Delayed Rejection in Reversible Jump Metropolis-Hastings

Peter J. Green[*]
University of Bristol, UK.

Antonietta Mira[†]
Università dell'Insubria, Varese, Italy.

Abstract

In a Metropolis-Hastings algorithm, rejection of proposed moves is an intrinsic part of ensuring that the chain converges to the intended target distribution. However, persistent rejection, perhaps in particular parts of the state space, may indicate that locally the proposal distribution is badly calibrated to the target. As an alternative to careful off-line tuning of state-dependent proposals, the basic algorithm can be modified so that on rejection, a second attempt to move is made. A different proposal can be generated from a new distribution, that is allowed to depend on the previously rejected proposal. We generalise this idea of delaying the rejection and adapting the proposal distribution, due to Tierney and Mira (1999), to generate a more flexible class of methods, that in particular applies to a variable dimension setting. The approach is illustrated by two pedagogical examples, and a more realistic application, to a change-point analysis for point processes.
Keywords: Adaptive, Change points, Efficiency of MCMC estimates, Integrated autocorrelation time, Peskun ordering.

1 Introduction

Metropolis-Hastings kernels (Hastings, 1970) are used in almost all practical Markov chain Monte Carlo (MCMC) methods. In a Metropolis-Hastings update, proposed new values for some or all variables are generated from a proposal distribution of convenience, and then an accept/reject decision is made, depending on the generated values and on the form of the proposal and target distributions. In the case of rejection, the chain remains in the current state, time is advanced, and the next transition considered.

Although remaining in the current state contributes to preserving the stationary distribution through detailed balance, intuitively it increases autocorrelation in the realised chain; substance is given to this intuition by a result stated, and proved for the case of a finite state space, by Peskun (1973); a proof for general state spaces is given by Tierney (1998). This result says that given two transition kernels P and Q both reversible with respect to a given

[*]Department of Mathematics, University of Bristol, Bristol BS8 1TW, UK.
Email: P.J.Green@bristol.ac.uk
[†]Department of Economics, Università dell'Insubria, Via Ravasi 2, 21100 Varese, Italy.
Email: anto@ipvaim.unipv.it

distribution π, with P dominating Q off the diagonal in the sense that $P(x, B) \geq Q(x, B)$ for all x and all measurable B not containing x, then the asymptotic variances of empirical averages of *any* function of x are smaller in a chain generated using P than in one using Q. Loosely speaking, performance is improved by shifting probability off the diagonal, that is, by reducing the probability of remaining in the current state.

This idea is exploited in the method of Tierney and Mira (1999). When a reject decision is taken, instead of turning to the next transition, suppose we make another attempt to move: that is, we make a second proposal, using a different distribution possibly dependent on the rejected value, and accept or reject that second attempt using a suitably computed probability. This certainly reduces the overall probability of remaining in the current state and thus leads to an improved sampler in the Peskun sense. Whether such a modification is useful in practice depends on whether the reduction in variance obtained compensates for the additional computational cost. The process can be repeated, with more attempts being made if the second is rejected.

An interesting feature of this scheme is that the proposal distribution at the second stage is allowed to depend on the rejected value at the first stage as well as on the current position of the chain. Fortunately, allowing the proposal distribution to learn from previously rejected candidates (within the same sweep) does not destroy the Markovian property of the sampler, thus all the asymptotic Markov chain theory used for standard MCMC methods holds true.

The method of Tierney and Mira (1999) achieves detailed balance overall, by imposing it separately at each stage of this iterated proposal process. Consideration of reversibility leads to balancing the probability of, for example, starting at x, proposing and rejecting a move to y and then proposing and accepting a move to z, against that of the reverse path from z via y to x. It turns out that requiring the intermediate state y to be the same in each direction is an obstacle to making the fullest use of the idea both in terms of efficiency and in terms of the range of problems to which it can be applied. In particular, it inhibits application to variable-dimension samplers.

In the present paper we generalise the strategy in Tierney and Mira (1999), which we call 'delayed rejection', to remove these obstacles. We set delayed rejection in the formalism of the reversible jump approach (Green, 1995) to Metropolis-Hastings in a general state space. The possible efficiency gains of delaying rejection are thus combined with the flexibility of reversible jump.

The paper is structured as follows. In Section 2, after introducing a pedagogical example in a fixed dimension context, we review the Tierney and Mira approach, and reversible jump, respectively. Our extension of the delayed rejection algorithm (DRA) to a general state space is given in Section 3, which includes a continuation on the pedagogical example. In Section 4, we spell out details of the dimension-matching needed in a variable-dimension problem, again illustrated by an example. Section 5 includes a more significant application, to a Bayesian analysis of multiple change points in a point process; this requires a further methodological extension to allow random choice of moves in the DRA. In Section 6 we report the results of some experiments comparing the performance of the DRA strategy with more standard Metropolis-Hastings. These are performed on the pedagogical example and on change point application introduced earlier. We close with a brief mention of extensions, and a discussion of the potential of the method in other applications.

2 Background and motivation

We begin by outlining a pedagogical example for the purpose of making the theoretical concepts introduced in the first part of the paper more accessible. This example will be revisited in Sections 3.2 and 6.1. We then present the two main ideas that we combine to generate a new flexible and efficient class of MCMC samplers: the delaying rejection strategy (Section 2.2) and the reversible jumps framework (Section 2.3).

2.1 A pedagogical example

Consider a target distribution given by an equally weighted mixture of two uniforms: $\pi(x) = (1/2\delta_1)\mathbf{1}_{[-\delta_1,0]}(x) + (1/2\delta_2)\mathbf{1}_{(0,\delta_2]}(x)$. When the value of δ_2 is big relative to δ_1 the shape of this distribution is such that the "optimal" spread of the proposal depends on the current position of the chain. When in the interval $[-\delta_1, 0]$ the spread should be quite small otherwise proposals are likely to be rejected. On the other hand, using the same small spread when in $(0, \delta_2]$ will give high acceptance rate, but the chain is going to mix (explore this portion of the support) very slowly.

Every time we find ourselves in such situations (and this happens quite often for multidimensional target distributions), the DRA can be of great help. Here is an example of what the delaying rejection strategy would allow you to do.

Generate the first stage proposal y from a normal distribution centered at the current position of the chain x. Accept this proposal with the usual Metropolis-Hastings probability: $\alpha_1(x, y) = \min\ \{1, \pi(y)/\pi(x)\}$. In case of rejection, instead of retaining the same position and advancing time, propose a second stage candidate z by generating it from a normal centered at x but with different variance. The second stage acceptance probability is

$$\alpha_2(x, z) = \min\left\{1, \frac{\pi(z)[1 - \alpha_1(z, y^\star)]}{\pi(x)[1 - \alpha_1(x, y)]}\right\}, \tag{1}$$

where $y^\star$ could be equal either to y as in the Tierney and Mira original recipe for the DRA, or, in more generality, to some function of x and y. In particular we will show in Section 3.2 that the choice $y^\star = z - (y - x)$ is a valid one.

2.2 The delayed rejection algorithm

In this section, we suppose that the target and all proposal distributions have a common dominating measure, typically Lebesgue, and so we work with their density functions.

Our target is a distribution π on $\mathcal{X} \subset \mathcal{R}^d$. When at x, we propose a new state y with density $q_1(x, y)$. Following Hastings (1970), this is accepted with probability

$$\alpha_1(x, y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\frac{q_1(y, x)}{q_1(x, y)}\right\}. \tag{2}$$

Tierney and Mira (1999) propose that if the move to y is rejected, a second proposal is made, say z with density $q_2(x, y, z)$. The acceptance probability of the new candidate has to be determined in order to preserve the stationary distribution; Tierney and Mira (1999) use

$$\alpha_2(x, z) = \min\left\{1, \frac{\pi(z)q_1(z, y)q_2(z, y, x)[1 - \alpha_1(z, y)]}{\pi(x)q_1(x, y)q_2(x, y, z)[1 - \alpha_1(x, y)]}\right\}, \tag{3}$$

that preserves detailed balance (DB) at the second stage, as shown in Mira (1998). In the pedagogical example, since the proposal densities are symmetric they cancel out in the ratio appearing in the acceptance probability.

