Biometrika Advance Access published October 12, 2009

Biometrika (2009), *pp.* 1–8
doi: 10.1093/biomet/asp052

Printed in Great Britain

Adaptive approximate Bayesian computation

BY MARK A. BEAUMONT

School of Biological Sciences, University of Reading, PO Box 68, Whiteknights,
Reading RG6 6BX, U.K.
m.a.beaumont@reading.ac.uk

JEAN-MARIE CORNUET

Department of Epidemiology and Public Health, Imperial College, London SW7 2AZ, U.K.
jmcornuet@ensam.inra.fr

JEAN-MICHEL MARIN

Institut de Mathématiques et Modélisation de Montpellier, Université Montpellier 2,
Case Courrier 51, 34095 Montpellier cedex 5, France
Jean-Michel.Marin@univ-montp2.fr

CHRISTIAN P. ROBERT

Centre de Recherche en Mathématiques de la Décision, Université Paris Dauphine,
75775 Paris cedex 16, France
xian@ceremade.dauphine.fr

SUMMARY

Sequential techniques can enhance the efficiency of the approximate Bayesian computation algorithm, as in Sisson et al.'s (2007) partial rejection control version. While this method is based upon the theoretical works of Del Moral et al. (2006), the application to approximate Bayesian computation results in a bias in the approximation to the posterior. An alternative version based on genuine importance sampling arguments bypasses this difficulty, in connection with the population Monte Carlo method of Cappé et al. (2004), and it includes an automatic scaling of the forward kernel. When applied to a population genetics example, it compares favourably with two other versions of the approximate algorithm.

Some key words: Importance sampling; Markov chain Monte Carlo; Partial rejection control; Sequential Monte Carlo.

1. INTRODUCTION

When the likelihood function is not available in a closed form, as in population genetics, Pritchard et al. (1999) introduced approximate Bayesian computational methods as a rejection technique bypassing the computation of the likelihood function via a simulation from the corresponding distribution. Namely, if we observe $y \sim f(y \mid \theta)$ and if $\pi(\theta)$ is the prior distribution on the parameter θ, then the original approximate Bayesian computation algorithm jointly simulates $\theta' \sim \pi(\theta)$ and $x \sim f(x \mid \theta')$, and accepts the simulated θ' if and only if the auxiliary variable x is equal to the observed value, $x = y$. This algorithm is exact in that the accepted θs are distributed from the posterior. In the more standard case, where y is a continuous random variable, the algorithm relies on an approximation, where the equality $x = y$ is replaced by a tolerance condition, $\varrho(x, y) \leqslant \epsilon$, ϱ being a measure of discrepancy, for instance a distance between summary statistics, and $\epsilon > 0$ being the tolerance bound. The output is then distributed from the distribution with density proportional to $\pi(\theta)\,\mathrm{pr}_\theta\{\varrho(x,y)<\epsilon\}$, where pr_θ represents the probability distribution of x indexed by a value of θ. This density is denoted by $\pi\{\theta \mid \varrho(x,y)<\epsilon\}$, where the conditioning corresponds to the marginal distribution of $\varrho(x,y)$ given y. Improvements to this general scheme have been achieved either by modifying the proposal distribution of the parameter θ to increase the density of xs within a neighbourhood of y (Marjoram et al., 2003; Bortot et al., 2007) or by viewing the problem as a conditional density estimation and developing techniques to allow for larger ϵ (Beaumont et al., 2002).

Marjoram et al. (2003) defined a Markov chain Monte Carlo version of the approximate Bayesian computation algorithm that enjoys the same validity as the original algorithm, namely, if a Markov chain $(\theta^{(t)})$ is created via the transition that sets $\theta^{(t+1)}$ equal to a new value $\theta' \sim K(\theta' \mid \theta^{(t)})$ only if $x \sim f(x \mid \theta')$ is such that $x = y$ and if $u \sim \mathcal{U}(0,1) \leqslant \pi(\theta')K(\theta^{(t)} \mid \theta')\big/\pi(\theta^{(t)})K(\theta' \mid \theta^{(t)})$, then its stationary distribution is the posterior distribution $\pi(\theta \mid y)$. Since, in most settings, the distribution of y is absolutely continuous, the above constraint $x = y$ is replaced by the approximation $\varrho(x,y)<\epsilon$.

