FORMULATION FOR MAP CALCULATION

The MAP state is

$$x_{\mathrm{MAP}} = \arg\max \Pr\{x|g\}.$$

An efficient algorithm for finding x_{MAP} was given by Greig, Porteous and Seheult [3] by considering an equivalent min-cut/max-flow problem. We now present details of that network formulation and an algorithm for its solution.

Consider a directed network with capacity constraints on its arc flows. The network consists of a node s (called the source), another node t (called the sink) and N^2 other nodes corresponding to the pixels. Flow on an arc (i,j), from node i to node j, is denoted $w_{i,j}$. The arcs are defined as follows: If $\lambda_i > 0$ there is a directed arc (s,i) from s to pixel i with flow capacity constraints $0 \leq w_{s,i} \leq c_{s,i} = \lambda_i$, otherwise if $\lambda_i < 0$ there is a directed arc (i,t) with capacity set by $c_{i,t} = -\lambda_i$. There is also a directed arc (i,j) if $i \sim j$, i.e. pixel j is a neighbour of pixel i, with capacity bound $c_{i,j} = 4\theta$. The maximum flow problem requires finding a set of feasible arc flows that maximizes the flow from source to sink.

A *cut* (that separates s and t) is a partition of nodes onto a set X containing s, and its compliment $\bar{X}$ containing t. The *flow across a cut* is defined as

$$w(X) = \sum_{i \in X, j \in \bar{X}} w_{i,j}$$

and hence equals the flow out of the set X (into $\bar{X}$). The *capacity* of a cut is

$$c(X) = \sum_{i \in X, j \in \bar{X}} c_{i,j} \tag{1}$$

which is the maximum flow that can flow out of X. Since the flow out of the source, across a cut, is bounded by the capacity of the cut, and that is true for any cut, the minimum capacity over all cuts bounds the maximum flow achievable in the network. The min cut - max flow theorem due to Ford and Fulkerson states that in fact the maximum flow equals the minimum cut capacity[4]. A corresponding cut is called a minimum cut.

Given a cut X we define node labels $\{x_i\}_{i=1,2,\cdots N^2}$ by: $x_i = 1$ if $i \in X$ otherwise $i \in \bar{X}$ and we set $x_i = 0$. Eqn. 1 can then be written by splitting the sum to sums over the source, sink, and pixel nodes to give

$$\begin{aligned} c(X) &= \sum_{j \in \bar{X}} c_{s,j} + \sum_{i \in X} c_{i,t} + \sum_{i \in X \backslash \{s\}, j \in \bar{X} \backslash \{t\}} c_{i,j} \\ &= \sum_{\lambda_j > 0} (1 - x_j) \lambda_j + \sum_{\lambda_i < 0} x_i (-\lambda_i) + \frac{1}{2} \sum_{i=1}^{N^2} \sum_{j \sim i} 4\theta (x_i - x_j)^2 \end{aligned}$$

where the factor 1/2 in the last term occurs because arcs out of and in to X are included in the sum. Collecting terms gives the cut capacity in the simple form

$$c(X) = -\sum_{i=1}^{N^2} x_j \lambda_j + \theta \sum_{i=1}^{N^2} \sum_{j \sim i} 2(x_i + x_j) - 4x_i x_j \tag{2}$$

which differs from $-\Pr\{x|g\}$ by a constant. Hence the labelling of the minimum cut gives the MAP state.

The maximum flow, and hence the minimum cut, can be constructed by the (Ford-Fulkerson) algorithm in which a feasible flow is repeatedly examined to find a chain of arcs from s to t through which an increase in flow can be sent and then increasing

$\underline{X,p = \text{label}(c,w,s)}$

$T \leftarrow \{s\}$

$X \leftarrow \{\}$

while $T \neq \{\}$

  $U \leftarrow \{\}$

  for $i \in T$

    for $k \in \{k = (i,j)$ is an arc : $j \notin X$ and $w_k < c_k\}$

      $U \leftarrow U \cup \{j\}, p_j = k$

    for $k \in \{k = (j,i)$ is an arc : $j \notin X$ and $w_k > 0\}$

      $U \leftarrow U \cup \{j\}, p_j = k$

  $X \leftarrow X \cup T$

  $T \leftarrow U$

Algorithm 1: A breadth-first search label-setting algorithm used to find a flow-augmenting path. During the algorithm, X contains the currently labelled and nodes that have been scanned, while U contains the unscanned labelled nodes. The two inner-most for loops scan node i for augmentable forward and backward arcs, respectively, to an unlabelled node. Any such arcs are stored in p for subsequent construction of a flow-augmenting path.