If the candidate at the second stage is also rejected we could either stay in the current state x or move on, to a third stage, and so on. Since DB is imposed on each stage separately, it is valid to make any fixed or random number of attempts. A recursive expression for the i^{th} stage acceptance probability is given in Mira (1998). In the pedagogical example we could start with a first stage proposal with large variance and then reduce the proposal variances of each successive stage by a factor of 2. As noted before, the advantage of this strategy is apparent if the optimal spread of the proposal is not constant throughout the state space.

Notice that in order to obtain the numerator of the second stage acceptance probability we need to compute the probability that a fictional stationary Markov chain started in z proposes a move to y at the first stage, rejects it and accepts a second stage move to x. This is only a mental experiment and this chain is not in fact implemented, but thinking of the second stage acceptance probability in these terms will help in understanding the proposed modification of the DRA.

2.3 Reversible jump methods

The usual formulation of Metropolis-Hastings in terms of densities is a little restrictive when we come to consider MCMC samplers for certain tasks, for example where there is no elementary dominating measure for the target distribution. From now on, we work with the probability measures themselves.

The DB condition for a general transition kernel P and its invariant distribution π is written in integral form as

$$\int_{(x,y)\in A\times B} \pi(dx)P(x,dy) = \int_{(x,y)\in A\times B} \pi(dy)P(y,dx) \tag{4}$$

for all Borel sets $A, B \subset \mathcal{X}$. In Metropolis-Hastings, we make a transition by first drawing a proposed new state y from the proposal measure $q(x,dy)$ and then accepting it with probability $\alpha(x,y)$. If we reject, we stay in the current state, so that $P(x,dy)$ has an atom at x. This makes an equal contribution to each side of equation (4), so can be neglected, and we are left with the requirement

$$\int_{(x,y)\in A\times B} \pi(dx)q(x,dy)\alpha(x,y) = \int_{(x,y)\in A\times B} \pi(dy)q(y,dx)\alpha(y,x). \tag{5}$$

It can be shown (Green, 1995; Tierney, 1998) that $\pi(dx)q(x,dy)$ is dominated by a *symmetric* measure μ on $\mathcal{X}\times\mathcal{X}$, and has density (Radon-Nikodym derivative) f with respect to μ. Then (5) becomes

$$\int_{(x,y)\in A\times B} \alpha(x,y)f(x,y)\mu(dx,dy) = \int_{(x,y)\in A\times B} \alpha(y,x)f(y,x)\mu(dy,dx) \tag{6}$$

and, using the symmetry of μ, this is clearly satisfied for all appropriate A, B if

$$\alpha(x,y) = \min\left\{1, \frac{f(y,x)}{f(x,y)}\right\}. \tag{7}$$

If we wrote this rather more informally as

$$\alpha(x, y) = \min\left\{1, \frac{\pi(dy)q(y, dx)}{\pi(dx)q(x, dy)}\right\} \tag{8}$$

then the similarity with the usual expression (2) using densities is apparent.

This may seem rather abstract. Fortunately, in many cases, the dominating measure and Radon-Nikodym derivatives can be generated almost automatically, by considering how the transition will actually be implemented. Take the case where $\mathcal{X} \subset \mathcal{R}^d$, and suppose π has a density (also denoted π) with respect to d–dimensional Lebesgue measure. At the current state x, we generate, say, r random numbers u from a known density g, and then form the proposed new state as some suitable deterministic function of the current state and the random numbers: $y = h^+(x, u)$, say. In the pedagogical example we have taken u to be a standard normal random variable and set $y = x + \sigma_1 u$ where σ_1 is the spread of the first stage proposal. We can then re-express the left-hand side of (5) as an integral with respect to (x, u):

$$\int_{(x,y)\in A\times B} dx\, du\; \pi(x)g(u)\alpha(x, y).$$

Now consider how the reverse transition from y to x would be made, with the aid of random numbers $u' \sim g'$ giving $x = h^-(y, u')$. The superscripts $+$ and $-$ indicate the different functions that could be used in each direction though these functions might often be the same in practice. If the transformation from (x, u) to (y, u') is a diffeomorphism (injective, differentiable and with continuous derivative), then we can first re-express the right-hand side of (5) as an integral with respect to (y, u'), and then apply the standard change-of-variable formula, to see that the $(d + r)$–dimensional integral equality (5) holds if

$$\pi(x)g(u)\alpha(x, y) = \pi(y)g'(u')\alpha(y, x)\left|\frac{\partial(y, u')}{\partial(x, u)}\right|,$$

whence a valid choice for α is

$$\alpha(x, y) = \min\left\{1, \frac{\pi(y)g'(u')}{\pi(x)g(u)}\left|\frac{\partial(y, u')}{\partial(x, u)}\right|\right\}. \tag{9}$$

It is often easier to work with this expression than with the usual one, equation (2). This reversible jump formalism perhaps appears a little cumbersome, but proves a flexible framework for constructing quite complex moves using only elementary calculus. In particular, the possibility that $r < d$ covers the case, typical in practice, that given $x \in \mathcal{X}$, only a lower-dimensional subset of $\mathcal{X}$ is reachable in one step, where this subset need not be parallel to coordinate axes. Note that there is a deliberate redundancy in separating the generation of the random innovation u and the calculation of the proposal value through the deterministic function $y = h^+(x, u)$. This allows the same proposal distribution to be expressed in many different ways, a choice that can be exploited to the user's convenience. The real bonus of this formalism, however, is that expression (9) applies, without change, in a variable dimension context. Provided that the transformation from (x, u) to (y, u') is a diffeomorphism, the individual dimensions of x and y (and hence of u and u') can be different. In this and other contexts, the proposal densities g and g' may be conditioned on x and y respectively.

3 Generalising the delayed rejection algorithm

Detailed balance with respect to some distribution is a sufficient but not necessary condition for that distribution to be preserved by a Markov chain. Furthermore, imposing DB still allows considerable freedom in constructing the Markov chain so that the acceptance probability in (3) is, by no means, the only second stage acceptance probability that ensures that π is the stationary distribution of the chain. In particular, the acceptance probability in (3) is obtained under the implicit assumption that the return path of a fictional chain started from z and moving to x at the second stage, proposes a move to y at the first stage. This is really not a required condition to preserve the stationary distribution π, and represents in fact a limitation of the formulation given in Tierney and Mira (1999) that may in general restrict efficiency, and completely prohibits its natural application in a variable dimension setting. If x and z lie in different spaces, symmetry seems to prevent y lying in either! To generalise the method, we adopt the kind of formulation used in Section 2.3 but since we will be considering more than one stage we add a subscript to both the random numbers u's and the deterministic functions h's indicating the stage in question.

3.1 Detailed balance at the second stage

Consider the Markov chain transition defined by the following construction of the first and second stage proposals. We use two vectors of random numbers u_1 and u_2, drawn from densities g_1 and g_2 respectively, and two deterministic functions mapping these and the current state into the proposed new states. The first proposal is $y = h_1^+(x, u_1)$ and the second is $z = h_2^+(x, u_1, u_2)$. In the pedagogical example we take $z = x + \sigma_2 u_2$ where σ_2 is the spread of the second stage proposal. Dependence of z on u_1 as well as u_2 allows this second-stage proposal to be influenced by the first stage proposed (and rejected) value y. In the pedagogical example we could for instance truncate the second stage proposal to the interval $x \pm |y - x|$.

The first stage proposal is accepted with probability $\alpha_1(x, y)$ calculated in the usual way, as in (9), with an appropriate change of notation:

$$\alpha_1(x, y) = \min\left\{1, \frac{\pi(y)g_1'(u_1')}{\pi(x)g_1(u_1)} \left|\frac{\partial(y, u_1')}{\partial(x, u_1)}\right|\right\},$$

where u_1' is such that $x = h_1^-(y, u_1')$.

Suppose that the move to y is rejected. We want to find the correct acceptance probability $\alpha_2(x, z)$ for the second-stage proposal z, without forcing the return path from z to x to go through y.

