d d a d a

c h a c h

University of Reading

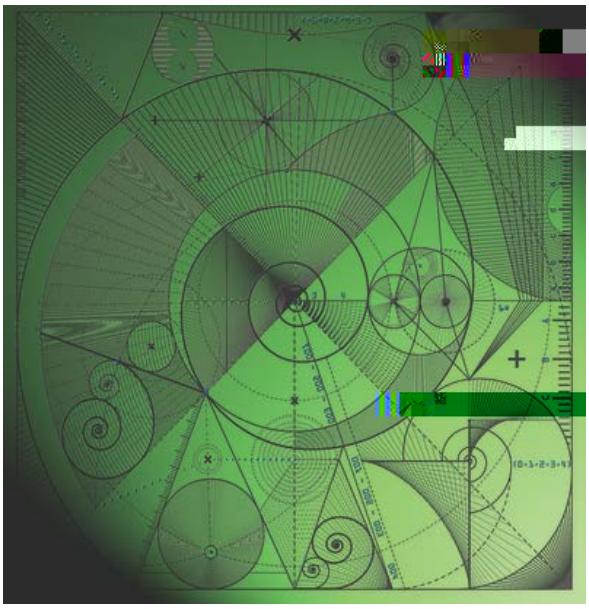
e g c d h c h h

g e g c

Lazy ABC

by

c c h g c a



1 Introduction

Approximate Bayesian computation (ABC) algorithms are a popular method of inference for a wide class of otherwise intractable probability models in applications such as population genetics, ecology, and systems biology (Beaumont, 2010; Marin et al., 2012). They select parameter vectors for which datasetsy simulated from the model of interest are su ciently close to the observations. A bottleneck is the computational cost of producing the large quantity of model simulations needed, which becomes increasingly severe for more detailed models. However, it is often clear during a simulation that it is unpromising. For example it is likely to produce a poor match or to require excessive computation time. This paper presentslazy ABC, an importance sampling method which abandons some such simulations, a step referred to asearly stopping exploiting information from incomplete simulations to save time. The result me. inflationel17 16(infl)-267(ar16(inf)-285(8im)2415)-376(inf(detai(infscal(prob

The nal likelihood estimator is based on$X$ and Y

predictions given forming X in the notation above) to gain speed bene ts without incurring additional approximation errors. More generally, there has been much interest over the past decade in Bayesian inference algorithms with random weights (e.g. Beaumont, 2003; Andrieu and Roberts, 2009; Fearnhead et al., 2010; Tran et al., 2014). A novelty of lazy ABC is that it introduces a random factor to the weights to reduce computation time, rather than to deal with intractability.

Rejection control in sequential Bayesian algorithms (Liu et al., 1998) uses a similar idea to lazy ABC. Here after the rst t stages of the sequential analysis, a proposal (typically a sequence of latent states at times $1; 2; : : : ; t$) with weight w is allowed to continue with probability $= \min(1; w=c)$ for some constant c. On continuation the weight is updated to $w=$ and otherwise a new proposal is generated. Novelties of the current work are nding an optimal form for and allowing it depend on information other than w.

The remainder of the paper is structured as follows. Section 2 contains background material. To help later developments this presents ABC within the framework of random weight importance sampling. Section 3 gives the lazy ABC algorithm and proves it targets the correct distribution. Section 4 presents theory and practical methods for tuning the algorithm. A related result is also given on the optimal importance distribution for standard ABC importance sampling. Section 5 contains the application to spatial extremes and Section 6 is a discussion. Appendices contain proofs and material on lazy ABC with multiple stopping decisions.

2 Importance sampling

Consider analysing data y_{obs} under a probability model with density $(y|\ )$ and parameters . The likelihood is de ned as $L(\ ) = (y_{obs}|\ )$. Bayesian inference introduces a prior distribution with density $(\ )$ and aims to nd the posterior distribution $(\ |y_{obs}) = (\ )L(\ )=(y_{obs})$, where $(y_{obs}) = \int (\ )L(\ )d$, or at least to estimate the posterior expec-

tation $E[h(\ )|y_{obs}]$ of a generic function $h(\ )$. Importance sampling is a simple method to do this. Parameter values $_{1:N}$ are simulated independently from an importance density $g(\ )$ and given weights $w_i = L(\ _i)\ (\ _i)/g(\ _i)$ (n.b. $_{1:N}$ represents the sequence $_i$, $1 \ i \ N$. Similar notation is used later.) It is assumed throughout that $g(\ ) > 0$ whenever $(\ ) > 0$. Each of the $(\ _i; w_i)$ pairs can be computed in parallel, allowing for e cient implementation.

