AMSI VACATIONRESEARCH SCHOLARSHIPS 2020–21

Get a Thirst for Research this Summer

Adaptive tolerance selection for sequential Monte Carlo - approximate Bayesian computation (SMC-ABC) replenishment algorithm

Abhishek Varghese

Supervised by Christopher Drovandi and Mitchell O'Sullivan

Queensland University of Technology

Vacation Research Scholarships are funded jointly by the Department of Education, Skills and Employment and the Australian Mathematical Sciences Institute.

Contents

1 Abstract

Sequential Monte Carlo approaches to approximate Bayesian computation (SMC-ABC) offer an efficient means to estimate posterior distributions in likelihood-free scenarios. Tolerance sequencing in SMC-ABC is critical to the efficiency of the algorithm. We consider a method for adaptive tolerance selection for another particle filter approach to ABC (PMC-ABC), which considers the gained distance between distributions between sequences. We implement this to SMC-ABC, and reduce the computational cost of implementing this approach. We utilise examples to demonstrate the computational efficiency improvements achieved.

2 Statement of Authorship

The workload was divided as follows:

- Abhishek Varghese implemented the SMC ABC algorithm (and other ABC algorithms); ran testing; interpreted and reported results; and wrote the report.
- Chris Drovandi developed the SMC ABC algorithm used and supervised the work.
- Mitchell O'Sullivan helped adapt the SMC-ABC replenishment algorithm, proofread and tested code, and supervised the work.

3 Introduction

Since the advent of the 21st century, models have increased in complexity as our understanding of natural and physiological processes has deepened, and the availability of computational resources and large datasets have increased.

These complex models present significant challenges for parameter inference, as it may be intractable to evaluate the likelihood function. Approximate Bayesian computation (ABC) is a likelihood-free inference technique that enables parameter estimation and uncertainty quantification in scenarios where the likelihood is intractable[1]. ABC bypasses the likelihood function to generate approximate posterior distributions of parameters of interest for models[2]. It has been used for a wide variety of applications, including network data, disease epidemiology, medical imaging, ecology, cell biology, climate extremes and airport design[3].

Among the many approaches to ABC, particle filter methods such as Population Monte Carlo (PMC) and Sequential Monte Carlo (SMC) are found to be effective at approximating the posterior distributions of model parameters while leveraging computational resources economically [1]. Both methods iteratively approach the posterior from the prior through a sequence of target distributions. At each iteration, these methods filter a proportion of samples of the current distribution by a tolerance measure (α), and jitter particles through a perturbation kernel to ensure *iid* (independent and identically distributed) samples.

In SMC-ABC, particles that do not meet the tolerance threshold are discarded and resampled from the remaining particles to maintain the same population size [4]. A well-accepted classical SMC-ABC algorithm by Drovandi and Pettit [1] set the tolerance threshold to be a proportion of the maximum particle discrepancy at each iteration as the algorithm progresses.

This approach opens the possibility for the algorithm to keep poor proposals initially, and discard good proposals as the algorithm approaches the posterior distribution. It is also important to note that resampling is generally cheap when the algorithm starts, and becomes more expensive as the algorithm progresses - which can introduce further computational inefficiencies.

Simola et. al [5] have recently proposed an adaptive tolerance selection approach to identify an optimal tolerance for each iteration in an ABC-PMC algorithm, by leveraging the density ratio between current and target distributions at each iteration. We adapt this approach to the replenishment algorithm in SMC-ABC and further reduce the computational expense of this approach. We test our adaptive SMC algorithm on two toy model scenarios, and evaluate the performance improvements achieved through this approach.

4 Method

4.1 Approximate Bayesian computation

Model parameters ϕ may be inferred by its posterior density $p(\phi \mid y)$ given the observed dataset y . The posterior density can be written by Bayes' theorem as,

$$p(\phi \mid y) = \frac{\pi(\phi)\ p(y|\phi)}{m(y)} \tag{1}$$

where $\pi(\phi)$, $p(\phi \mid y)$ and $m(x) = \int \pi(\phi)\ p(y|\phi)\, d\phi$ are, correspondingly, the prior density on the parameter ϕ , the likelihood function, and the marginal likelihood. The prior density $\pi(\phi)$ enables a way to leverage the learning of parameters from prior knowledge.