For Borel sets A, B, C in the state space $\mathcal{X}$, let $\widehat{\pi}(A, B, C)$ be the probability, in equilibrium, of beginning at a state in A, proposing and rejecting a first stage candidate in B and then proposing and accepting a second stage move to a state in C. Detailed balance at the second stage means

$$\widehat{\pi}(A, \mathcal{X}, C) = \widehat{\pi}(C, \mathcal{X}, A) \tag{10}$$

for all A, C. Now

$$\widehat{\pi}(A, B, C) = \int_{(x,y,z)\in A\times B\times C} dx\, du_1\, du_2\, \pi(x)g_1(u_1)g_2(u_2)[1 - \alpha_1(x, y)]\alpha_2(x, z) \tag{11}$$

and, similarly,

$$\widehat{\pi}(C, B^\star, A) = \int_{(z,y^\star,x)\in C\times B^\star\times A} dz\, d\widetilde{u_1}\, d\widetilde{u_2}\, \pi(z)\widetilde{g_1}(\widetilde{u_1})\widetilde{g_2}(\widetilde{u_2})[1-\alpha_1(z,y^\star)]\alpha_2(z,x), \tag{12}$$

where $y^\star = h_1^-(z,\widetilde{u_1})$ and $x = h_2^-(z,\widetilde{u_1},\widetilde{u_2})$.

Imposing the DB condition (10) using these integral forms is problematical. In any application of practical interest, it is unlikely that the integrals over $\mathcal{X}$ could be computed explicitly. One way to ensure the integral equality is to require it to hold with $B = B^\star$, for all Borel B (and all such A, C) – this is the method used in Tierney and Mira (1999), in different notation. More generally, however, we can augment the mappings already defined to establish a diffeomorphism from (x, u_1, u_2) to $(z, \widetilde{u_1}, \widetilde{u_2})$, and require the integrands to be equal after making the associated change of variable.

Suppose we have such a transformation with x, y, z and $y^\star$ defined as above. In this notation (12) becomes, after the change of variable:

$$\begin{aligned}\widehat{\pi}(C, B^\star, A) \;\; = \;\; & \int_{(z,y^\star,x)\in C\times B^\star\times A} dx\, du_1\, du_2 \pi(z)\widetilde{g_1}(\widetilde{u_1})\widetilde{g_2}(\widetilde{u_2}) \\ & [1-\alpha_1(z,y^\star)]\alpha_2(z,x)\left|\frac{\partial(z,\widetilde{u_1},\widetilde{u_2})}{\partial(x,u_1,u_2)}\right|. \end{aligned} \tag{13}$$

The DB condition (10) can be ensured by setting the integrands in (11) and (13) equal for all (x, u_1, u_2), i.e.

$$\begin{aligned} & \pi(x)g_1(u_1)g_2(u_2)[1-\alpha_1(x,y)]\alpha_2(x,z) \\ = \; & \pi(z)\widetilde{g_1}(\widetilde{u_1})\widetilde{g_2}(\widetilde{u_2})[1-\alpha_1(z,y^\star)]\alpha_2(z,x)\left|\frac{\partial(z,\widetilde{u_1},\widetilde{u_2})}{\partial(x,u_1,u_2)}\right|. \end{aligned}$$

Solving for α_2 we find that a valid choice for the required acceptance probability is

$$\alpha_2(x,z) = \min\left\{1, \frac{\pi(z)}{\pi(x)}\frac{\widetilde{g_1}(\widetilde{u_1})\widetilde{g_2}(\widetilde{u_2})}{g_1(u_1)g_2(u_2)}\frac{[1-\alpha_1(z,y^\star)]}{[1-\alpha_1(x,y)]}\left|\frac{\partial(z,\widetilde{u_1},\widetilde{u_2})}{\partial(x,u_1,u_2)}\right|\right\}. \tag{14}$$

This is our generalisation of (3). The similarity of form of equations (14) and (9) will be apparent. The term $\alpha_1(z, y^\star)$ needed here is obtained, similarly to $\alpha_1(x, y)$, by

$$\alpha_1(z,y^\star) = \min\left\{1, \frac{\pi(y^\star)}{\pi(z)}\frac{\widetilde{g_1}'(\widetilde{u_1}')}{\widetilde{g_1}(\widetilde{u_1})}\left|\frac{\partial(y^\star,\widetilde{u_1}')}{\partial(z,\widetilde{u_1})}\right|\right\}$$

where $\widetilde{u_1}'$ is such that $z = h_1^+(y^\star, \widetilde{u_1}')$.

3.2 An example with dimension fixed

Here are reported, in detail, the various 'ingredients' and the 'recipe' of the generalised version of the DRA for the pedagogical example introduced in Section 2.1.

Ingredients.

- proposal densities for u_1 and u_2: we take both g_1 and g_2 to be standard normal;
- a mapping from (x, u_1) to $y = h_1^+(x, u_1)$: we take $y = x + \sigma_1 u_1$;
- a mapping from (x, u_1, u_2) to $z = h_2^+(x, u_1, u_2)$: we take $z = x + \sigma_2 u_2$;
- reverse mappings h_1^- and h_2^-: we take $h_1^- = h_1^+$ and $h_2^- = h_2^+$;
- augmentation of these mappings so that $(x, u_1, u_2) \mapsto (z, \widetilde{u_1}, \widetilde{u_2})$ is a diffeomorphism, and satisfies both $z = h_2^+(x, u_1, u_2)$ and $x = h_2^-(z, \widetilde{u_1}, \widetilde{u_2})$: we take $\widetilde{u_1} = -u_1$ and $\widetilde{u_2} = -u_2$. The transformation, being linear, is obviously differentiable, and it is readily checked that its Jacobian is nonzero, so it is invertible:

$$\left|\frac{\partial(z, \widetilde{u_1}, \widetilde{u_2})}{\partial(x, u_1, u_2)}\right| = \begin{vmatrix} 1 & 0 & \sigma_2 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{vmatrix} = 1.$$

Note that the first and second stage proposals are simply normally-distributed 'random walk' perturbations with different variances. The choice that $\widetilde{u_1} = -u_1$ has been made merely to preserve reflective symmetry. A consequence is that the virtual first stage proposal for the reverse move is $y^\star = z + \sigma_1 \widetilde{u_1} = x + \sigma_2 u_2 - \sigma_1 u_1 = z - (y - x)$; see Figure 1.

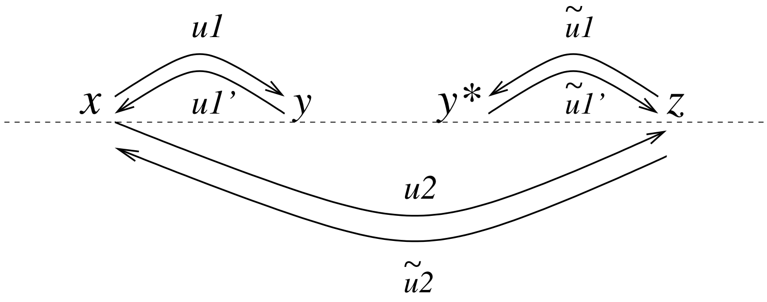

Figure 1: Toy example with fixed dimensions: variables in play.

A lot of freedom is left to the researcher in designing mappings alternative to these. The only requirement, as pointed out earlier, is that the resulting transformations are diffeomorphisms.

Recipe.