A Monte Carlo estimate of $E[h(\ )|y_{obs}]$ is $_h = \frac{\sum_{i=1}^{N} h(\ _i) w_i}{\sum_{i=1}^{N} w_i}$. Two properties of importance sampling estimates are

$$_h \ ! \ E[h(\ )|y_{obs}] \quad \text{almost surely as} \quad N \ ! \ 1; \tag{1}$$

$$E[N^{-1} \sum_{i=1}^{N} w_i] = \ (y_{obs}): \tag{2}$$

See Geweke (1989) for

be seen by noting that Algorithm 1a is equivalent to a deterministic weight importance

Input (general):

Prior density $\pi(\theta)$ and importance density $g(\theta)$.

Number of iterations to perform N.

Input (RW-IS):

Likelihood estimator $\hat{L}$.

Input (ABC):

Observed data y_{obs}.

Summary statistics $S(\cdot)$, distance function $d(\cdot,$

choice of $S(\cdot)$ involves a trade-off between low dimension and informativeness. For further background details on all aspects of ABC see the review articles of Beaumont (2010) and Marin et al. (2012).

3 Lazy ABC

This section defines lazy ABC and shows it produces valid results.

Definition 1. Lazy ABC is Algorithm 1a, using a likelihood estimator of the form (3) under conditions C1-C3.

$$\hat{L}_{\text{lazy}} = \begin{cases} \hat{L}_{\text{ABC}} = \epsilon(\theta, X) & \text{with probability } \epsilon(\theta, X) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

C1 $\epsilon(\theta, x)$ is a function with codomain $[0, 1]$.

C2 $\epsilon(\theta, x) > 0$ whenever $\Pr(\hat{L}_{\text{ABC}} > 0 \mid \theta, x) > 0$.

C3 The random variable X is such that both $X \mid \theta$ and $Y \mid \theta, x$ can be simulated from.

The following theorem shows that the estimator $\hat{L}_{\text{lazy}}$ can be used in a RW-IS algorithm, or a pseudo-marginal MCMC algorithm, and give valid results.

Theorem 1. Conditional on θ, $\hat{L}_{\text{lazy}}$ is a non-negative unbiased estimator of $L_{\text{ABC}}(\theta)$.

Proof. Non-negativity is immediate. For unbiasedness first observe that $E(\hat{L}_{\text{lazy}} \mid \theta, x)$ equals zero when $\epsilon(\theta, x) = 0$ and $E(\hat{L}_{\text{ABC}} \mid \theta, x)$ otherwise. By C2 if $\epsilon(\theta, x) = 0$ then $\Pr(\hat{L}_{\text{ABC}} > 0 \mid \theta, x) = 0$ and so $E(\hat{L}_{\text{ABC}} \mid \theta, x) = 0$. Hence $E(\hat{L}_{\text{lazy}} \mid \theta, X) = E(\hat{L}_{\text{ABC}} \mid \theta, X)$. Taking expectations ov

the continuation simulation stage. It is often useful later to have $(\ ;x) = \ (\ (\ ;x))$ where $(\ ;x)$ is referred to as the *decision statistics*.

Notation is now introduced for expected CPU times: $T_1(\ )$ is for steps 1 and 2 above conditional on $\ $, $T_2(\ ;\ )$ is for step 3 conditional on $(;\ )$ and $T(\ )$ is for simulation from $\hat{L}_{ABC}$ conditional on $\ $. The first two are roughly the times of the initial simulation and continuation stages, but are defined to cover all steps involved in simulating from $\hat{L}_{lazy}$.

It is assumed that

$$T(\ ) \quad T_1(\ ) + E[\,T_2(\ ;\ )j\ ]: \tag{4}$$

Roughly speaking this states that drawing from $\hat{L}_{lazy}$ conditional on no early stopping takes at least as long as drawing from $\hat{L}_{ABC}$. The difference is due to computational overheads of considering stopping. It is also convenient to define $T_1 = E[\,T_1(\ )]$ and $T_2(\ ) = E[\,T_2(\ ;\ )j\ ]$.

3.1 Lazy importance sampling

The above approach can be generalised to non-ABC situations to give *lazy importance sampling* (LIS). This is Algorithm 1a using a likelihood estimator of the form:

$$\hat{L}_{lazy} = \begin{cases} \hat{L} = (\ ;X\,) & \text{with probability} \quad (\ ;X\,) \\ 0 & \text{otherwise} \end{cases}$$

In addition to conditions C1 and C2 above assume:

C4 The distribution $(X;\hat{L})j\ $ is such that $\hat{L}j\ $ is a non-negative unbiased estimator of $L(\ )$, and both $Xj\ $ and $\hat{L}j\ ;x$ can be simulated from.

This framework can be used when $\hat{L}$ is an expensive unbiased estimator. It also allows cases where either or both of X and $\hat{L}$ are non-random. For example, X may be a deterministic approximation of the likelihood and $\hat{L}j\ $ may be a point mass at $L(\ )$.