ABC bypasses the evaluation of the likelihood function by instead simulating data from the model to generate an approximate posterior distribution. Due to the high dimensionality of the observed data, y , the data set is often reduced to a set of summary statistics, $S(y)$ [6]. Thus, ABC targets the posterior conditional on the summary statistics:

$$p(\phi|S(y)) \propto (S(y)|\phi)\ \pi(\phi) \tag{2}$$

However, this too requires the evaluation of a typically intractable likelihood, $p(S(y)|\phi)$ [6]. Therefore, ABC approximates this intractable likelihood through the following integral:

$$p_\epsilon(S(y)|\phi = \int_y K_\epsilon(\rho(S(x), S(y)))p(x|\phi).dx \tag{3}$$

where $\rho(S(x), S(y))$ is a discrepancy function that compares the simulated and observed summary statistics, and $K_\epsilon(\cdot)$ is a kernel weighting function with bandwidth ϵ that weights simulated summaries in accordance with their closeness to the observed summary statistic [2]. While the integral in (3) is analytically intractable, it may be estimated by taking n iid simulations from the model $x_{i\,i=1}^{n} \sim p(x|\phi)$, evaluating their corresponding summary statistics $S_{i\,i=1}^{n}$ where $S_i = S(x_i)$, and calculating the following ABC likelihood:

$$p_\epsilon(S(y)|\phi) \approx \frac{1}{n}\sum_{i=1}^{n} K_\epsilon(\rho(S_i, S(y))) \tag{4}$$

The unbiased likelihood estimator described in (4) is generally sufficient to obtain a Bayesian algorithm that targets the posterior distribution $p_\epsilon(\phi|S(y)) \propto p_\epsilon(S(y)|\phi)p(\phi)$. The summary statistics, $S(\cdot)$, discrepancy measure, $\rho(\cdot,\cdot)$, and tolerance value, ϵ, utilised in the ABC method introduce approximation errors to the target posterior distribution. In order to minimise these errors, these factors must be chosen and tuned carefully to maximise accuracy while ensuring a computationally feasible operation [6].

The most basic implementation of ABC is known as rejection sampling [2]. In this algorithm, the parameter is estimated by generating model realisations x corresponding to different parameter values ϕ promoted from the prior. The summaries $S(x)$ are computed and compared to $S(y)$ through the discrepancy measure $\rho(\cdot, \cdot)$. If the discrepancy between the simulated and observed summaries is lower than the tolerance, ϵ , then the corresponding ϕ is accepted as part of the approximate posterior distribution.

The pseudo-code for an ABC rejection sampling scheme is provided below [2]:

for $i \in 1 : n$ do

Draw $\phi \sim \pi(\phi)$

Draw $x \sim p(\cdot|\phi)$

Accept ϕ if $\rho(S(x), S(y)) \leq \epsilon$

end for

where n is the number of iid samples to be taken from the prior $\pi(\phi)$

4.1.1 Likelihood-free Sequential Monte Carlo (SMC-ABC)

In SMC-ABC, the following sequence of distributions is defined:

$$\pi_t(\theta, x|y, \epsilon) \propto f(x|\theta)\pi(\theta)1(\rho(x, y) \leq \epsilon_t), \tag{5}$$

for $t = 1, \ldots$,T, with a non-increasing set of tolerances $\epsilon_1 \geq \epsilon_2 \geq \cdots \geq \epsilon_T$. The algorithm traverses a set of N particles through a sequence of target distributions by iteratively applying re-weighting, re-sampling and move steps to each particle [1].

At each iteration, particles are sorted via the discrepancy measure and a proportion of the particles with the largest distance are dropped (say $100 \cdot \alpha\%$). Then, new particles are resampled from the remaining particles to replenish the entire population. Finally, resampled particles are moved according to an MCMC kernel using the target discrepancy of the iteration, to ensure particle uniqueness in the population [1].