- Generate u_1 from g_1, and set $y = x + \sigma_1 u_1$.
- Find u_1', the solution to $x = h_1^-(y, u_1')$. In our example $u_1' = -u_1$. Evaluate the Jacobian:

$$\left|\frac{\partial(y, u_1')}{\partial(x, u_1)}\right| = 1.$$

- Compute the first-stage acceptance probability from (9):

$$\alpha_1(x,y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\frac{g_1'(u_1')}{g_1(u_1)}\left|\frac{\partial(y,u_1')}{\partial(x,u_1)}\right|\right\} = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}.$$

- Accept y with probability $\alpha_1(x,y)$.
- Generate u_2 from g_2, and set $z = h_2^+(x, \widetilde{u_1}, \widetilde{u_2}) = x + \sigma_2 u_2$.
- Set $\widetilde{u_1} = -u_1$ and calculate $y^\star = h_1^-(z, \widetilde{u_1}) = z + \sigma_1 \widetilde{u_1}$.
- Find $\widetilde{u_1}'$, the solution to $z = h_1^+(y^\star, \widetilde{u_1}')$. In our setting $\widetilde{u_1}' = -\widetilde{u_1} = +u_1$. Evaluate the Jacobian:

$$\left|\frac{\partial(y^\star, \widetilde{u_1}')}{\partial(z, \widetilde{u_1})}\right| = 1.$$

- Compute the virtual first-stage acceptance probability from (9):

$$\alpha_1(z,y^\star) = \min\left\{1, \frac{\pi(y^\star)}{\pi(z)}\frac{\widetilde{g_1}'(\widetilde{u_1}')}{\widetilde{g_1}(\widetilde{u_1})}\left|\frac{\partial(y^\star,\widetilde{u_1}')}{\partial(z,\widetilde{u_1})}\right|\right\} = \min\left\{1, \frac{\pi(y^\star)}{\pi(z)}\right\}.$$

- Set $\widetilde{u_2} = -u_2$, and evaluate the Jacobian:

$$\left|\frac{\partial(z, \widetilde{u_1}, \widetilde{u_2})}{\partial(x, u_1, u_2)}\right| = 1.$$

- Compute the second-stage acceptance probability from (14):

$$\begin{aligned}\alpha_2(x,z) &= \min\left\{1, \frac{\pi(z)}{\pi(x)}\frac{\widetilde{g_1}(\widetilde{u_1})\widetilde{g_2}(\widetilde{u_2})}{g_1(u_1)g_2(u_2)}\frac{[1-\alpha_1(z,y^\star)]}{[1-\alpha_1(x,y)]}\left|\frac{\partial(z,\widetilde{u_1},\widetilde{u_2})}{\partial(x,u_1,u_2)}\right|\right\} \\ &= \min\left\{1, \frac{\pi(z)}{\pi(x)}\frac{[1-\alpha_1(z,y^\star)]}{[1-\alpha_1(x,y)]}\right\}.\end{aligned}$$

- Accept z with probability $\alpha_2(x,z)$. Otherwise remain at x.

A performance comparison of the strategy described above with the more standard random walk Metropolis algorithm is presented in Section 6.1.

4 The variable dimension case

The second stage acceptance probability derived from (14) is quite general and applies without change to problems where the dimension of the objects considered is not fixed. In this setting we only need to take extra care to ensure that the transformations among the variables are diffeomorphisms; in practice, this is usually straightforward once the dimensions of the random variables involved in the simulation are properly matched.

4.1 Matching dimensions

Consider a general setting in which $x, y^\star \in \mathcal{R}^{d_1}$ and $y, z \in \mathcal{R}^{d_2}$. Without loss of generality assume that $d_1 < d_2$. As before, we set $y = h_1^+(x, u_1)$, $z = h_2^+(x, u_1, u_2)$ and $y^\star = h_1^-(z, \widetilde{u_1})$; the variables $u_1, u_2, u_1', \widetilde{u_1}, \widetilde{u_2}, \widetilde{u_1}'$ needed for the acceptance probabilities have dimensions $r_1, r_2, r_1', \widetilde{r_1}, \widetilde{r_2}, \widetilde{r_1}'$ respectively. To ensure invertibility we need three equalities to match the dimensions of the moves between subspaces. To balance the first stage move from x to y we need (as in the standard reversible jump framework) that $d_1 + r_1 = d_2 + r_1'$. The second constraint balances the dimensions for the z to $y^\star$ move: $d_2 + \widetilde{r_1} = d_1 + \widetilde{r_1}'$. The last constraint is needed to balance globally the second stage move from x to z: $d_1 + r_1 + r_2 = d_2 + \widetilde{r_1} + \widetilde{r_2}$. Figure 2 displays the variables in play and their relationships.

Commonly, dimension-jumping moves can be set up so that in jumping *down* dimensions, moves are deterministic. Thus, supposing that $d_1 < d_2$, we have $r_1' = \widetilde{r_1} = 0$. Then it follows that $r_1 = \widetilde{r_1}' = d_2 - d_1$, and $r_2 = \widetilde{r_2}$.

Our framework does not actually require the dimensions of the first and second stage proposals to be the same. Relaxing this might be useful: if a proposal to move to a state in a higher dimensional space is rejected, it may be appropriate to try a second stage proposal that is more 'timid' in not increasing dimension so much. Thus, with d_x denoting the dimension of x, etc., we might have $d_{y^\star} < d_x < d_z < d_y$. For symmetry, suppose $d_y - d_x = d_z - d_{y^\star} = b$ (for 'bold') and $d_z - d_x = t$ (for 'timid'), with $t < b$; then the dimensions are matched correctly if we take $r_1' = \widetilde{r_1} = 0$ again, $r_1 = \widetilde{r_1}' = b$ and $\widetilde{r_2} - r_2 = b - t$. Counter-intuitively, the second stage down-dimension move requires more random numbers than its up-dimension counterpart.

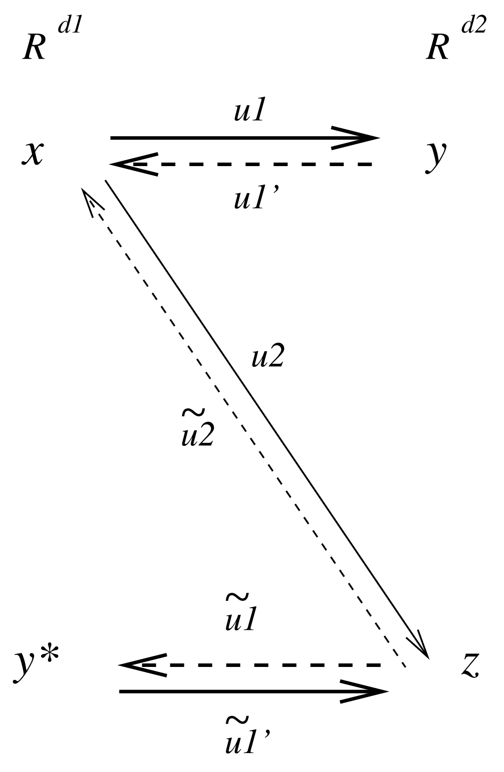

Figure 2: Relationships among variables for DRA in the variable dimension case.

4.2 A toy example with variable dimension

Consider a target distribution given by a mixture of two components; the first is one dimensional, a density on the unit interval, and the second is two dimensional, a density on the triangular region $y_1 \geq 0$, $y_2 \geq 0$, $y_1 + y_2 \leq 1$. The forms of the two densities do not matter in the following discussion.

When we attempt to jump from the one to the two dimensional state space we have $x, y^\star \in \mathcal{R}$ and $y, z \in \mathcal{R}^2$ so that $d_1 = 1$ and $d_2 = 2$. Valid values for the dimensions of the u variables, those used here, are $r_1 = \widetilde{r_1}' = r_2 = \widetilde{r_2} = 1$, $r_1' = \widetilde{r_1} = 0$.

To save space, we summarise the moves we implemented for this setting in the form of a table, Table 1. This gives one possible way to move between these two spaces making sure that dimension matching is properly done and that all transformations are diffeomorphisms. The notation used is such that the move from dimension 2 to dimension 1 is expressed in terms of a move from z, attempting first to move to $y^\star$, then x, with now y as the virtual reverse first stage proposal. This perhaps makes it easier to appreciate how the forward and reverse moves correspond. The various assignments and random number generations should be executed in the order of the lines in the table. Both g_1 and g_2 are densities on the unit interval.

To understand the idea in this setting, it may be helpful to imagine a graphical depiction of the two spaces, in which the two dimensional space is the triangular region described above and the hypotenuse is identified with the one dimensional space, with $(1-x, x)$ mapping to x. Then the first and second stage proposals correspond to horizontal and vertical perturbations: those from $\mathcal{R}^2$ to $\mathcal{R}$ being multiplications by the random numbers generated from g_1 and g_2, the reverse moves being simply projections onto the hypotenuse, parallel to the axes.

Table 1: Delayed rejection algorithm for the toy example with variable dimension.