All theorems and proofs of this paper also hold for lazy importance sampling, replacing $L_{ABC}(\ )$ and $\hat{L}_{ABC}$ with $L(\ )$ and $\hat{L}$, and making other small modifications noted in the text. In particular Theorem 1 shows that given conditions C1, C2 and C4 $\hat{L}_{lazy}$ is a non-negative unbiased estimator of $L(\ )$. However the practical application of lazy importance sampling is challenging as discussed in Section 6.

4 Tuning

There is considerable freedom to tune lazy ABC through the choice of χ (when to consider stopping) and $\ $ (the function assigning continuation probabilities). Section 4.1 proves a result on the most efficient choice of $\ $. This theory is used in Section 4.2 to motivate practical tuning methods.

Note that the case where $\ (\ )$ is based on $\ 2\ A$ for discrete A does not require the theoretical results below. Here $\ (\ )$ values can be selected by numerical optimisation of an estimate of the algorithm's efficiency based on pilot simulations. The methods that follow detail construction of such an estimate.

4.1 Theory

A commonly used tool for the analysis of importance sampling algorithms is the effective sample size (ESS). Liu (1996) argued that typically the variance of the importance sampling estimator is roughly equal to that of N_e independent samples where

$$N_e = NE(W)^2/E(W)^2;$$

and the random variable W is the weight generated in an iteration of importance sampling. The argument of Liu generalises immediately to RW-IS algorithms through the interpretation of them as importance sampling algorithms on an augmented parameter space given in Section 2.2.

Define efficiency as N_e/T where T is the CPU time of the algorithm (i.e. ignoring any execution time savings due to parallelisation.) Assume that T follows a central limit theorem in N. Then the delta method gives that for large N efficiency asymptotically equals

$$\frac{E(W)^2/E(W^2)}{E(T)/N}.$$

Theorem 2. Fix some decision statistics (θ, x). Amongst continuation probability functions of the form $\pi(\theta, x) = \pi(\theta, x))$, asymptotic efficiency is maximised by the following expression for some > 0,

$$\pi(\theta) = \min\left\{1, \left[\frac{E[\hat{L}_{ABC}\frac{(\theta)^2}{g(\theta)^2} \mid \theta]}{T_2(\theta)}\right]^{1/2}\right\}. \tag{5}$$

Proof. See Appendix B.1. □

Remark 1. Suppose $(\theta)/g(\theta) = u(\theta)$ i.e. this fraction is completely determined by θ. For example this is the case in ABC rejection sampling where $g(\theta) \quad (\theta)$. Then (5) becomes

$$\pi(\theta) = \min\left\{1, u(\theta)\left[\frac{(\theta)}{T_2(\theta)}\right]^{1/2}\right\}; \tag{6}$$

where $(\theta) = E[\hat{L}_{ABC} \mid \theta] = \Pr(d(S(Y), S(y_{obs})) \quad \mid \theta)$.

Remark 2. Theorem 2 and Remark 1 hold for LIS with $\hat{L}_{ABC}$ replaced by $\hat{L}^2$.

A simple closed form expression for does not appear possible. In the practical tuning methods below is found numerically, and the behaviour of this numerical estimate investigated by simulation study (see Figure 1B).

By viewing ABC-IS as a special case of lazy ABC, Theorem 2 can be applied to find the optimal choice of $g(\theta)$ for ABC-IS.

Corollary 1. The asymptotic efficiency of ABC-IS is maximised by $g(\theta) \propto (\theta)\left[\frac{(\theta)}{T(\theta)}\right]^{1/2}$, where $(\theta) = E(\hat{L}_{ABC} \mid \theta)$.

Proof. See Appendix B.2. □

Remark 3. A corresponding result to Corollary 1 holds for RW-IS with $\psi(\theta) = \mathrm{E}(\hat{L}^2 | \theta)$.

Remark 4. The special case of Corollary 1 with $T(\theta)$ constant matches the result of Appendix A in Fearnhead and Prangle (2012).

Note that it is not clear what the optimal choice of $g(\theta)$ is for lazy ABC. The examples later use typical choices from the ABC literature, but a better choice may improve lazy ABC performance further.

4.2 Methods

Theorem 2 motivates choosing θ by estimating (6). This section details a method to implement this approach. Its effectiveness is discussed in Section 6.