Example 1. For the toy model studied in Sisson et al. (2007), where $\theta \sim \mathcal{U}(-10,10)$ and $x \mid \theta \sim 0{\cdot}5\,\mathcal{N}(\theta,1)+0{\cdot}5\,\mathcal{N}(\theta,1/100)$, the posterior distribution associated with the observation $y=0$ is the normal mixture $\theta \mid y=0 \sim 0{\cdot}5\,\mathcal{N}(0,1)+0{\cdot}5\,\mathcal{N}(0,1/100)$ restricted to the set $[-10,10]$. As in regular Markov chain Monte Carlo settings, the performance of Marjoram et al.'s (2003) algorithm depends on the choice of the scale τ in the random walk proposal, $K(\theta' \mid \theta)=\tau^{-1}\varphi\{\tau^{-1}(\theta-\theta')\}$, where φ is often a standardized normal or a t density. Even when $\tau = 0{\cdot}15$ as in Sisson et al. (2007), the Markov chain mixes slowly, but still produces an acceptable fit over $T = 10^6$ iterations. Furthermore, the true target is available here as

$$\pi(\theta \mid |x|<\epsilon) \propto \Phi(\epsilon-\theta)-\Phi(-\epsilon-\theta)+\Phi\{10(\epsilon-\theta)\}-\Phi\{-10(\epsilon+\theta)\}$$

and, for the value $\epsilon = 0{\cdot}025$, it is indistinguishable from the exact posterior density $\pi(\theta \mid y=0)$. We can thus clearly separate the issue of poor convergence of the algorithm, depending on τ, from the issue of approximating the posterior density, depending on ϵ.

Sisson et al. (2007) modified the approximate Bayesian computation algorithm using partial rejection control (Liu, 2001). The method is sequential in that simulated populations of N points are generated at each iteration of the algorithm and that those populations are exploited to produce better proposals for a given target distribution in later iterations. As demonstrated in Douc et al. (2007), the reliance on earlier populations to build proposals preserves convergence properties provided an importance sampling perspective is adopted. Progressive improvement in the performance of proposals is the appeal of using a sequence of samples, rather than a single sample. The sequential feature of the problem is augmented by the fact that the tolerance ϵ decreases with iterations, being chosen either from a deterministic scale or from quantiles of earlier iterations as in Beaumont et al. (2002).

Sisson et al.'s (2007) algorithm produces samples $(\theta_1^{(t)},\ldots,\theta_N^{(t)})$ by using, at iteration $t=1$, a regular approximate Bayesian computation step and, at each iteration $t=2,\ldots,T$, Markov transition kernels K_t for the generation of the $\theta_i^{(t)}$s, namely, $\theta_i^{(t)} \sim K_t(\theta \mid \theta^\star)$, until $x \sim f(x \mid \theta_i^{(t)})$ is such that $\varrho(x,y)<\epsilon$, where $\theta^\star$ is selected at random among the previous $\theta_i^{(t-1)}$s with probabilities $\omega_i^{(t-1)}$ $(i=1,\ldots,N)$. The probability $\omega_i^{(t)}$ is derived by an importance sampling argument,

$$\omega_i^{(t)} \propto \pi\left(\theta_i^{(t)}\right)L_{t-1}\left(\theta^\star \mid \theta_i^{(t)}\right)\left\{\pi(\theta^\star)K_t\left(\theta_i^{(t)} \mid \theta^\star\right)\right\}^{-1}, \qquad (1)$$

where L_{t-1} is an arbitrary transition kernel. In their examples, Sisson et al. (2007) use $L_{t-1}(\theta' \mid \theta) = K_t(\theta \mid \theta')$, so the weights are all equal under a uniform prior. The ratio (1) is based on Del Moral et al. (2006) who use a sequence of backward kernels L_{t-1} in a sequential Monte Carlo algorithm to achieve unbiasedness, up to the renormalization effect, in the marginal distribution of the current value without computing this intractable marginal.

In this paper, we show by both theoretical and experimental arguments that the weight (1) is biased. Moreover, we introduce a population Monte Carlo correction to the algorithm, based on genuine importance sampling arguments, we demonstrate its applicability and the improvement it brings compared to the partial rejection control version. We also illustrate its efficiency in a population genetics example, when compared to two standard alternatives.