	$1 \to 2$			$2 \to 1$	
variable	dimension	formula	variable	dimension	formula
x	1	given	z	2	given
u_1	1	$\sim g_1$	$\widetilde{u_1}$	0	null
y	2	$((1-x)u_1, x)$	$y^\star$	1	z_2
u_1'	0	null	$\widetilde{u_1}'$	1	$z_1/(1-z_2)$
u_2	1	$\sim g_2$	$\widetilde{u_2}$	1	$\sim \widetilde{g_2}$
z	2	$(1-x, xu_2)$	x	1	$1-z_1$
$\widetilde{u_1}$	0	null	u_1	1	$\widetilde{u_2}$
$y^\star$	1	z_2	y	2	$((1-x)u_1, x)$
$\widetilde{u_1}'$	1	$(1-x)/(1-xu_2)$	u_1'	0	null
$\widetilde{u_2}$	1	u_1	u_2	1	$z_2/(1-z_1)$

Note that in the $1 \to 2$ column in Table 1, $\widetilde{u_2}$ is set $(= u_1)$, apparently redundantly, since it is not used to obtain any of the variables involved in the simulation. On the other hand, in the $2 \to 1$ column, $\widetilde{u_2}$ is generated and not used directly. However u_1 is set to $\widetilde{u_2}$ and that is used to propose the value of y. The $u_1 \mapsto \widetilde{u_2}$ mapping used here does not need to be the identity, but must be such that the $(x, u_1, u_2) \mapsto (z, \widetilde{u_1}, \widetilde{u_2})$ mapping is a diffeomorphism. Of course, in *both* directions, $\widetilde{u_2}$ is used to compute the second stage acceptance probability.

This rather elliptical use and identification of the u variables is typical of variable-dimension delayed rejection.

The Jacobians needed to calculate the acceptance probabilities are given in Table 2.

Table 2: Jacobians for the toy example with variable dimension.

Jacobians	$1 \to 2$	$2 \to 1$
$\left\lvert \frac{\partial(y, u_1')}{\partial(x, u_1)} \right\rvert$	$1-x$	$1/(1-x)$
$\left\lvert \frac{\partial(y^\star, \widetilde{u_1}')}{\partial(z, \widetilde{u_1})} \right\rvert$	$1/(1-y^\star)$	$1-y^\star$
$\left\lvert \frac{\partial(z, \widetilde{u_1}, \widetilde{u_2})}{\partial(x, u_1, u_2)} \right\rvert$	x	$1/x$

5 Application to change point analysis

5.1 Model and MCMC sampler

As a more significant application of the theory developed in this paper consider a change point problem. We will follow the setting and the notation in Green (1995, Section 4). Our implementation of delayed rejection for this problem includes a new ingredient, that will be more generally useful: state-dependent random choice of 'sub-moves'.

The observations are a point process of events at times $y_1, y_2, \ldots, y_N$ in an interval $[0, L)$. We assume the events occur at random, that is, as a Poisson process, but possibly not uniformly. The objective is to make inference about the rate $x(t)$ of the Poisson process, which we model as a step function. We allow an unknown number of change points (steps), with random locations and heights. Here we add two additional features to the model in Green (1995): unknown hyperparameters for the prior distribution of heights, and a parametric periodic component.

The sampler needs to move between points in spaces of different dimensions (that is, step functions with a different number of steps) and this is achieved by designing birth and death moves. We leave discussion of the other moves to the next subsection.

Consider a birth move that proposes a new step, when there are currently k steps. We have $d_1 = 2k+1$ and $d_2 = 2k+3$, in the notation of Section 4.1. One choice that preserves the dimension balancing requirement is $r_1 = r_2 = \widetilde{r_2} = \widetilde{r_1}' = 2$ and $r_1' = \widetilde{r_1} = 0$ and this is what we adopt. We take all the u variables used for the simulation to be independent uniforms on the interval $[0, 1]$. The way these are used and re-used in the forward and reverse moves is precisely like that of the example in the previous section, except that here they come in pairs rather than singly. Each time a birth is proposed, one of the $r_1 = r_2 = 2$ uniform random variables generated is used to choose the position of the step. The second variable is used to split the current height into two new heights, preserving, as in Green (1995), the integral of the logarithm of the step function. Conversely, in the death move, a step is deleted, and the step function heights to either side of the deleted step are combined, deterministically, to preserve the same integral.

We now need to introduce some notation to allow us to describe these proposals more precisely. Let x denote a step function on $[0, L)$ with k steps, at positions $s_1^x < s_2^x < \cdots < s_k^x$, and taking the value h_j^x on the interval $[s_j^x, s_{j+1}^x), j = 0, 1, \ldots, k$. We use the conventions that $s_0^x = 0$ and $s_{k+1}^x = L$. Similarly, y and z are step functions with $(k+1)$ steps, and $y^\star$ one with k steps; their step positions and heights are denoted with corresponding superscripts.

We suppose y is obtained from x by adding a step in the j_1^{th} subinterval $[s_{j_1-1}^x, s_{j_1}^x)$, for some $j_1 = 1, 2, \ldots, k+1$. Similarly, z is obtained from x by adding a step in the j_2^{th} subinterval, $j_2 \neq j_1$. Turning now to the death moves required to design the reverse path, we obtain $y^\star$ and x from z by deleting respectively the j_1^{th} and j_2^{th} steps.

Given the auxiliary random variables (u_1^s, u_1^h) and the choice of sub-interval j_1 for the new step, the following algebra defines the proposed updated step function y:

$$\begin{aligned}
s_j^y &= s_j^x \quad \text{for } j = 0, 1, \ldots, j_1 - 1 \\
s_{j_1}^y &= s_{j_1-1}^x + u_1^s(s_{j_1}^x - s_{j_1-1}^x) \\
s_j^y &= s_{j-1}^x \quad \text{for } j = j_1 + 1, \ldots, k+2 \\
h_j^y &= h_j^x \quad \text{for } j = 0, 1, \ldots, j_1 - 2 \\
h_{j_1-1}^y &= h_{j_1-1}^x \left(\frac{u_1^h}{1 - u_1^h} \right)^{(1-u_1^s)} \\
h_{j_1}^y &= h_{j_1-1}^x \left(\frac{1 - u_1^h}{u_1^h} \right)^{u_1^s} \\
h_j^y &= h_{j-1}^x \quad \text{for } j = j_1 + 1, \ldots, k+1.
\end{aligned}$$

Expressions for $\{s_j^z\}$ and $\{h_j^z\}$ are obtained in the same way, but using (j_2, u_2^s, u_2^h) in place of (j_1, u_1^s, u_1^h). The virtual state $y^\star$ is specified by the $\{s_j^{y^\star}\}$ and $\{h_j^{y^\star}\}$ that are defined by solving the above equations for $\{s_j^x\}$ and $\{h_j^x\}$, but starting from z instead of y.

These moves are, individually, identical to those used in Green (1995, Section 4). As there, the probabilities of choosing birth or death moves, when there are currently k steps, are $b(k)$ and $d(k)$ respectively. The only modifications here concern how to choose j_1 and j_2, that is, in which subinterval the new step is to go (in the birth move) or which step is to be deleted (in the death move) at both the first and second stages of the delayed rejection procedure. There is an almost unlimited range of possibilities here; we have not explored many options, and certainly do not claim optimality for the one we choose to implement for illustration. We make these choices at random and the probabilities of our choices appear as additional terms in the acceptance ratios.

For a birth, at the first stage, an interval between steps is chosen at random, with probability proportional to the length of that interval; at the second stage, the same procedure is followed, but conditional on a different between-step interval being chosen. Thus, at x, we choose j_1 with probability $b_1(j_1; x) = (s_{j_1}^x - s_{j_1-1}^x)/L$, and if y is not accepted, draw j_2 with probability $b_2(j_2; j_1, x) = (s_{j_2}^x - s_{j_2-1}^x)/(L - (s_{j_1}^x - s_{j_1-1}^x))$ for $j_2 \neq j_1$. For a death, at the first stage, a step is chosen at random to be deleted, with equal probabilities; at the second stage, a random choice is made among the other available steps. Thus at z, the distributions of j_1 and j_2 are respectively $d_1(j_1; z) = 1/(k+1)$, $j_1 = 1, 2, \ldots, k+1$ and $d_2(j_2; j_1, z) = 1/k$, $j_2 = 1, 2, \ldots, k+1$, but $j_2 \neq j_1$.