A step-by-step description of the SMC-ABC algorithm by Drovandi and Pettit [1] is provided below:

1. Set N_a as the integer part of αN

2. Draw N particles from prior and compute distance function for each particle. This produces a set of particles $\{\theta^i, \rho^i\}_{i=1}^N$

3. Sort the particle set by ρ, so that $\rho^1 \leq \rho^2 \leq \cdots \leq \rho^N$, and set $\epsilon_t = \rho^{N-N_a}$ and $\epsilon_{\max} = \rho^N$. If $\epsilon_{\max} \leq \epsilon_T$ then finish, otherwise go to next step

4. Compute the tuning parameters of the MCMC kernel $q_t(\cdot|\cdot)$ using the particle set $\{\theta^i\}_{i=1}^{N-N_a}$

5. Resample $\{\theta^j\}_{j=N-N_a+1}^N$ from the kept samples $\{\theta^i\}_{i=1}^{N_a}$ and copy over corresponding discrepancy values

6. Perform S trial MCMC iterations on the $j = N - N_a + 1$ to N resampled particles using q_t and ϵ_t. Record the acceptance of this as p_t.

7. Compute $R_t = \lceil \log(c)/\log(1-p_t) \rceil$. This is the estimated number of MCMC steps required for the iteration.

8. Perform remaining $R_t - S$ trial MCMC iterations on the $j = N - N_a + 1$ to N resampled particles using q_t and ϵ_t. Record the overall acceptance rate of all R_t MCMC iterations as p_t. If p_t is too small then stop the algorithm. Set $S = \lfloor R_t/2 \rfloor$ and go to step 3.

4.2 Adaptive tolerance selection for SMC-ABC replenishment algorithm

We focus our efforts on adaptively selecting an optimal α for each iteration, to improve the computational efficiency of the SMC-ABC replenishment algorithm.

Simola et. al [5] tackle a similar problem for 'population Monte Carlo' (PMC) ABC. They propose to use the estimated ABC posteriors to select a quantile to update the tolerance for the next iteration, and adjust the next tolerance based on how slowly or rapidly the sequential ABC posteriors are changing. They compute the following density ratio for each iteration $t > 1$:

$$\begin{aligned} R(\theta) &= \frac{\hat{\pi}_{\epsilon_t}(\theta)}{\hat{\pi}_{\epsilon_{t-1}}(\theta)} \\ c_t &= \sup_\theta R(\theta) \end{aligned} \tag{6}$$

Simola et. al [5] use proper densities for $\hat{\pi}_{\epsilon_t}(\theta)$ and $\hat{\pi}_{\epsilon_{t-1}}(\theta)$ such that $c_t = 1$ when the two densities are exactly the same, or there must be a point where $\hat{\pi}_{\epsilon_t}(\theta) > \hat{\pi}_{\epsilon_{t-1}}(\theta)$, resulting in $c_t < 1$.

They use c_t to inform the optimal 'quantile to keep' for each iteration as follows:

$$q_t = \frac{1}{c_t} \tag{7}$$

Under this methodology, Simola et. al [5] assert that small values of q_t imply that q_{t-1} led to a large improvement between $\hat{\pi}_{\epsilon_t}(\theta)$ and $\hat{\pi}_{\epsilon_{t-1}}(\theta)$, which subsequently results in a larger tolerance reduction for the next iteration. Conversely, q_t tends to 1 as $\hat{\pi}_{\epsilon_t}(\theta)$ and $\hat{\pi}_{\epsilon_{t-1}}(\theta)$ become more similar, and as the ABC posterior stabilises. The supremum in (6) is calculated using an optimiser over the parameter space.

The optimisation procedure utilised by Simola et. al [5] to calculate the supremum of $\hat{c}_t$ can be computationally expensive. Instead, we reduce the computational expense of this approach by approximating $\hat{c}_t$. Firstly, we approximate $R(\theta)$ by evaluating using an unconstrained Least-Squares Importance Fitting (uLSIF) algorithm [7]. The uLSIF algorithm provide the approximate density ratio at samples of $\hat{\pi}_{\epsilon_{t-1}}(\theta)$, such that:

$$R^* = \{\hat{R}(x) |\, x \in \{\theta_i^t\}_{i=1}^N\} \tag{8}$$

We may use this to evaluate an approximation of c_t as follows:

$$\hat{c}_t = \arg\max R^* \tag{9}$$

We believe this is a reasonable approach as it can be expressed that:

$$P(c_t \neq \hat{c}_t) \to 0 \text{ as } N \to \infty \tag{10}$$

Thus, we may leverage $\hat{c}_t$ to inform the tolerance threshold (α) in the SMC-ABC replenishment algorithm at each iteration:

$$\alpha_{t+1} = 1 - \frac{1}{\hat{c}_t} \tag{11}$$

4.3 Examples

This methodology was studied using several example model simulation scenarios to evaluate the performance of this adaptive SMC-ABC approach (aSMC-ABC). A brief summary is provided for each model utilised in the analysis.