2. Bias of the partial rejection control version

2·1. *Distribution of the partial rejection control sample*

In order to display the bias in the weight (1) as clearly as possible, we consider the limiting case when $\epsilon = 0$, and both Pritchard et al.'s (1999) and Marjoram et al.'s (2003) algorithms are correct samplers from $\pi(\theta \mid y)$. The corresponding partial rejection control version selects a $\theta^{(t-1)}$ from the previous sample and then generates both $\theta' \sim K_t(\theta \mid \theta^\star)$ and $x \sim f(x \mid \theta')$ until $x = y$. To properly evaluate the bias associated with one step of Sisson et al.'s (2007) algorithm, we also assume that the previous sample is correctly generated from the target, i.e. that $\theta^\star \sim \pi(\theta \mid y)$. Then, denoting by $\theta^{(t-1)}$ the selected $\theta^\star$, the joint density of the accepted pair $(\theta^{(t-1)}, \theta^{(t)})$ is proportional to $\pi(\theta^{(t-1)} \mid y)K_t(\theta^{(t)} \mid \theta^{(t-1)})f(y \mid \theta^{(t)})$, where the normalization constant only depends on y. Using (1), the weighted distribution of $\theta^{(t)}$ is such that, for an arbitrary integrable function $h(\theta)$, $E\{\omega_t h(\theta^{(t)})\}$ is proportional to

$$\int\!\!\int h\left(\theta^{(t)}\right) \frac{\pi\left(\theta^{(t)}\right) L_{t-1}\left(\theta^{(t-1)} \mid \theta^{(t)}\right)}{\pi\left(\theta^{(t-1)}\right) K_t\left(\theta^{(t)} \mid \theta^{(t-1)}\right)}\, \pi\left(\theta^{(t-1)} \mid y\right) K_t\left(\theta^{(t)} \mid \theta^{(t-1)}\right) f\left(y \mid \theta^{(t)}\right) \mathrm{d}\theta^{(t-1)} \mathrm{d}\theta^{(t)}$$

$$\propto \int\!\!\int h\left(\theta^{(t)}\right) \frac{\pi\left(\theta^{(t)}\right) L_{t-1}\left(\theta^{(t-1)} \mid \theta^{(t)}\right)}{\pi\left(\theta^{(t-1)}\right) K_t(\theta^{(t)} \mid \theta^{(t-1)})} \pi\left(\theta^{(t-1)}\right) f\left(y \mid \theta^{(t-1)}\right) K_t\left(\theta^{(t)} \mid \theta^{(t-1)}\right) f\left(y \mid \theta^{(t)}\right) \mathrm{d}\theta^{(t-1)} \mathrm{d}\theta^{(t)}$$

$$\propto \int h(\theta^{(t)}) \pi\left(\theta^{(t)} \mid y\right) \left\{ \int L_{t-1}\left(\theta^{(t-1)} \mid \theta^{(t)}\right) f\left(y \mid \theta^{(t-1)}\right) \mathrm{d}\theta^{(t-1)} \right\} \mathrm{d}\theta^{(t)}$$

with all proportionality terms being functions of y only. Therefore, we can conclude that there is a bias in the weight (1) unless the inner function integrates in $\theta^{(t-1)}$ to the same constant for all values of $\theta^{(t)}$. Apart from this special case, which is achievable when $L_{t-1}(\theta^{(t-1)} \mid \theta^{(t)}) = g(\theta^{(t-1)})$ but not in the random-walk-type proposal when $L_{t-1}(\theta^{(t-1)} \mid \theta^{(t)}) = \varphi(\theta^{(t-1)} - \theta^{(t)})$, (1) is incorrect since the weighted output is not distributed from $\pi(\theta \mid y)$.

Paradoxically, the weight (1) used in this partial rejection control version lacks a $f(y \mid \theta^{(t-1)})$ term in its denominator, while the method is used when $f(y \mid \theta)$ is not available. This is exactly the difference between the weights used in Sisson et al. (2007) and those used in Del Moral et al. (2006), namely that, in the latter paper, the posterior $\pi(\theta^{(t-1)} \mid y)$ explicitly appears in the denominator instead of the prior. The accept-reject principle at the core of approximate Bayesian computation allows the replacement of the posterior by the prior in the numerator of the ratio (1), but not in the denominator.

2·2. *A mixture illustration*

When using the standard version of the algorithm that includes an additional approximation due to the tolerance zone $\varrho(x, y) < \epsilon$, there is no reason for the bias to vanish, even though experiments show that the bias in the weights (1) for the tolerance target $\pi\{\theta \mid \varrho(x, y) < \epsilon\}$ generally decreases as ϵ increases, which agrees with the fact that the limiting case is the prior. The following example illustrates this point:

Example 1. For the mixture target and a normal random walk kernel K_t, the first row of Fig. 1 shows the output of five consecutive iterations of Sisson et al.'s (2007) algorithm, using a decreasing sequence of ϵ_ts, from $\epsilon_1 = 2$ down to $\epsilon_5 = 0{\cdot}01$, and a standard deviation in K_t equal to $\tau = 0{\cdot}15$. Clearly, using this algorithm leads to a bias in terms of the tolerance target for all values of ϵ_t, the tails being poorly covered. By contrast, using a much larger $\tau = 1/0{\cdot}15$ results in a good fit of the tolerance target, as shown by Fig. 2 in Sisson et al. (2007). This fit does not contradict the bias exhibited above since using a large scale in the proposal K_t is very close to using a flat prior distribution. The corresponding normal density is then almost constant in the interval $[-3, 3]$ that supports the posterior distribution. For $\tau = 1/0{\cdot}15$, using the weight (1) is thus equivalent to Pritchard et al.'s (1999) original algorithm and therefore as inefficient for this target.