In both birth and death, a rudimentary form of learning is being applied: the second attempt is of the same kind as the first, but at a different location.

Given all these sub-move probabilities, the additional factors needed in the Metropolis-Hastings ratios are, for $\alpha_1(x,y)$: $d_1(j_1;y)/b_1(j_1;x)$, for $\alpha_1(z,y^\star)$: $b_1(j_1;y^\star)/d_1(j_1;z)$, and finally for $\alpha_2(x,z)$: $[d_1(j_1;z)d_2(j_2;j_1,z)]/[b_1(j_1;x)b_2(j_2;j_1,x)]$.

The Jacobian for the first stage birth move can be written as

$$\frac{(s^x_{j_1} - s^x_{j_1-1})h^y_{j_1-1}h^y_{j_1}}{u^h_1(1-u^h_1)h^x_{j_1-1}}$$

and the others are of similar form.

To give just one example in full detail, here is the complete expression for the second-stage acceptance probability for a birth move:

$$\begin{aligned}\alpha_2(x,z) &= \min\left\{1, (\text{prior ratio}) \times (\text{likelihood ratio}) \times \frac{g(\widetilde{u_2})}{g(u_1)g(u_2)} \times \frac{(1-\alpha_1(z,y^\star))}{(1-\alpha_1(x,y))}\right.\\ &\times \left.\frac{d(k+1)}{b(k)}\frac{d_1(j_1;z)d_2(j_2;j_1,z)}{b_1(j_1;x)b_2(j_2;j_1,x)} \times \frac{(s^x_{j_2} - s^x_{j_2-1})h^z_{j_2-1}h^z_{j_2}}{u^h_2(1-u^h_2)h^x_{j_2-1}}\right\}.\end{aligned}$$

To save space, we do not spell out the forms of the prior and likelihood ratios, which are rather obvious in structure.

We also add a second stage to the change of height move. The second stage move is identical to the first stage except that a different step is chosen.

5.2 Cyclones in the Bay of Bengal

We will apply the MCMC methodology we have developed to a data set on the times of cyclones hitting the Bay of Bengal between 1877 and 1977 inclusive: over a period of 101 years, there were 141 cyclones (or, to be precise, severe cyclonic storms, that is, storms with winds exceeding 88 km h^{-1}). The data, which are taken from Mooley (1981), are plotted in Figure 3; there is more detail on their reversible jump analysis in Green (2001).

The model is generally as described in Green (1995). An additional model component here is a simple sinusoid representing seasonal effects, fixed to have a six month period and a uniformly distributed phase, that multiplies the step function term whose modelling we have described. (The period is very clear from an exploratory data analysis, and arises from the meteorological fact that the incidence of cyclones peaks twice a year in the Bay of Bengal, due to the close proximity to the equator.) The additional MCMC move to deal with this is a simple random walk Metropolis move on the amplitudes of the sine and cosine component.

Finally, we place hyperpriors on the parameters (α,β) of the height distribution, and use straightforward random walk Metropolis and Gibbs updates for these. Other MCMC moves are as in Green (1995). In all, there are six move types: the birth/death move, and the moves updating a step height, a step position, α, β, and the harmonic component. In contrast to Green (1995), we choose a random scan over these types, with probabilities proportional to $0.5, 0.7, 0.2, 0.15, 0.15, 0.02$ respectively, instead of a fixed cyclic scan. This is because it is easily proved that, when a random scan is used, improving the kernels of individual moves in the Peskun sense improves the overall kernel, but there is no guarantee of this with a fixed scan.

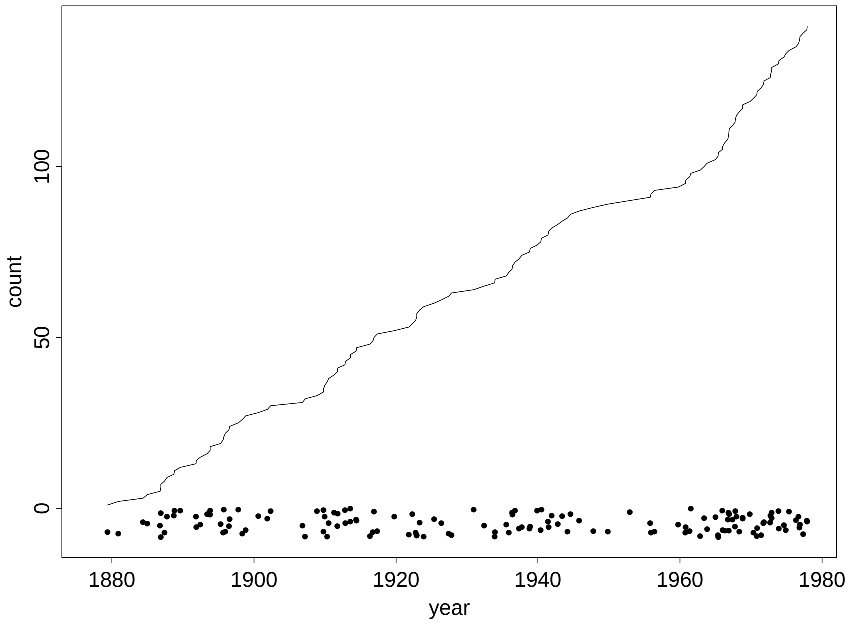

Figure 3: Cyclones data shown as a jittered dot-plot and as cumulative counting process.

We use the following priors for illustration. The prior on the number of steps k is taken to be Poisson, with mean $\lambda = 3$. The step heights $\{h_j\}$ are taken to be independently Gamma(α, β) distributed (that is, with mean α/β and variance α/β^2), where $\alpha \sim \text{Gamma}(2, 2)$ and $\beta \sim \text{Gamma}(1, N/L)$.

Figure 4 shows the estimated posterior distribution for k, the number of change points, based on a run of 1 000 000 sweeps of the DRA. Thus, according to this analysis, there are, most probably, three change points in the observed data set, although there is substantial posterior variability: a 90% credible interval covers [2,6]. Note that only very limited learning about k is achieved from these data. In Figure 5 are shown kernel density estimates of the posterior densities for the positions of the change points, conditional on there being 2 or 3 change points in the interval. We interpret this as showing there is rather strong evidence of change points around years 1885, 1945 and 1961. In the case $k = 2$, the posterior probability for the first change point is split between the first two of these modes. As is often the case with variable-dimension Bayesian inference, multiple explanations for the data are being uncovered.

These and other aspects of posterior inference are essentially numerically identical to those obtained by the standard reversible jump method, of course. The difference is that, as we see below, the estimates obtained using the DRA have smaller asymptotic variance for the same computing time.

6 Performance comparisons

In general, comparing the performance of MCMC samplers is not an easy task. Here, we define 'performance' as the efficiency of estimating the expectation $E_\pi \phi$ of a given function ϕ

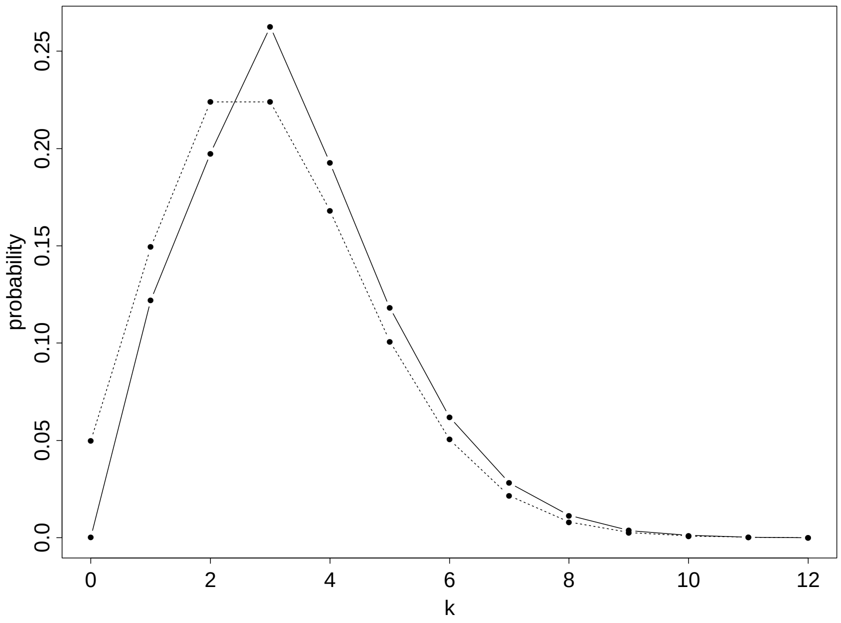

Figure 4: Cyclones data; prior (dotted) and estimated posterior distribution of k.