the flow if possible. If such a *flow-augmenting path* exists, it can be constructed using a label-setting algorithm. Alg. 1 gives a suitable breadth-first search algorithm (cf. Prop. 2.2 of Bertsekas [4]) that terminates with either the sink t labelled, in which case there exists a flow-augmenting path, or without t labelled, in which case the labelling X defines a "saturated" cut for which the flow equals the cut capacity. The latter case occurs only when w is the maximum flow and implies that the cut is the minimum cut that we seek. The Ford-Fulkerson successive-labelling algorithm is shown in Alg. 2.

$\underline{X,w = \text{FordFulk}(c,s,t)}$

$w \leftarrow 0$

$X,p \leftarrow \text{label}(c,w,s)$

while $t \in X$

  Construct a flow-augmenting path P using p

  $w \leftarrow w +$ maximum increase of flow along P

  $X,p \leftarrow \text{label}(c,w,s)$

Algorithm 2: The Ford-Fulkerson successive labelling algorithm for maximum flow.

The algorithm terminates having found the maximum flow, though it is actually the final labelling, of the minimum cut, that gives the MAP state.

For the particular topology of networks generated for MAP calculation we simplified the general labelling algorithm by labelling and scanning the source as an initial step and used the general scanning algorithm for the pixel nodes, only, with each scan first checking to see if an augmentable arc exists to the sink. Since neighbouring pixels are connected by directed links in each direction, we actually stored the flows as $w_{i,j} = -w_{j,i}$, for i, j pixel nodes, and so only the first test for augmentable arcs is needed, also simplifying the path-finding step as no "backward" arcs occur. Our MatLab implementation on a PIII 450 took about 700 seconds to converge for the 64×64 examples given in section . Considerable speed up could have been achieved using the modification described by GPS, though better again would be to use the preflow-push algorithm due to Karzanov[4].

PERFECTLY SAMPLING A BINARY MRF

Propp and Wilson[5] give an algorithm for perfectly sampling the Ising model as a first example of perfect simulation. The algorithm they describe is a Gibbs sampler, and could be used directly to sample the distribution we treat here. The perfect simulation algorithm described here is derived from a Metropolis sampler, and for which a C-code implementation has been available on the web at

$$\texttt{http://www.math.auckland.ac.nz/\~{}nicholls/}$$

since 1996. We recommend interested readers see the article by Dimakos[6] for a simple introduction to these topics, and the site

$$\texttt{http://dimacs.rutgers.edu/\~{}dbwilson/exact/}$$

for a substantial bibliography.

Let $\Omega = \{0,1\}^{N^2}$ denote the space of $N \times N$ binary images. Let $X^{(t)}$, $t = -1, -2, -3 \ldots$ be a Markov chain of random variables taking values x in Ω with probability $\pi_x = \Pr\{X^{(t)} = x\}$. In this paper π corresponds to the posterior distribution for a binary image of 1's and 0's, observed with additive Gaussian noise as described earlier. Under the Ising binary Markov random field prior the posterior probability for the unknown true image f to be some particular image x given data g is

$$\ln(\pi_x) = -\ln(Z) - \theta \sum_{i=1}^{N^2} \sum_{j \sim i} (2x_i - 1)(2x_j - 1) - \sum_{i=1}^{N^2} (2x_i - 1) g_i / \sigma^2.$$

In this formula Z is an unknown but finite normalizing constant, and the sum over $j \sim i$ is a sum over all neighbours on a square lattice with free boundary conditions.

The transition probability $P_{x,y}$ between two states $x, y \in \Omega$ is defined implicitly by the (Metropolis Hastings) stochastic update algorithm given in Alg. 3. Let $u = (u^{(-1)}, u^{(-2)}, \ldots)$ denote a fixed stream of random triples $u^{(t)} = (u_1^{(t)}, u_2^{(t)}, u_3^{(t)})$. For $t < 0$ and $a = 1, 2, 3$, each variate $u_a^{(t)}$ is supposed to be a realization of a corresponding uniform random variable $U_a^{(t)} \sim \mathrm{U}(0,1)$.