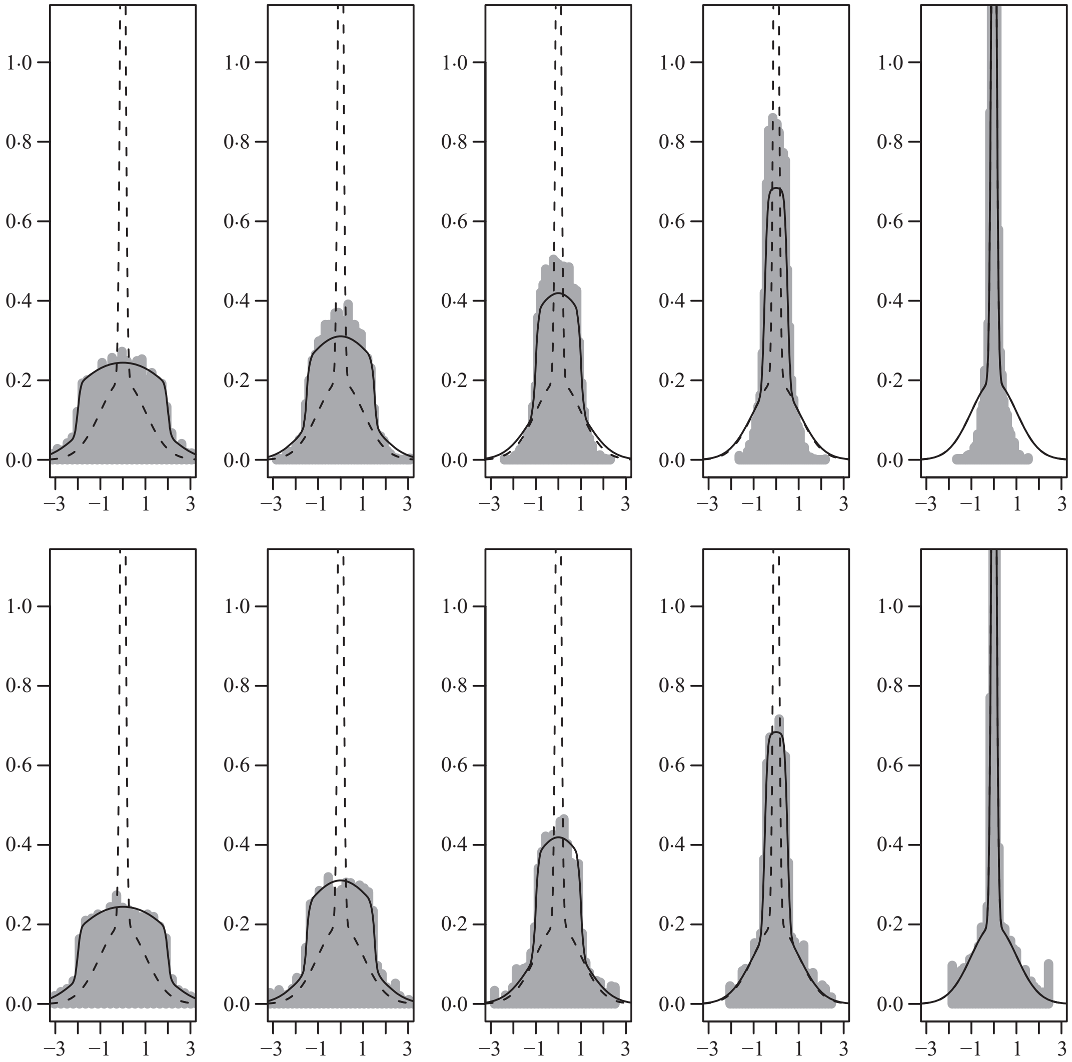

Fig. 1. Histograms of the weighted samples of size $M = 5 \times 10^3$ produced by five consecutive uses of (1) on the first row and the population Monte Carlo version on the second row, for the mixture target, with $\epsilon_t = 2, 1{\cdot}5, 1, 0{\cdot}5, 0{\cdot}01$, using for both rows the scale $\tau = 0{\cdot}15$. The exact posterior density is plotted as a dotted curve, while the target of the simulation algorithm, $\pi\{\theta \mid \varrho(x, y) < \epsilon\}$, is represented by solid lines. Both densities are identical in the last column.