4.3.1 Univariate g-and-k distribution

The g-and-k distribution is a quantile distribution which is defined by its inverse cumulative distribution function (cdf). These functions generalise the quantile functions for standard distributions, enabling the creation of

a variety of probability distributions. The likelihood of these functions is expensive to calculate, due to an analytically intractable cdf. Conversely, simulation from these distributions is computationally cheap. Therefore, estimating the parameters for these set of functions is an amenable task for ABC [8].

The g-and-k distribution is governed by five parameters $\theta = (a, b, c, g, k)$, and is given by:

$$Q(p; \boldsymbol{\theta}) = a + b\left(1 + c\frac{1 - \exp(-gz(p))}{1 + \exp(-gz(p))}\right)\left(1 + z(p)^2\right)^k z(p) \tag{12}$$

It must be noted that c is held constant at 0.8 [8]. We use the robust summary statistics developed by Drovandi and Pettit [8] for our testing, as they have been found to be effective at estimating parameters for this model. Furthermore, we use a random walk multivariate Normal proposal with its parameters estimated adaptively based on the particles satisfying the current target through the evaluation of a covariance matrix.

4.3.2 Banana Bunchy Top Virus (BBTV) network-based epidemiological model

Varghese et al. [9] develop a network-based epidemiological model to model the spread of BBTV in a plantation. The model simulates the probability of a node being infected in time t in months from $t \in [1 \cdots T]$. The model is governed by six parameters $\theta = (\theta_{00},\ \theta_{01},\ \theta_{01},\ \theta_{10},\ \theta_{11},\ \theta_{12})$. Where:

- $\theta_{(0\cdot)}$ denote the probability of node recovery.
- $\theta_{(1\cdot)}$ denote the probability of a node infecting a neighbouring node.
- $\theta_{(2\cdot)}$ denote the probability of a node infecting a non-neighbouring node.

These three parameters are duplicated for summer $(\theta_{(\cdot 0)})$ and winter $(\theta_{(\cdot 1)})$, giving six in total. We use the summary statistics developed by Varghese et. al [9] and apply the same adaptive multivariate normal random walk proposal for the MCMC kernel in the aSMC-ABC algorithm.

5 Results and Discussion

We tested the performance of the aSMC-ABC algorithm against the SMC-ABC algorithm developed by Drovandi and Pettit [8], by estimating the parameters for the two models mentioned in the previous section.

We generate a dataset with known parameter values, and treat this as our observed data for each model. We run both ABC algorithms on the same observed data, and estimate the posterior distributions of the parameters. This procedure enables a fair platform for benchmarking both algorithms.

Our results demonstrate that both the benchmark SMC-ABC algorithm [8], and the aSMC-ABC algorithm developed in this paper generally achieve the same posterior accuracy for both models. Therefore, these results are not covered in this paper.

Algorithm performance is measured by the number of model simulations required by each algorithm to achieve the target discrepancy, where the algorithm with the least model simulations is considered to be most efficient. These results are provided in detail for each model in the following sections.

The results indicate that there is no perceivable improvement in required model simulations compared to the benchmark SMC-ABC algorithm [8].

5.1 Parameter estimation of univariate g-and-k distribution

We create three different sets of observed data for the univariate g-and-k distribution, and estimate the parameters for each 'dummy' dataset three times.
The results for the SMC-ABC algorithm (seen in [8]) are provided in Table 1, and the results for aSMC-ABC algorithm developed in this paper are in Table 2.

Parameter Set	Trial #1	Trial #2	Trial #3	Mean Model Sims
1	1004260	1123971	1051168	1059800
2	986090	776656	876814	879853
3	807568	814246	853303	825039

Table 1: Model simulations required by SMC-ABC algorithm (seen in [8]) to estimate parameters for g-and-k distribution. Three 'true' parameter sets were chosen at random, and parameters were estimated with each ABC algorithm for three trial runs.