Consider the partial ordering $x \leq y$ iff $x_i \leq y_i$ for each $i = 1, 2 \ldots N^2$. The extremal states are **1** (all white) and **0** (all black). The update given in Alg. 3 is *stochastically*

$\underline{y = \phi(x,v)}$
$i \leftarrow \lceil v_1 N^2 \rceil$
$b \leftarrow \lceil v_2 - \frac{1}{2} \rceil$
$x' \leftarrow (x_1, x_2 \ldots x_{i-1}, b, x_{i+1} \ldots x_{N^2})$
if $v_3 \leq \min(1, \pi_{x'}/\pi_x)$
  $y = x'$
else
  $y = x$

Algorithm 3: Algorithm evaluating $\phi(x,v)$, where $x \in \Omega$ is the current image state, and π is the target distribution, while $v = (v_1, v_2, v_3)$ are U$(0,1)$, independently realised, random numbers. For $r \in \mathbb{R}$, $\lceil r \rceil$ is the smallest integer greater than r.

monotone in the sense that $x \leq y \Rightarrow \phi(x,v) \leq \phi(y,v)$ whenever $\theta > 0$. The apparently "stupid" idea of choosing the new colour at a pixel independent of the present colour (and hence possibly wasting time by proposing no change at all) ensures that this works.

Let $T > 0$ be a fixed integer. Imagine two realizations $\check{x}_{-T}(t)$ and $\hat{x}_{-T}(t)$ of the Markov chain, initialized with $\check{x}_{-T}(-T) = \mathbf{0}$ and $\hat{x}_{-T}(-T) = \mathbf{1}$, and simulated for $-T \leq t \leq 0$ using

$$\check{x}_{-T}(t+1) = \phi(\check{x}_{-T}(t), u^{(t)})$$

and

$$\hat{x}_{-T}(t+1) = \phi(\hat{x}(t)_{-T}, u^{(t)}).$$

The two realizations share a common stream u of random numbers. It follows that $\check{x}_{-T}(t) \leq \hat{x}_{-T}(t)$ for each $-T \leq t < 0$.

Now, fix the random number stream u and let $\tau > 0$ be the smallest positive integer such that $\check{x}_{-\tau}(0) = \hat{x}_{-\tau}(0)$. We call τ a coalescence time and use $x_{-\tau}(0) = \check{x}_{-\tau}(0) = \hat{x}_{-\tau}(0)$ to denote this particular state. The coalescence time is the number of updates we must go back in time so that the two paths $\check{x}_{-T}$ and $\hat{x}_{-T}$ coalesce before $t = 0$ is reached. We will see that

$$\lim_{T \to \infty} x_{-T}(0) = x_{-\tau}(0). \tag{3}$$

Note that $x_{-T}(0)$ is the result of simulation from $-T$ up to $t = 0$ using the same random number stream u used to simulate $\check{x}_{-\tau}(t)$ and $\hat{x}_{-\tau}(t)$. Eqn. 3 follows since for $-T \leq -\tau$ we have $\mathbf{0} \leq x_{-T}(-\tau) \leq \mathbf{1}$ and hence $\check{x}_{-\tau}(t) \leq x_{-T}(t) \leq \hat{x}_{-\tau}(t)$ for $-\tau \leq t \leq 0$. All paths are squeezed between the coalescing paths $\check{x}_{-T}$ and $\hat{x}_{-T}$, and hence simulation from any backward time $-T < -\tau$, carried out using the random number stream u, must realize $x_{-\tau}(0)$ at $t = 0$. Since the Metropolis Hastings algorithm we have given specifies an irreducible transition matrix, reversible with respect to π, it follows that $X(t)$ is ergodic and hence $x_{-\tau}(0) \sim \pi$ as required.

It is not necessary to find τ exactly in order to compute the unique $x(0)$ determined by a given stream u of uniform random numbers. One may simply test for coalescence, simulating forward to $t = 0$ from some decreasing sequence of $-T$-values (for example

$T = 1,2,4,8\ldots$). Once T exceeds τ, coalescence will occur. The algorithm $\psi(u,T)$ given in Alg. 4 uses this method to determine the perfect sample $x(0)$ corresponding to the

$x(0) = \psi(u,T)$:

$t \leftarrow -T$

$\check{x}_{-T}(-T) \leftarrow \mathbf{0}$

$\hat{x}_{-T}(-T) \leftarrow \mathbf{1}$

do

  $\check{x}_{-T}(t+1) = \phi(\check{x}_{-T}(t), u^{(t)})$

  $\hat{x}_{-T}(t+1) = \phi(\hat{x}_{-T}(t), u^{(t)})$

  $t \leftarrow t+1$

while $t < 0$

if $\hat{x}_{-T}(0) = \check{x}_{-T}(0)$

  $x(0) = \hat{x}_{-T}(0)$

else

  $x(0) = \psi(u, 2T)$

Algorithm 4: Algorithm evaluating $\psi(u,T)$ using a stream of u triples $u^{(t)}$ of IID U$(0,1)$ random numbers associated with times $t = -1,-2,-3\ldots$, and $T > 0$ a given integer. The return value $\hat{x}$ is distributed according to π.

input random number stream u. Note that one must be careful to fix u^t throughout these repeated simulations. Once a perfect sample $x(0)$ is realized, an independent stream is taken for the simulation of the next independent perfect sample.