3. CORRECTION VIA IMPORTANCE SAMPLING AND THE POPULATION MONTE CARLO ALGORITHM

3·1. *The population Monte Carlo algorithm*

Since the missing factor in (1) is the unknown likelihood $f(x \mid \theta^{(t-1)})$, a first resolution of the problem is to resort to estimating the likelihood based on earlier samples. But a standard importance sampling perspective allows for a more direct approach, in a spirit similar to the population Monte Carlo algorithm of Cappé et al. (2004).

Since the tth iteration sample is produced from the proposal distribution

$$\hat{\pi}_t\left(\theta^{(t)}\right) \propto \sum_{j=1}^{N} \omega_j^{(t-1)} K_t\left(\theta^{(t)} \mid \theta_j^{(t-1)}\right),$$

a natural importance weight associated with an accepted simulation $\theta_i^{(t)}$ is $\omega_i^{(t)} \propto \pi(\theta_i^{(t)})/\hat{\pi}_t(\theta_i^{(t)})$. The unbiasedness of this correction results from the identity

$$E\left\{\omega^{(t)} h\left(\theta^{(t)}\right)\right\} \propto \int h\left(\theta^{(t)}\right) \frac{\pi\left(\theta^{(t)}\right)}{\hat{\pi}\left(\theta^{(t)}\right)}\, \hat{\pi}\left(\theta^{(t)}\right) f\left(\theta^{(t)} \,|\, y\right) \tilde{\pi}\left(\theta^{(t-1)}\right) \mathrm{d}\theta^{(t)}\, \mathrm{d}\theta^{(t-1)},$$

which does not depend on the distribution $\tilde{\pi}$ of the $\theta_j^{(t-1)}$s. As in the original population Monte Carlo method, the fact that K_t may depend on simulations from earlier iterations does not jeopardize the validity of the method. In addition, Douc et al. (2007) proved that K_t must be modified at each iteration for the iterations to bring an asymptotic improvement on the Kullback–Leibler divergence between the proposal $\hat{\pi}_t$ and the target: for instance, if the variance of the random walk does not change from one iteration to the next, the approximation of the target by $\hat{\pi}_t$ does not change either, and it is then more efficient to run a single iteration with twice as many points. Since $\hat{\pi}_t$ is an approximation to the distribution of the sample simulated at iteration t, when marginalized against all previous samples, this version of the approximate Bayesian computation algorithm can be interpreted as an approximate version of the sequential Monte Carlo algorithm of Del Moral et al. (2006), when using the optimal backward kernel.

When considering componentwise independent random walk proposals,

$$K_t\left(\theta_k^{(t)} \,\middle|\, \theta_k^{(t-1)}\right) = \tau_k^{-1}\varphi\left\{\tau_k^{-1}\left(\theta_k^{(t)} - \theta_k^{(t-1)}\right)\right\}$$

for each component $\theta_k^{(t)}$ of the parameter vector $\theta^{(t)}$, the asymptotically optimal choice of the scale factor τ_k is available. Indeed, when using a Kullback–Leibler measure of divergence between the target and the proposal,

$$E\left(\log\left[\pi\left(\theta^{(t)} \,|\, y\right) \Big/ \prod_k \tau_k^{-1}\varphi\left\{\tau_k^{-1}\left(\theta_k^{(t)} - \theta_k^{(t-1)}\right)\right\} f(y \mid \theta^{(t)})\right]\right),$$

where the expectation E is taken under the product distribution $(\theta^{(t)}, \theta^{(t-1)}) \sim \pi(\theta^{(t)} \mid y) \times \pi(\theta^{(t-1)} \mid y)$, the minimization of the Kullback divergence leads to the componentwise maximization of $E[\log \tau_k^{-1}\varphi\{\tau_k^{-1}(\theta_k^{(t)} - \theta_k^{(t-1)})\}]$ under the product distribution $\pi(\theta^{(t)} \mid y) \times \pi(\theta^{(t-1)} \mid y)$. The optimal scale is then equal to $E\{(\theta_k^{(t)} - \theta_k^{(t-1)})^2\}$, that is, $\tau_k^2 = 2\mathrm{var}(\theta_k \mid y)$, under the posterior distribution. The implementation of this updating scheme on the scale is straightforward.