Tuning begins with a pilot run of N' iterations of ABC-IS. This is used to estimate $\psi(\theta)$ and $T_2(\theta)$ for various choices of X and θ, considering only θ is such that Remark 1 can be applied. Under each of these choices, θ is found by numerically maximising an estimate of efficiency. The optimal choice of X and θ is then made. Note that estimation of $\psi(\theta)$ and $T_2(\theta)$ is challenging if θ is high dimensional e.g. for $\theta = (\theta, x)$. Therefore a low dimensional θ is recommended. To ensure Remark 1 applies $\psi(\theta) = g(\theta)$ can form one component of θ if necessary. Following tuning, N iterations of lazy ABC are performed (unless the estimated efficiency gains are judged inadequate). Detailed comments on several aspects of this method follow.

4.2.1 Estimation of $T_2(\theta)$

It may often suffice to treat $T_2(\theta)$ as constant and estimate it as the mean CPU time of the

shows$T_2(\ )$ varies little relative to $(\ )$. Alternatively, statistical methods such as regression can be used for estimation, which is straightforward when is low dimensional.

4.2.2 Estimation of $(\ )$

Estimation of $(\ )$ is more di cult. Two approaches are suggested: the \standard" approach, producing $\hat{}^{(1)}$, attempts accurate estimation but involves strong assumptions; the \conservative" approach, producing $\hat{}^{2\text{ves strongi5}}$

4.2.3 Estimating e ciency

The tuning method outlined above requires the use of N^0 pilot run iterations to estimate the e ciency of lazy ABC under various choices of tuning details (in particular X, and). It is su cient to estimate $[E(W^2)E(T)]^{-1}$, as this equals e ciency up to a constant of proportionality. This can be used to estimate e ciency relative to ABC-IS, which is a particularly interpretable form of the results as it shows the e ciency improvement of using lazy ABC.

Assume that for a particular choice of tuning details the following are available for $1 \le i \le N^0$: $t_i^{(1)}$ - initial simulation stage time; $t_i^{(2)}$ - continuation simulation stage time; γ_i - continuation probability; $\hat{\alpha}_i$ - estimate of $E(\hat{L}_{ABC} \mid \gamma_i)$; u_i - ratio $(\ )=g(\ )$. An estimate up to proportionality of e ciency is then $[\hat{W}^2\hat{T}]^{-1}$ where $\hat{W}^2 = N^{0-1}\sum_{i=1}^{N^0} u_i^2\hat{\alpha}_i/\gamma_i$ and $\hat{T} = \sum_{i=1}^{N^0} t_i^{(1)} + \sum_{i=1}^{N^0} \gamma_i t_i^{(2)}$. An estimate of e ciency of ABC-IS is formed by taking $\gamma_i \equiv 1$. Note that this typically overestimates T due to the overheads of considering stopping (see (4)). A more precise estimate would be possible using further pilot simulations of standard ABC.

4.2.4 Combining pilot and main run output

To make e cient use of the pilot run, it can be used in the nal output as well as for tuning. This is done by appending the pilot sequence of (θ, w) pairs to that from the main algorithm. Loosely speaking, since each individual sequence targets the same distribution, so does the combined sequence. More technically, it is straightforward to see that ABC versions of relations (1) and (2) are roughly true for the combined sequence when N and N^0 are large, and are exactly true as $N \to \infty$ regardless of N^0. Also note that on appending the sequences, gains in e ciency are possible by multiplying the weights of one sequence by a constant, but this is not implemented here as little improvement was observed in the application later.

4.2.5 Choice of

In ABC-IS, an appropriate value is often unknown a priori and is instead chosen based on the simulated $d(S(Y); s(y_{obs}))$ values. For lazy ABC in this situation one can use the pilot run to select a preliminary conservative choice of $_1$ as in Section 4.2.2 and perform lazy ABC with $= _1$. Alternative values of can then be investigated by updating the realised $\hat{L}_{ABC}$ values in the weight calculations. For $< _1$ this simply reduces the number of non-zero weights. However $_1$ is not recommended as this may introduce large weights and destabilise the importance sampling approximation.

Erhardt and Smith focus on the Whittle-Matérn correlation function with zero nugget

$$(h; c;) = \frac{2^{1 \quad v}}{( )} \quad \frac{h}{c} \quad K \quad \frac{h}{c} \quad ;$$

where is the gamma function and K is the modified Bessel function of the third kind with order . This has two parameters: range $c > 0$ and smoothness > 0.