It remains to show that the distribution of τ is concentrated at sufficiently small values to make the algorithm workable. In fact we find that for large values of the smoothing parameter θ, and large image size N^2, coalescence does not occur in any useful time. This corresponds to θ-values for which the Ising model has separated phases ($\theta \gtrsim 0.44069$ in the limit $N \to \infty$, but the effective bound at which coalescence is lost for practical purposes is higher on smaller lattices). The simulations given here were not tedious, however, as the presence of data tends to bring the coevolving states together, and thereby hastens coalescence. For each θ-value explored, 1000 perfect samples were generated. This took a few minutes for $\theta = 0.125$ and about 2 hours for $\theta = 0.675$ on a machine with SPECfp95 equal about 12.

PROSPECTS FOR PERFECT SIMULATION

It is not clear yet whether perfect simulation will prove to be of practical value, outside a handful of elegant applications in statistical mechanics [5, 7]. Families of algorithms have been given for spatial point processes [8] which may yet prove valuable.

One of the key difficulties has been to give perfect simulation algorithms for the case of repulsive potentials (notice that the algorithm we give here fails for $\theta < 0$, the repulsive case). Algorithms have been given to treat this case, with the first algorithms being given in the treatment of point processes. However point processes have a number of features making them tractable, not the least of which is the ease with which events in 2-dimensional point processes may be pictured in the mind.

Little progress has been made on problems of practical value for sample based inference in statistics. Interesting exceptions include the recent work of Green and Murdoch[9] and Lund[10]. One of us (GN) has given an algorithm which is able to perfectly sample some such problems [11], though it still requires a certain amount of imagination to implement that framework as, generally, case-specific details must be exploited if a reasonably efficient algorithm is to be achieved. However, it is possible to treat real sampling problems, well beyond the complexity of those described by Møller and Nicholls[11].

Perfect sampling must be impracticable for certain problems without knowledge of case-specific detail, for example sampling uniformly at random from the Hamiltonian cycles of a large random graph. However we judge that perfect simulation will be practicable for distributions of the kind which typically appear in applied Bayesian analysis, for moderate sized problems. We have in mind problems with little simplifying analytical structure in the way of symmetry, or conditional independence, and of the order of twenty or thirty variables.

NUMERICAL EXPERIMENTS

We now give examples of the MAP state, as well as images calculated as expectations, for a simple synthetic data set. The initial image consists of a 64×64 binary image shown in Fig. 1 (true) with 0 shown as black and 1 shown as white. The data consists of a degraded version of this resulting from pixel-wise addition of zero-mean Gaussian noise with standard deviation $\sigma = 1$. The degraded image is shown in Fig. 1 (noisy) as a grey-scale image.

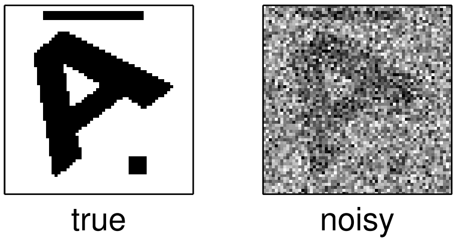

FIGURE 1. Original (true) and noise-degraded (noisy) images.

In Fig. 2 we show the MAP state along with the mean, a sample from the posterior and the marginal posterior mode (MPM). The latter shows each pixel as the mode of the marginal distribution of that pixel, and hence takes the value that the pixel most frequently took in the samples drawn from the posterior. For the case of binary images, the MPM is just the thresholded mean, $x_{\mathrm{MPM}} = [x_{\mathrm{mean}}]$. It would be interesting to repeat these experiments also using Baddeley's delta loss [1]. Each of the MAP, mean, sample, and MPM are shown for the smoothing parameters $\theta = 0.125, 0.25, 0.375, 0.5, 0.625$.