The algorithmic rendering of the corresponding optimized population Monte Carlo scheme is thus:

Population Monte Carlo approximate Bayesian computation:

```
Given a decreasing sequence of tolerance thresholds $\epsilon_1 \geqslant \cdots \geqslant \epsilon_T$,

1. At iteration $t = 1$,
     For $i = 1, \ldots, N$
       Simulate $\theta_i^{(1)} \sim \pi(\theta)$ and $x \sim f(x \mid \theta_i^{(1)})$ until $\varrho(x, y) < \epsilon_1$
       Set $\omega_i^{(1)} = 1/N$
   Take $\tau_1^2$ as twice the empirical variance of the $\theta_i^{(1)}$s
2. At iteration $2 \leqslant t \leqslant T$,
     For $i = 1, \ldots, N$, repeat
       Pick $\theta_i^\star$ from the $\theta_j^{(t-1)}$s with probabilities $\omega_j^{(t-1)}$
       Generate $\theta_i^{(t)} \mid \theta_i^\star \sim \mathcal{N}(\theta_i^\star, \tau_t^2)$ and $x \sim f(x \mid \theta_i^{(t)})$
     Until $\varrho(x, y) < \epsilon_t$
     Set $\omega_i^{(t)} \propto \pi(\theta_i^{(t)}) / \sum_{j=1}^N \omega_j^{(t-1)} \varphi\{\tau_t^{-1}(\theta_i^{(t)} - \theta_j^{(t-1)})\}$
   Take $\tau_{t+1}^2$ as twice the weighted empirical variance of the $\theta_i^{(t)}$s
```

The expression of the importance weight $\omega_i^{(t)}$ involves a sum of N terms, so the computational cost of this step of the population Monte Carlo approximate Bayesian computation algorithm is $\mathrm{O}(T\,N^2)$. However, in realistic applications of approximate Bayesian computations, the main cost is associated with the previous step, namely, the repeated simulations from the sampling density. For instance, in the population genetics example, the algorithm spends at least 95% of the computing time in the repeat loop and less than 5% on the remaining computations.

3·2. *A mixture illustration*

For the mixture target, in Example 1 using the population Monte Carlo version leads to a recovery of the target, whether using a fixed standard deviation $\tau = 0{\cdot}15$ as shown on the second row of Fig. 1, or $\tau = 1/0{\cdot}15$, or a sequence of adaptive τ_ts as in the above algorithm. The graphical difference between these implementations is genuinely difficult to spot; the estimated variance stabilizes very quickly in this toy example.

3·3. *A population genetics example*

This example considers a simple evolutionary scenario of two populations having diverged from a common ancestral population. Data consists of the genotypes at five microsatellite loci of 50 diploid individuals from each of the populations. Loci are assumed to evolve according to the strict stepwise mutation model: when a mutation occurs, the number of repeats of the mutated gene increases or decreases by one unit with equal probability. Once diverged, populations do not exchange genes and there is no migration. The natural parameters of this model are the three effective population sizes N_1, N_2 and N_{anc}; the time of divergence t_{div}; and the mutation rate μ, assumed here to be common to all loci. While the analyses will be performed with these natural parameters, only the identifiable combinations of those, such as the three parameters $\theta_1 = 4N_1\mu$, $\theta_2 = 4N_2\mu$ and $\theta_a = 4N_{\text{anc}}\mu$, and the parameter $\tau_{\text{div}} = t_{\text{div}}\mu$, will be considered in the output.

In this experiment, several simulated datasets have been produced using the software developed by Cornuet et al. (2008), with the following parameter values: $N_1 = N_{\text{anc}} = 10\,000$, $N_2 = 2000$, $t_{\text{div}} = 1000$ and $\mu = 0{\cdot}0005$, of which one is presented in this paper, but identical conclusions were drawn from the other datasets.

The tolerance region $\{\varrho(x, y) < \epsilon\}$ used in the approximate Bayesian computation schemes is based on twelve summary statistics as in Cornuet et al. (2008), namely, the mean number of alleles, mean genic diversity, mean size variance and mean Garza–Williamson M index for each population sample, F_{ST} and $(\delta\mu)^2$ distances between population samples and mean probability of assignment of each sample to the other population. The distance $\varrho(x, y)$ is then chosen as the Euclidean distance between the observed and the simulated summary statistics, normalized by their standard deviation under the predictive model. The derivation of the distance thus requires a preliminary evaluation of those standard deviations for each summary statistic by simulation.