A density function for the Schlather process is not available for $D > 2$, making inference difficult. Schlather (2002) provides a near-exact algorithm to simulate from the process based on only a finite number of copies of Y_i, motivating the use of ABC by Erhardt and Smith. They applied ABC rejection and importance sampling with a uniform prior on $[0; 10]^2$ and investigated several choices of summary statistics. The analysis here focuses on the choice they find most successful, based on tripletwise extremal coefficient estimators. Given a triple of 3 locations, $i; j; k$, this estimator is

$$\hat{}_{ijk} = P \frac{T}{\frac{T}{t=1} 1=\max(y_{t;i}\,; y_{t;j}\,; y_{t;k})}:$$

There are $O(D^3)$ such summaries, so Erhardt and Smith calculate a vector m of mean values within 100 clusters of triples, and use these as summary statistics. Their clustering process finds triples of similar shapes, ignoring differences of location and rotation. The ABC distance function between two vectors m_1 and m_2 of cluster means is

$$d(m_1; m_2) = \sum_{i=1}^{100} jm_{1i} \quad m_{2i}j: \qquad (7)$$

Although applying dimension reduction techniques to such high dimensional summaries has been shown to often improve ABC results (Fearnhead and Prangle, 2012), this is not investigated here as the aim is to investigate the efficiency improvements of lazy ABC.

of the approach of Erhardt and Smith.

5.2 Methods

Exploratory investigation of ABC code with $D = 20$ and $T = 100$ showed that the majority of time was spent simulating the data (7.1ms/iteration) and calculating extremal coefficient estimates (17.9ms/iteration), with the remaining steps being brief (3.1ms/iteration). The time costs of the first two of these scaled with D as roughly proportional to D and D^3 respectively, so the latter is expected to dominate for large D. Furthermore, interrupting and then resuming operations during the calculation of extremal coefficients is much simpler to implement than during simulation of data. Therefore the initial simulation stage of the lazy ABC analysis was chosen to be simulating the data at all locations, and extremal coefficient estimates at a subset of locations L. The continuation simulation stage was to calculate the remaining extremal coefficient estimates.

The decision statistic $\hat{d}$ was constructed as follows. Let m_{1i} be the ith cluster mean for the observed data. Let $\hat{m}_{2i}$ be the ith cluster mean for the simulated data using only extremal coefficient estimates available at the initial simulation stage, and B be the set of clusters for which any such estimates are available. Then define $\hat{d} = \sum_{i \in B} |m_{1i} - \hat{m}_{2i}|$. This is an estimate of the ABC distance $d(m_{1i}, \hat{m}_{2i})$

simulations with nearby values of $\hat{d}$. This was done for several $\hat{d}$ values and interpolated estimates elsewhere formed $\hat{l}_1(\hat{d})$. For the importance sampling case, $\log u$ was also included in each regression so that a number of functions mapping u to estimates of $\Pr(d = \hat{d} \mid \hat{d}, u)$ for various $\hat{d}$ values were produced which were used for interpolation. Calculation of $\hat{l}_2$ was as described in Section 4.2.2, taking g to give 100 acceptances in the pilot run. Given estimates of T_2 and , tuning was performed as described in Section 4.2, with optimisation over possible choices of d by backwards selection.

Three simulation studies were performed. The first replicated the rejection sampling analysis of Erhardt and Smith on several simulated datasets. These used $D = 20, T = 100$ and true parameter values shown in Table 1. Each dataset used a different set of observation locations with integer coordinates sampled from $[0, 10]^2$. The first analysis was a replication of the standard ABC analysis, using values corresponding to 200 acceptances. Then lazy ABC was performed on the same datasets under each method of estimating. To compare the methods fairly, lazy ABC used the same value as standard ABC and reused its random seeds so that the sequence of (X, Y) realisations is also the same.

The second simulation study investigated rejection sampling for a single larger simulated dataset with $D = 35$, $c = 0.5$ and $= 1$. Locations were chosen as before. As in a real application was not assumed to be known in advance and the approach of Section 4.2.5 was used to select this post-hoc. A complication for this dataset was that the simulation of Gaussian processes was difficult when both parameters were large: the default "direct method", based on Choleski decomposition, sometimes produced numerical errors. Simulation was possible via the turning bands method (TBM) but much slower (roughly 150 times the CPU time). A two stage simulation method was implemented. First the direct method was attempted and if this failed TBM was used. To save time lazy ABC was implemented with multiple stopping decisions, the first taking place after attempting the direct method. This has a binary decision statistic indicating success or failure. The second stopping decision is as described earlier. Tuning was performed as described in Appendix A.1.2, using g fitted as described

above by either the standard or conservative tuning method. The standard method used ϵ_1 to give 30 acceptances in the pilot run. As before all analyses reused the same random seeds.

Finally an importance sampling analysis was performed on the larger dataset. A sample of 10^4 log parameter values was taken from simulations of the preceding standard ABC analysis with distances below the 0.3 quantile. A Gaussian mixture distribution was constructed with locations given by this sample and variances equal to twice the empirical variance of the sample. After truncation to the prior support, this was used to give $g(\theta)$, where θ now represents the log parameters. This choice follows the suggestions of Beaumont et al. (2009), noting that using the log scale produced a better fit to the sample and that the subsample was used to avoid slow density calculations. The preceding $D = 35$ analysis was then repeated.