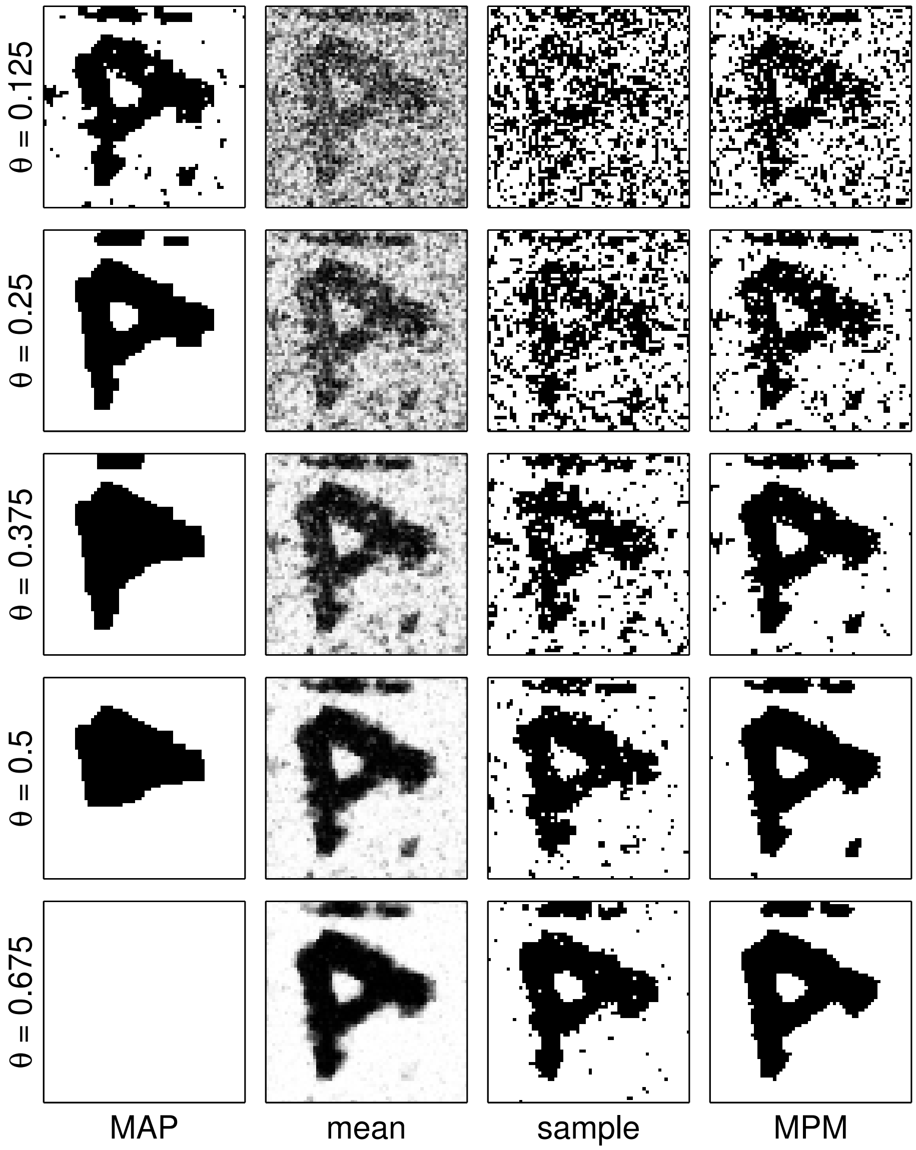

FIGURE 2. Tableau of maximum *a posteriori* (MAP) state, mean, a single sample from the posterior, and the marginal posterior mode (MPM) for a range of smoothing parameters θ.

Note that as the smoothing parameter θ increases, x_{MAP} becomes smoother and goes from being a reasonable recovered image at $\theta \approx 0.25$, first loosing the centre of the A, then the "legs" and finally, for all θ greater than some critical value in the range $(0.5, 0.675)$, $x_{\mathrm{MAP}} = \mathbf{1}$, i.e. the all-white state.

The mean image indicates that the position of the bulk of posterior probability mass, clusters increasingly around better reconstructions as θ increases, with good reconstructions for $\theta \gtrsim 0.5$. While the mean is not a feasible reconstruction for binary images, for large $\theta \gtrsim 0.5$ it is "close" to the MPM which does provide a good recovered image in that range.