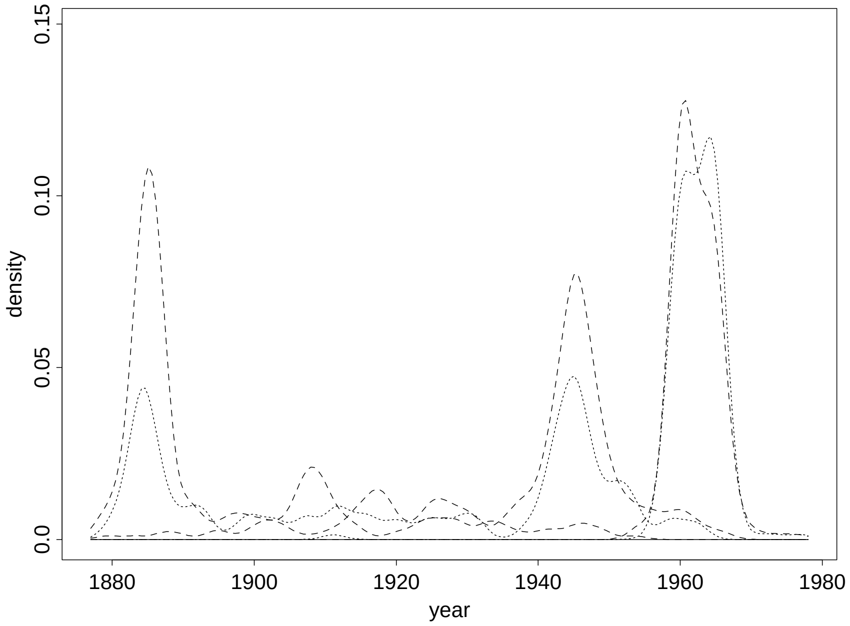

Figure 5: Cyclones data; estimated posterior densities for the positions of the change points, conditional on $k = 2$ (dotted lines) and 3 (dashed).

on the state space. There are two quantities to take into account: the running time T needed to obtain a fixed number of sweeps n, and the integrated autocorrelation time, $\tau = \sum_{k=-\infty}^{\infty} \rho_k$ where $\rho_k = \mathrm{Cov}_P[\phi(X_0), \phi(X_k)]/\sigma^2$ and σ^2 is the variance of ϕ under π (assumed to be finite). The product of τ and T gives a reasonable measure of the efficiency of an MCMC algorithm – it estimates the running time to obtain the same precision in estimating $E_\pi \phi$ as from n independent draws from π. The problem is how to measure T and estimate τ.

In order to measure T we need, in theory, the best possible code to implement the two Markov chains under comparison. Here, the DRA is being compared with a one-stage algorithm, which actually forms the first stage of the DRA; then, the DRA code is a superset of the other, and it is reasonable to believe the two codes are similarly efficient. This makes things a little easier, but still a definitive comparison remains elusive.

To estimate τ we used Sokal's adaptive truncated periodogram estimator (Sokal, 1989), $\sum_{|k| \leq M} \widehat{\rho}_k$ with the window width M chosen adaptively as the minimum integer with $M \geq 3\widehat{\tau}$. We used the fast Fourier transform to estimate the autocorrelations $\widehat{\rho}_k$ from the last 32 768 sweeps of our runs.

6.1 Pedagogical example with fixed dimension

Consider the target distribution introduced in Section 2.1. In our simulations we take the model parameters to be $\delta_1 = 1$ and $\delta_2 = 25$: with this choice the shape of the target is such that random walk Metropolis with a badly calibrated proposal is very slowly mixing. A smaller ratio δ_1/δ_2 would favour the DRA even more in the following performance comparison. The function of interest is $\phi(x) = \cos(7x)$, chosen since estimating the expectation of such a rapidly oscillating function is sensitive to the choice of proposal spread.

We will compare the performance of the DRA as described in Section 3.2, with that of standard random walk Metropolis with $y = h_1^+(x, u_1)$ as the proposal. Of course if we spent a lot of time and energy in many pilot runs (and, to be fair, this time should also be taken into account when comparing strategies) we can find a good value for the spread of the proposal (σ_1) in random walk Metropolis. And if we do so, then there is not much space left for the delaying rejection mechanism to improve the performance of the sampler. But one of the points of the delaying rejection strategy is exactly to avoid spending too much time in pilot runs.

For the sake of comparison we have indeed run some pilot experiments that lead us to believe that the optimal random walk Metropolis spread for a single stage proposal is approximately $\sigma_1 = 0.4$. This value is very close to half of the period of the function of interest, namely $\pi/7$. Heuristically, by proposing jumps of this size we are close to maximising $E[\phi(X_t) - \phi(X_{t+1})]^2$, the mean square change in value of ϕ in one sweep. We note that for different functions ϕ the optimal spread changes dramatically. Using this spread we get an acceptance rate of 84% and $\widehat{\tau}$ is 1.76 (s.d. 0.21, running time 27.4 sec.).

Given such a high Metropolis acceptance rate there is not much gain is adding a second stage to the simulation. Nevertheless by adding a second stage proposal with $\sigma_2 = \sigma_1$ the estimated value of τ goes down to 1.52 (s.d. 0.07, running time 31.7 sec.) and the second stage acceptance rate is approximately 4%. This comparison (based on 10 runs of length 10 000 000 each) shows that, even if we have a good guess of the spread of the first stage proposal, by adding a second stage with the same spread we can still improve the performance in terms of smaller τ and more stable estimates. Furthermore we do not do any worse even taking the extra computational time into account.

Let us now compare the performance of regular random walk Metropolis with DRA with various values for σ_1 and σ_2. Table 3 shows our simulation results obtained on a Pentium II, 333 MHz, under the Linux operating system with code written in Fortran. The figures reported above are obtained by averaging 10 replicates of 1 000 000. Since the sampler is started in stationarity no burn-in is required. In the columns named '1st % accept.' and '2nd % accept.' we report the first and second stage acceptance rates respectively. The running time of the DRA is higher in all cases as expected, but even taking this into account the DRA performs better, sometimes much better, than the non-delayed version of the algorithm. Our experiments show that overestimating the spread of the proposal is better than underestimating it. This is true both for the first and the second stage proposal, and extends the scope of the observation of Roberts, Gelman and Gilks (1997) for the random walk Metropolis algorithm with a normal proposal.

Table 3: Comparing performance in the pedagogical example: $n = 1\,000\,000$.