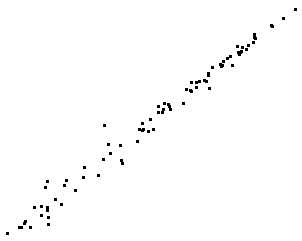


Figure 1: Details of a simulation study applying lazy ABC to spatial extremes corresponding to the first row of Table 1. Panels A-C concentrate on the standard tuning approach. Panel A Pilot run values of $\hat{d}$ and d. The dashed line shows the value of . Panel B Estimated efficiency for different values of . The dashed line shows the realised efficiency. Panel C Estimated efficiency for the best choices of L of various lengths output by backwards selection. The dashed line shows the realised efficiency. Panel D Values of $\hat{d}$ and from non-pilot simulations under standard (solid line) and conservative (dashed line) tuning. The marks on the horizontal axis indicate the simulations which resulted in positive weights. (For this panel conservative tuning was performed using as selected by standard tuning.)

		Standard	Lazy			Relative e ciency	
Range	Smooth	Time (10^3s)	Time (10^3s)	Sample size	ESS	Estimated	Actual
0.5	1	32.0	8.0 (11.6)	196 (199)	196.0 (198.7)	4.08 (3.28)	4.00 (2.79)
1	1	31.3	7.3 (9.8)	200 (200)	199.9 (200.0)	4.34 (4.31)	4.35 (3.25)
1	3	31.3	8.2 (11.2)	194 (198)	182.5 (196.5)	3.77 (3.43)	3.51 (2.79)
3	1	31.2	7.7 (11.1)	194 (200)	189.9 (200.0)	4.18 (3.56)	3.89 (2.86)
3	3	31.2	7.4 (11.0)	192 (199)	175.8 (199.0)	4.43 (3.65)	3.79 (2.87)
5	3	31.3	8.3 (11.1)	200 (200)	200.0 (200.0)	3.73 (3.49)	3.85 (2.87)

		Standard			Lazy			Relative
		Time (10^3s)	Sample size	ESS	Time (10^3s)	Sample size	ESS	e ciency
RS standard	2.61	241.4	210	210	22.8	200	139.6	7.0
RS conservative	2.61	241.4	207	207	25.5	200	197.7	9.0
IS standard	2.33	136.4	209	168	61.9	200	165	2.2
IS conservative	2.33	136.4	209	168	51.0	200	162	2.6

Table 2: Simulation study on a spatial extremes dataset with $D = 35$. Results are shown for rejection and importance sampling with standard and conservative tuning. The rejection sampling output was used to create the importance density. The nal choice of is shown. For IS the two values are equal but there is a small di erence for RS. The lazy ABC output includes the pilot run and the tuning time.

and weighting the accepted simulations accordingly, the algorithm targets exactly the same distribution as standard ABC, in the sense that Monte Carlo estimates of functions $h(\ )$ and of the model evidence converge to unchanged values.

Results have been provided on the optimal tuning of the lazy ABC stopping rule and used to motivate a practical tuning method. This has been demonstrated for a computationally challenging application where it has produced improvements in e ciency (ESS/CPU time) over standard ABC of up to 8 times. One case of this application involved multiple stopping decisions. This illustrated two potential uses of lazy ABC: rstly to consider stopping every simulation based on whether it appears promising, secondly to consider stopping after particular events which are suspected a priori to indicate unpromising results.

The tuning method is based on estimating the optimal choice of $(\ )$, (6). The most di cult part was estimating $(\ ) = \Pr(\ d(S(Y); S(y_{obs})) \quad j\ )$ from pilot run data. Two approaches to this were described, a standard approach of direct estimation and a conservative approach of estimation using a larger value than is of interest for ABC. The latter approach improves robustness and make estimation simpler at the cost of some ine ciency. Both approaches performed well in the simulation studies but some improvements are desirable. Firstly, estimation of $(\ )$ involves extrapolation which may produce inaccurate results. Secondly, several choices by the user are required, especially for the standard approach. A more automated approach would be useful for lazy versions of ABC SMC algorithms, where a new choice of would be needed for each value, or alternatively for lazy ABC algo-

rithms which adapt as more simulations become available. It would be of interest to find suboptimal but robust choices of addressing these issues.

Lazy ABC with multiple stopping decisions is an extension to the framework of the main paper and is described in Appendix A. A tuning method is given when the decision statistics for all stopping decisions are discrete, and also some cases where one decision statistic is continuous. For more complex cases tuning results are not available. For now it is recommended to discretise most decision statistics to avoid this difficulty.