The sample images show what a typical state from the posterior looks like. Across the range of θ the MAP state never looks like a typical state and, indeed, for larger θ is something of an outlier. One could argue that for θ in the range $(0.125, 0.25)$ x_{MAP} makes a good recovered image with the square feature beneath the A being present and with the image not too spotty. But then one needs to choose θ carefully, reflecting the standard practice of having to be particular about the smoothing parameter in regularized inversion. A greater complaint about x_{MAP} as an estimator of x_{true} is that at larger smoothing parameters, $\theta \gtrsim 0.5$ for this example, when the prior is doing an excellent job of shaping the posterior so that the bulk of posterior probability mass contains smooth images that fit the data well and themselves make good reconstructions, the MAP state is a hopeless reconstruction precisely because it is entirely unrepresentative of typical samples. For $\theta \gtrsim 0.675$ this situation is extreme: Then the MAP state is an extreme outlier and is completely useless, while the posterior is dominated by states from which a good recovered image could be formed.

CONCLUSIONS

The commonly stated view that x_{MAP} fails for large smoothing parameter because the Ising MRF causes the posterior to only have non-zero probability mass near the completely flat image $x = \mathbf{1}$ is clearly untrue for this example. Instead we find that using an Ising MRF prior causes the bulk of posterior probability mass to be centred around states that are good reconstructions across a wide range of smoothing parameters – including θ significantly greater than the value giving separated phases in the infinite Ising model. It seems clear that good image estimators can be found that do not have the sensitivity on θ displayed by the MAP state. Indeed, very effective estimators, based on novel loss functions, have been available for some time [2]. For the case of recovering a pixel-wise degraded binary image, we found that reconstructions based on x_{MAP} do not make good reconstructions and show sensitivity to θ precisely because that state is not representative of the bulk of feasible reconstructions.

We conjecture that those properties of x_{MAP} commonly occur in other image recovery problems where the mode is not necessarily the mean. For those common problems we believe that the focus in the image recovery community should move away from finding the MAP state, and that research effort is needed to find efficient ways of calculating those statistics that do provide good reconstructions and are insensitive to choices in the prior such as our smoothing parameter θ. It is clear that such statistics exist for our example, and we expect also for the majority of image recovery problems.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge financial support for this work from the Marsden Fund under grant UOA812, and heartily thank Andy Philpott for a quick lesson on min cut - max flow.

REFERENCES

1. D. L. Wilson, A. J. Baddeley, and R. A. Owens, "A new metric for grey-scale image comparison," *International Journal of Computer Vision*, **24**, pp. 5–17, 1997.
2. H. Rue, "New Loss Functions in Bayesian Imaging" *Journal of the American Statistical Association*, **90**, pp. 900–908, 1995.
3. D. M. Greig, B. T. Porteous, and A. H. Seheult, "Exact maximum *a posteriori* estimation for binar images," *J. R. Statist. Soc.* **B**, **51**, pp. 271–279, 1989.
4. D. P. Bertsekas, *Linear Network Optimization*, The MIT Press, Cambridge, Massachusetts, USA, 1992.
5. J. G. Propp and D. B. Wilson, "Exact sampling with coupled markov chains and applications to statistical mechanics," *Random Structures and Algorithms*, **9**, pp. 223–252, 1996.
6. X. K. Dimakos, "A guide to exact simulation," *International Statistical Review*, 1999. To appear.
7. O. Häggström, M. N. M. van Lieshout, and J. Møller, "Characterisation results and Markov chain Monte Carlo algorithms including exact simulation for some spatial point processes," *Bernoulli*, **5**, pp. 641–659, 1999.
8. W. S. Kendall and J. Møller, "Perfect Metropolis-Hastings simulation of locally stable point processes," Tech. Rep. R-99-2001, Department of Mathematics, Aalborg University, 1999.
9. P. J. Green and D. J. Murdoch, "Exact sampling for Bayesian inference: towards general purpose algorithms," in *Bayesian Statistics 6*, J. Bernardo, J. Berger, A. Dawid, and A. Smith, eds., Oxford University Press, 1999. Presented as an invited paper at the 6th Valencia International Meeting on Bayesian Statistics, Alcossebre, Spain, June 1998.
10. J. Lund, *Statistical inference and perfect simulation for point processes observed with noise*. PhD thesis, The Royal Veterinary and Agricultural University, Copenhagen, Department of Statistics, 1999.
11. J. Møller and G. K. Nicholls, "Perfect simulation for statistical inference," Tech. Rep. 1999:33, Department of Mathematical Statistics, Chalmers University of Technology, Göteborg, 1999.