$\delta_1 = 1; \delta_2 = 25$		no delay				delay			
σ_1	σ_2	T in sec.	$\hat{\tau}$	$T\times\hat{\tau}$	1st % accept.	T in sec.	$\hat{\tau}$	$T\times\hat{\tau}$	2nd % accept.
0.2	0.05	3.40	3.06	10.40	92.0	3.52	2.94	10.35	0.08
	0.1					3.53	2.95	10.41	0.1
	0.15					3.53	2.90	10.24	0.13
	0.3					3.56	2.88	10.25	0.35
	0.4					3.65	2.82	10.29	0.51
0.4	0.1	3.42	1.74	5.95	84.1	3.77	1.63	6.14	3.8
	0.2					3.76	1.59	5.98	3.9
	0.3					3.81	1.56	5.94	4.0
	0.6					3.82	1.59	6.07	3.6
	0.8					3.81	1.54	5.87	3.0
0.6	0.15	3.42	1.98	6.77	77.2	3.90	1.68	6.55	10.2
	0.3					3.90	1.50	5.85	9.6
	0.45					4.04	1.51	6.10	8.6
	0.9					4.07	1.60	6.51	5.7
	1.2					4.03	1.64	6.61	4.5
1	0.25	3.43	2.74	9.40	68.1	4.27	1.68	7.17	18.7
	0.5					4.29	1.62	6.95	14.8
	0.75					4.30	1.76	7.57	11.4
	1.5					4.35	2.05	8.92	6.4
	2.25					4.35	2.22	9.66	4.9
3	0.75	3.42	6.84	23.40	53.5	4.75	2.10	9.97	18.7
	1.5					4.79	2.98	14.27	10.5
	2.25					4.80	3.65	17.52	7.2
	4.5					4.83	4.52	21.83	3.7
	6.0					4.81	5.14	24.72	2.8

6.2 Change points example

To test the performance of the DRA for the change-point application, we considered two functions of interest, $\phi_1(x)$ = 'position of the first change point' and $\phi_2(x) = k$ = 'number of change points'. We ran the simulation for $n = 1\,000\,000$ sweeps with a burn-in of length 1 000. On average, at the first stage, the sampler accepts 41.3% of height change moves, and 25.1% of births and 21.8% of deaths (the difference in the last two rates is because b_k and d_k are unequal). Approximately 14.4% of the birth, 11.9% of the death, and 41.5% of the height moves rejected at the first stage are accepted at the second stage. Table 4 shows that for both functions of interest the DRA performs better than the single-stage method; the figures reported are obtained by averaging over 20 independent runs. The improvement in the performance is not as great as in many of the runs of the toy example but we believe that it is still worth the effort of setting up the DRA. As shown in Table 4, the degree of improvement will depend on the function whose expectation we want to estimate. The value in the column named 'ratio' is the ratio of T times $\widehat{\tau}$ for the standard reversible jump method and the DRA. Thus values greater than one indicate how much more efficient the delaying rejection strategy is. We know that by the Peskun-Tierney theorem (Peskun, 1973; Tierney, 1998) the DRA will always give smaller values of τ than a regular Metropolis-Hastings algorithm, for every function of interest ϕ. How much smaller the integrated autocorrelation time is depends on the function ϕ. For some ϕ the gain can more than compensate for the extra computational time needed for delayed rejection.

Table 4: Change points example: $n = 1\,000\,000$.

	no delay		delay		ratio
	T in sec.	$\widehat{\tau}$	T in sec.	$\widehat{\tau}$	
ϕ_1	29.2	42.17	31.5	30.54	1.28
ϕ_2		70.33		48.51	1.34

7 Discussion

In all of the examples presented, we attempt only two proposals before conceding rejection and advancing time. In our limited experience going on to third or higher stages requires an amount of computation and programming that is not worthwhile. The 'delayed' Markov chains are so much slowed down that the gain in efficiency on a sweep by sweep basis is not retained when the comparisons are made in terms of $T \times \widehat{\tau}$.

The experience on which we base this is entirely with problems where the natural implementation of DRA gives a workload increasing geometrically with the number of attempts, because of the proliferation of virtual states that need to be considered. To understand this point, we need to abandon the $\{x, y, z, y^\star\}$ notation that served for the states considered in the two-stage algorithm. We now name these $\{x, x_a, x_b, x_{ba}\}$, where the a and b subscripts denote the types of move proposed at the first and second stages, and the subscripts are listed in the same order as the moves are applied. For a three stage algorithm, we need to consider eight states: $\{x, x_a, x_b, x_{ba}, x_c, x_{ca}, x_{cb}, x_{cba}\}$ in order to be able to evaluate the three acceptance probabilities that are needed. In general, for k stages, 2^k states are involved. In some settings, however, it is plausible that these states need not all be different, and a workload linear in k may be obtained. Indeed, in the fixed dimension setting and for their special case of delayed rejection, Tierney and Mira (1999) demonstrate an implementation with four stages (although the running times do not come into the performance comparisons there).

We have focused in this paper on comparative performance as measured by estimated asymptotic variance, since that is the measure that DRA is guaranteed to improve, by Peskun's theorem, before allowing for additional computing time. As we have seen empirically, the computational advantage of DRA depends on which function of the Markov chain is being estimated. Given a pair of Markov kernels P_i, $i = 1, 2$, both irreducible and reversible with respect to the same invariant distribution π, it is reasonable to ask for what ϕ is it most advantageous, in terms of asymptotic variance, to use P_2 rather than P_1 to estimate $E_\pi \phi$ by MCMC. Since each asymptotic variance is an inner product of the form $\langle \phi, R_i \phi \rangle_\pi$, the 'optimal' ϕ is the solution to a generalised eigenvalue problem. For finite state spaces, such problems can be solved by numerical linear algebra routines, and some limited calculations support the intuition that the 'bold-then-timid' strategy is particularly useful for rapidly-varying ϕ.

In our illustrations of delayed rejection in this paper, we have not been very inventive in designing second-stage proposals, and have achieved correspondingly modest improvements in performance. We believe that more substantial performance improvements are possible, for any Metropolis-Hastings and reversible jump method, if a clever way of updating the proposal distribution upon a rejection can be devised. General advice towards this is to try 'bold' moves at early stages and 'timid' moves later on. Here 'bold' and 'timid' have meanings related to the specific application, and might involve the scale of a move or the number of dimensions jumped.

For example, in the change-point application in Section 5, a more timid birth could be obtained by favouring smaller differences between the heights of the new steps (perhaps by changing the distribution of u_1^h to be more concentrated around 0.5). Similarly, if a death move is rejected we could then try a move that leaves the number of steps unchanged but makes the heights of the steps we were trying to merge less different.

In a very different context, when running Metropolis-coupled MCMC (Geyer, 1991) on several parallel chains for targets at different temperatures, boldness and timidity could refer to the difference in the temperatures of the two chains we are trying to switch. There is scope for much further exploration of these ideas.

Acknowledgements

We are grateful to Neil Shephard for a question that led to the idea of being bold or timid with respect to dimension, mentioned in Section 5.1. We thank the editors and the referees for their careful review of the first version of this paper; their comments have helped us to remove a few errors and improve our notation and presentation. Antonietta Mira thanks the Department of Computer Science and Systems of the University of Pavia for use of computers and other facilities.

References

Geyer, C. J. (1991) Markov chain Monte Carlo maximum likelihood in *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, pp. 156–163. Ed. E. M. Keramidas. Interface Foundation, Fairfax Station.

Green, P. J. (1995) Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, **82**, 711–732.

Green, P. J. (2001) A primer on Markov chain Monte Carlo, pp. 1–62 of *Complex Stochastic Systems*, Barndorff-Nielsen, O. E., Cox, D. R. and Klüppelberg, C. (eds.), Chapman and Hall, London.

Hastings, W. K. (1970) Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57**, 97–109.

Mira, A. (1998) *Ordering, Slicing and Splitting Monte Carlo Markov Chains.* PhD thesis, University of Minnesota.

Mooley, D.A. (1981). Applicability of the Poisson probability model to the severe cyclonic storms striking the coast around the Bay of Bengal. *Sankhya*, **43** B, 187–197.

Peskun, P. H. (1973) Optimum Monte Carlo sampling using Markov chains. *Biometrika*, **60**, 607–612.

Roberts, G. O., Gelman, A. and Gilks, W. R. (1997) Weak convergence and optimal scaling of random walk Metropolis algorithms. *Annals of Applied Probability*, **7**, 110–120.

Sokal, A. D. (1989) *Monte Carlo methods in statistical mechanics: foundations and new algorithms.* Cours de Troisième Cycle de la Physique en Suisse Romande, Lausanne.

Tierney, L. (1998) A note on Metropolis-Hastings kernels for general state spaces. *Annals of Applied Probability*, **8**, 1–9.

Tierney, L. and Mira, A. (1999) Some adaptive Monte Carlo methods for Bayesian inference. *Statistics in Medicine*, **18**, 2507–2515.