Also, Section 3.1 showed that a generalisation to the non-ABC setting, lazy importance sampling, is possible, and the theoretical results of the paper carry over to this. However exploratory analysis suggests tuning this in practice is more challenging than lazy ABC. This is because it is necessary to estimate $\gamma(\theta) = \mathrm{E}(\hat{L}^2 | \theta)$ (see Remark 2), and this expectation can be strongly influenced by the upper tail of $\hat{L} | \theta$ which is hard to estimate from pilot run output. For lazy ABC, $\hat{L}_{\mathrm{ABC}} | \theta$ is Bernoulli avoiding this difficulty. A related point is that lazy ABC can be generalised to allow a non-uniform ABC kernel. This gives $\hat{L}_{\mathrm{ABC}}$ with a known upper bound so that estimation of $\gamma(\theta)$ seems feasible.

This paper has concentrated on importance sampling, which is widely used by ABC practitioners, but the lazy ABC approach can be extended to ABC versions of MCMC and SMC, which are more efficient algorithms. The tuning results are applicable to SMC algorithms, but further practical methods are needed, as mentioned above. Further theory on optimal tuning is necessary for MCMC, although good performance may be possible with ad-hoc tuning. Examining the connections between lazy ABC and rejection control (Liu et al., 1998) may also be fruitful, especially to design algorithms in which partial

A Multiple stopping decisions

The lazy ABC framework of Section 3 allows multiple stopping decisions, as follows. As in that section assume Y is a deterministic transformation of a latent vector $X_{1:p}$.

Example A1: Multiple stopping decisions. Let $X = X_{1:p}$ and $\pi(\theta; x) = \prod_{i=1}^{s} \pi^{(i)}(\theta; x_{1:t_i})$. Thus, for each $1 \leq i \leq s$, once simulation of $X_{1:t_i}$ has been performed then $\hat{L}_{lazy}$ is set to zero with a certain probability, in which case no further simulation is necessary. It is often be useful to let $\pi^{(i)}(\theta; x_{1:t_i}) = \pi^{(i)}(\psi_i(\theta; x_{1:t_i}))$. That is, each stopping decision has associated decision statistic ψ_i.

Example A2: Multiple random stopping times. As for Example A1 but with each t_i replaced with a random stopping time variable T_i. This permits stopping to be considered when various random events occur, without imposing a fixed order of occurrence.

The following alternative characterisation of these examples is useful below.

Lemma 1. For any $1 \leq i \leq s$, Examples A1 and A2 can be represented as a lazy importance sampling algorithm with continuation probability $\pi^{(i)}(\psi_i)$ and

$$\hat{L} = \begin{cases} \hat{L}_{ABC} / \psi_i(\theta; X) & \text{with probability } \psi_i(\theta; X) \\ 0 & \text{otherwise,} \end{cases}$$

where $\psi_i(\theta; x) = \prod_{j \leq i} \pi^{(j)}(\psi_j)$.

Proof. The likelihood estimator stated can easily be verified to have the same distribution as $\hat{L}_{lazy}$. □

It is also helpful to define $T_{2i}(\theta; \psi_{1:s})$ as the expected time remaining from the calculation of ψ_i until the likelihood estimate is computed conditional on θ and $\psi_{1:s}$, and $T_{2i}(\psi_i)$

A.1 Tuning

The e ciency estimate of Section 4.2.3 can be used in a multiple stopping decision setting given a choice of . It is necessary to update the estimator given there which is usually a straightforward task. Sections A.1.1 and A.1.2 describe situations of practical interest where the optimal form of can be derived. However in general the problem is challenging, as illustrated by Section A.1.3.

A.1.1 Discrete decision statistics

Suppose (; x.e[(h.676.9552 Tf 7..63505(=21 0 Td)-167(x.e[(.742 (6.e[9t2 Tf 7Q 0 Td)-TJ 27(hallengi

A.1.3 Multiple continuous decision statistics

Consider the setting of A.1.2 with the modification that every $\psi_i(\cdot;x)$ is continuous and there exists a corresponding function $u_i(\cdot)$ such that $\psi_i(\cdot;x) = u_i(\cdot)/g(\cdot)$. The same approach as above gives equations of the form

$$\phi^{(i)}(\cdot) = \min\left(1;\ \lambda_i u_i(\cdot)\ \frac{\psi_i(\cdot)}{T_{2i}(\cdot)}^{1/2}\right);$$

for $i = 1,\ldots,s$. The definition of ψ_i involves $\phi^{(j)}$ for all $j \neq i$, and T_{2i} will also involve many of these terms. Thus deriving the optimal $\phi^{(i)}$ functions involves solving a complicated system of non-linear implicit equations.

B Tuning proofs

All results are proved for the general case of LIS as described in Section 3.1. For lazy ABC replace $\hat{L}$ with $\hat{L}_{ABC}$.