Parameter Set	Trial #1	Trial #2	Trial #3	Mean Model Sims
1	907930	741466	861016	836804
2	946618	924025	807304	892649
3	886190	685879	838953	803674

Table 2: Model simulations required by the aSMC-ABC algorithm to estimate parameters for g-and-k distribution.

It may be noted that the aSMC-ABC algorithm required less model simulations on average for two out of

the three parameter sets. The aSMC-ABC algorithm was most efficient for parameter set 1, with average model simulation savings of 20% compared to the SMC-ABC algorithm by [8]. Modest savings are observed in parameter set 3, with an efficiency improvement of 2.5% over the SMC-ABC algorithm by [8]. Finally, the aSMC-ABC algorithm performs 1.5% worse that the benchmark algorithm in parameter set 2.

5.2 Parameter estimation of BBTV network-based epidemiological model

We create one set of observed data for the BBTV model, and estimate the parameters for this 'dummy' dataset three times for each ABC algorithm.

The results for both algorithms are provided in Table 3.

	Trial #1	Trial #2	Trial #3
aSMC-ABC	253966	8953*	38080*
SMC-ABC	232519	9262*	36620*

Table 3: Model simulations required by each ABC algorithm over 3 trials.
*ABC runs were stopped early due to an auxillary stopping rule.

It is hard to observe any perceivable improvement by the aSMC-ABC algorithm compared to the benchmark SMC-ABC algorithm by [8] for the BBTV model parameter estimation scenario. While aSMC-ABC required 8.6% more simulations than the benchmark SMC-ABC algorithm in trial 1, it performed comparably on the other two trials.

6 Future Research

The results demonstrate that our aSMC-ABC algorithm performance is inconsistent, with minimal efficiency improvements overall. It is possible that the inconsistency in algorithm performance is induced by the approximation errors we create to improve computational efficiency. We recommend further research in exploring adaptive resampling strategies for the SMC-ABC algorithm seen in [8].

7 Conclusion

We implement an adaptive resampling approach to SMC-ABC, based on existing work that leverages the density ratio between current and target distributions to identify an optimal quantile threshold for ABC-PMC. Despite significantly reducing the computational expense of implementing this approach, minimal performance improvements are achieved through this method. We recommend deeper study on adaptive resampling strategies for SMC-ABC, and recommend this adaptive SMC-ABC algorithm as a starting point for future research.

8 Acknowledgements

- Supervisors: Professor Chris Drovandi and Mitchell O'Sullivan (PhD student)
- Queensland University of Technology (QUT)
- Queensland University of Technology High Performance Computing (QUT HPC)
- Australian Mathematical Sciences Institute (AMSI)

References

[1] C. C. Drovandi and A. N. Pettitt. Estimation of parameters for macroparasite population evolution using approximate bayesian computation. *Biometrics*, 67(1):225–233, 2011.

[2] S. A. Sisson. *Handbook of Approximate Bayesian Computation*. Chapman and Hall/CRC Handbooks of Modern Statistical Methods Ser. CRC Press LLC, Milton, 2018.

[3] Jarno Lintusaari, Michael U. Gutmann, Ritabrata Dutta, Samuel Kaski, and Jukka Corander. Fundamentals and Recent Developments in Approximate Bayesian Computation. *Systematic Biology*, 66(1):e66–e82, 09 2016.

[4] S. A. Sisson, Y. Fan, and Mark M. Tanaka. Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 104(6):1760–1765, 2007.

[5] Umberto Simola, Jessi Cisewski-Kehe, Michael U. Gutmann, and Jukka Corander. Adaptive approximate bayesian computation tolerance selection. *Bayesian Anal.*, 2021. Advance publication.

[6] Prangle Dennis. *Summary Statistics*, book section Summary Statistics. CRC Press, 2018.

[7] Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. A least-squares approach to direct importance estimation. *The Journal of Machine Learning Research*, 10:1391–1445, 2009.

[8] Christopher C. Drovandi and Anthony N. Pettitt. Likelihood-free bayesian estimation of multivariate quantile distributions. *Computational Statistics & Data Analysis*, 55(9):2541–2556, 2011.

[9] Abhishek Varghese, Christopher Drovandi, Antonietta Mira, and Kerrie Mengersen. Estimating a novel stochastic model for within-field disease dynamics of banana bunchy top virus via approximate Bayesian computation. *PLOS Computational Biology*, 16(5):1–23, 2020. Publisher: Public Library of Science.