Each dataset has been submitted to three parallel analyses: a standard approximate Bayesian computation analysis following Beaumont et al. (2002), performed with Cornuet et al.'s (2008) software, a tempered Markov chain Monte Carlo analysis as described in Bortot et al. (2007) and a population Monte Carlo analysis. In each case, the same prior distributions have been used, consisting of a $U[10^2, 10^5]$ prior on the three effective sizes, a $U[10, 10^4]$ prior on the time of divergence, and a $U[10^{-4}, 10^{-3}]$ prior on the mutation rate. In addition, since Beaumont et al.'s (2002) algorithm includes a final local regression adjustment, this adjustment has been performed on the output of both alternative algorithms.

In order to operate a fair comparison between the three methods, the computing times are kept identical. Thus, reference posterior distributions have been constructed with Beaumont et al.'s (2002) approach, using $N = 3 \times 10^4$ simulated datasets and choosing ϵ as the $0{\cdot}01$ quantile of the distances. For Bortot et al.'s (2007) approach, the tuning parameter driving the acceptance threshold ϵ is drawn from an exponential prior with parameter equal to the mean distance of the ten closest datasets, among 5000 generated in a pilot simulation, followed by 23 000 regular iterations with a thinning factor of 23. Each Markov move combined two simultaneous updates: an independent draw of ϵ from its exponential prior and a random walk based on a lognormal deviate with standard deviation equal to $0{\cdot}3$ of one of the other parameters chosen uniformly at random, the latter involving the computation of a Hastings term. Lastly, in the population Monte Carlo version, ϵ_1 is based on the preliminary simulation as the $0{\cdot}1$ quantile and four iterations are performed with $\epsilon_2 = 0{\cdot}75\epsilon_1$, $\epsilon_3 = 0{\cdot}9\epsilon_2$ and $\epsilon_4 = 0{\cdot}9\epsilon_3$, and with truncated normals in the random walk. Each iteration is

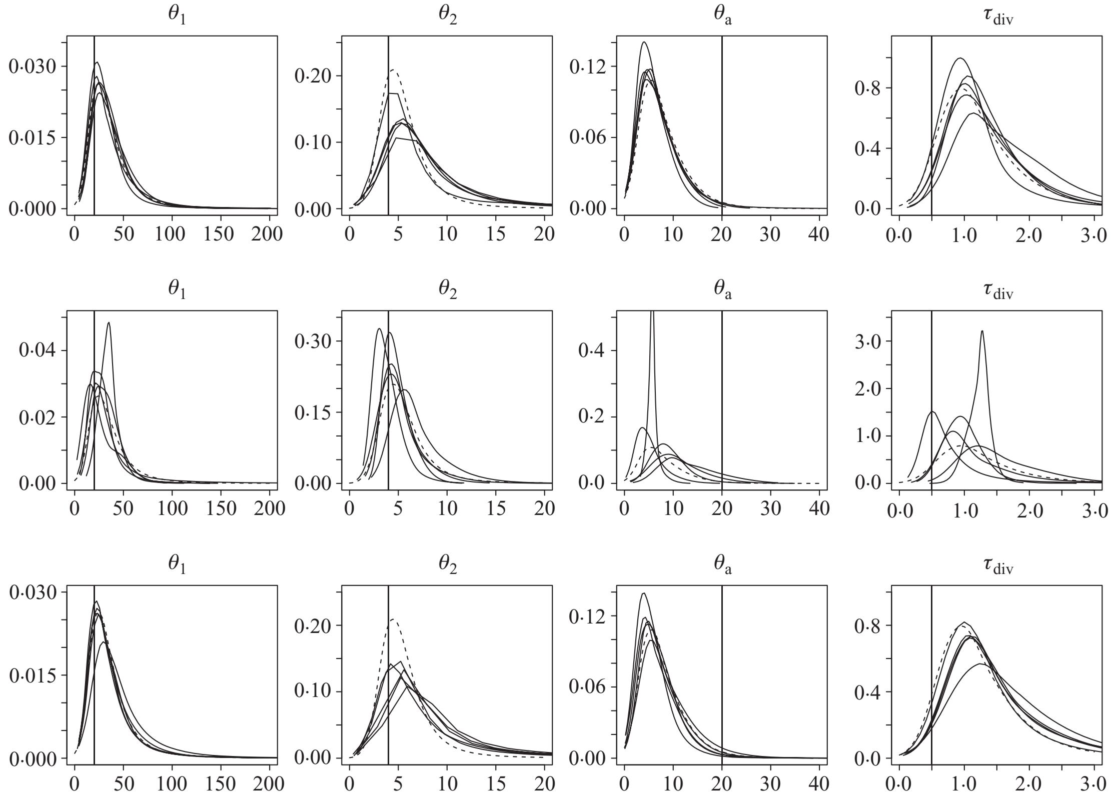

Fig. 2. Variability of the different alternative schemes evaluated through five replicates of the density estimates of the posterior distributions of four identifiable parameters of the population genetics model. First line: population Monte Carlo version; second line: tempered Markov chain Monte Carlo version; third line: Beaumont et al.'s (2002) version of the approximate Bayesian computation algorithm, against the reference posterior, obtained by 5×10^5 simulations in Beaumont et al.'s (2002) version (dotted line). The vertical lines identify the true values of the parameters.