B.1 Proof of Theorem 2

In LIS the importance sampling weight W equals $\frac{\hat{L}(\cdot)}{\psi(\cdot)g(\cdot)}$ with probability $\psi(\cdot)$ and zero otherwise. Hence:

$$E(W^2) = \int \frac{E[\hat{L}^2 | \cdot;\cdot;y]\ \psi(\cdot)^2}{\psi(\cdot)g(\cdot)^2}\ \pi(\cdot;y|\cdot)g(\cdot)\,d\cdot\,d\cdot\,dy = \int \frac{\gamma(\cdot)}{\psi(\cdot)}g(\cdot)\,d\cdot; \qquad (9)$$

where $\gamma(\cdot) = E\left(\hat{L}^2\ \frac{\psi(\cdot)}{g(\cdot)}^2\right)$ (which equals $E\left(\hat{L}_{ABC}\ \frac{\psi(\cdot)}{g(\cdot)}^2\right)$ in the ABC case) and $g(\cdot) = \int \pi(\cdot|y)g(\cdot)\,d\cdot$.

The expected time of a single iteration of the LIS algorithm is

$$E(T)/N = T_1 + \int\int \psi(\cdot)T_2(\cdot;\cdot)\,\pi(\cdot|y)g(\cdot)\,d\cdot\,d\cdot = T_1 + \int \psi(\cdot)T_2(\cdot)g(\cdot)\,d\cdot: \qquad (10)$$

Note that $E(W)$ is a constant, so choosing $(\ )$ to maximise the expression for asymptotic e ciency in Section 4.1 is equivalent to minimising $E(W^2)E(T)/N$. Call this problem P. Consider also the problems $P(\ )$, minimising $E(W^2)$ under the constraint $E(T)/N =$ and $P(\ ;\ )$, minimising $E(W^2) + [E(T)/N \quad]$, or equivalently

$$Z \quad \frac{(\ )}{(\ )}$$

solution:

$$( )g( ) = \quad ( ) \frac{( )}{T( )}^{1=2} : \tag{13}$$

(Recall the LIS de nition of $( )$ from Remark 3 and note that the lazy ABC de nition in Corollary 1 can be derived from this.) Various choices of, such as 1, give a solution which also meets the constraint on . These all give algorithms which are equivalent to RW-IS with $g( ) / ( ) \frac{( )}{T( )}^{1=2}$ as claimed.

References

Andrieu, C. and Roberts, G. O. (2009). The pseudo-marginal approach for e cient Monte Carlo computations. The Annals of Statistics, pages 697{725.

Beaumont, M. A. (2003). Estimation of population growth or decline in genetically monitored populations. Genetics, 164(3):1139{1160.

Beaumont, M. A. (2010). Approximate Bayesian computation in evolution and ecology. Annual Review of Ecology, Evolution and Systematics, 41:379{406.

Beaumont, M. A., Cornuet, J. M., Marin, J.-M., and Robert, C. P. (2009). Adaptive approximate Bayesian computation. Biometrika, 96(4):983{990.

Buzbas, E. O. and Rosenberg, N. A. (2013). AABC: approximate approximate Bayesian computation when simulating a large number of data sets is computationally infeasible. Preprint. Available at http://www.arxiv.org/abs/1301.6282 .

Erhardt, R. J. and Smith, R. L. (2012). Approximate Bayesian computing for spatial extremes. Computational Statistics & Data Analysis, 56(6):1468{1481.

Fearnhead, P., Papaspiliopoulos, O., Roberts, G. O., and Stuart, A. (2010). Random-weight particle ltering of continuous time processes. Journal of the Royal Statistical Society: Series B, 72(4):497{512.

Fearnhead, P. and Prangle, D. (2012). Constructing summary statistics for approximate Bayesian computation: Semi-automatic ABC. Journal of the Royal Statistical Society, Series B, 74:419{474.

Geweke, J. (1989). Bayesian inference in econometric models using Monte Carlo integration. Econometrica, 57(6):1317{1339.

Liu, J. S. (1996). Metropolized independent sampling with comparisons to rejection sampling and importance sampling. Statistics and Computing, 6(2):113{119.

Liu, J. S., Chen, R., and Wong, W. H. (1998). Rejection control and sequential importance sampling. Journal of the American Statistical Association, 93(443):1022{1031.

Marin, J.-M., Pudlo, P., Robert, C. P., and Ryder, R. J. (2012). Approximate Bayesian computational methods. Statistics and Computing, 22(6):1167{1180.

Meeds, E. and Welling, M. (2014). GPS-ABC: Gaussian process surrogate approximate Bayesian computation. Preprint. Available at http:/[(.)-434(Av)54(ailable)-Mt051:135326(at)]v.c4appr

Wood, S. (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. Journal of the Royal Statistical Society: Series B, 73(1):3{36.