based on samples of size 10^3. All three versions then require an average 4·5 minutes. Figure 2 summarizes the output of the comparison between the three methods over five independent runs for the same simulated dataset and it shows that, at least in this example, the posterior evaluations based on population Monte Carlo are more stable than Bortot et al.'s (2007) algorithm, while comparable with Beaumont et al.'s (2002).

4. Conclusion

While Sisson et al.'s (2007) algorithm induces biased weights, with a visible impact on the quality of the approximation, we have shown that the same Markov transition kernels and thus the same computing power can be used to produce an unbiased scheme.

The population Monte Carlo scheme is based on an importance argument that does not require a backward kernel as in (1). We have thus established that the adaptive scheme of Cappé et al. (2008) is also appropriate in this setting, towards a better fit of the proposal kernel K_t to the target $\pi\{\theta \mid \varrho(x, y) < \epsilon\}$. From a practical point of view, the number of iterations T can be controlled via the modifications in the parameters of K_t, a stopping rule being that the iterations should stop when those parameters have settled, while the more fundamental issue of selecting a sequence of ϵ_ts towards a proper approximation of the true posterior can rely on the stabilization of the estimators of some quantities of interest associated with this posterior.

ACKNOWLEDGEMENTS

This research is partly supported by the Agence Nationale de la Recherche through the Misgepop project. Marin and Robert are affiliated with CREST, Paris. Parts of this paper were written by Robert in the Isaac Newton Institute, Cambridge, whose peaceful and stimulating environment was deeply appreciated. Helpful comments from the editorial board of *Biometrika* and from O. Cappé are gratefully acknowledged.

REFERENCES

BEAUMONT, M., ZHANG, W. & BALDING, D. (2002). Approximate Bayesian computation in population genetics. *Genetics* **162**, 2025–35.

BORTOT, P., COLES, S. & SISSON, S. (2007). Inference for stereological extremes. *J. Am. Statist. Assoc.* **102**, 84–92.

CAPPÉ, O., DOUC, R., GUILLIN, A., MARIN, J.-M. & ROBERT, C. (2008). Adaptive importance sampling in general mixture classes. *Statist. Comp.* **18**, 447–59.

CAPPÉ, O., GUILLIN, A., MARIN, J.-M. & ROBERT, C. (2004). Population Monte Carlo. *J. Comp. Graph. Statist.* **13**, 907–29.

CORNUET, J.-M., SANTOS, F., BEAUMONT, M. A., ROBERT, C. P., MARIN, J.-M., BALDING, D. J., GUILLEMAUD, T. & ESTOUP, A. (2008). Inferring population history with DIYABC: a user-friendly approach to Approximate Bayesian Computation. *Bioinformatics* **24**, 2713–19.

DEL MORAL, P., DOUCET, A. & JASRA, A. (2006). Sequential Monte Carlo samplers. *J. R. Statist. Soc.* B **68**, 411–36.

DOUC, R., GUILLIN, A., MARIN, J.-M. & ROBERT, C. (2007). Convergence of adaptive mixtures of importance sampling schemes. *Ann. Statist.* **35**, 420–48.

LIU, J. (2001). *Monte Carlo Strategies in Scientific Computing*. New York: Springer.

MARJORAM, P., MOLITOR, J., PLAGNOL, V. & TAVARÉ, S. (2003). Markov chain Monte Carlo without likelihoods. *Proc. Nat. Acad. Sci. USA* **100**, 15324–28.

PRITCHARD, J. K., SEIELSTAD, M. T., PEREZ-LEZAUN, A. & FELDMAN, M. W. (1999). Population growth of human Y chromosomes: a study of Y chromosome microsatellites. *Molec. Biol. Evol.* **16**, 1791–98.

SISSON, S. A., FAN, Y. & TANAKA, M. (2007). Sequential Monte Carlo without likelihoods. *Proc. Nat. Acad. Sci. USA* **104**, 1760–65.

[*Received July* 2008. *Revised April* 2009]