**2. Method**

Intro to the chapter…

**2.1. Calibration Methods Compared**

We compare the following calibration methods; Rejection Approximate Bayesian Computation (Rejection ABC), Sequential Approximate Bayesian Computation (Sequential ABC) and Bayesian Maximum Likelihood estimation (BMLE). In the next sections, we will shortly explain each of these.

**2.1.1. Rejection ABC**

Rejection ABC is the most basic form of ABC. This method operates by sampling parameter values () from the prior distributionand given these sampled parameter values, data (*y*) is simulated under a model. A summary statistic () of the simulated data () must satisfy a proximity criterion with the target statistic () of the observed data (*x*) such that, where *d* expresses the distance between the target and summary statistics *t* and *s*, and *ϵ* represents a tolerance level. A target statistic is a data point from the observed data to be considered during the simulation procedure before a decision is taken, as to whether a certain parameter combination is to be retained or discarded. The algorithm retains sampled parameter values for which the model produces simulated summaries (*s*) that are closer to the target statistic (t) than the tolerance (ϵ) (Sunnaker). *Figure 1* illustrates how the rejection ABC algorithm functions. The simulator (M) is run each time with a newly sampled parameter value (θ) from the prior distribution obtaining a simulated summary statistic. When the distance between a summary statistic and the target statistic () is smaller than (ϵ); the parameter value is retained (red dot). Otherwise, it is discarded.

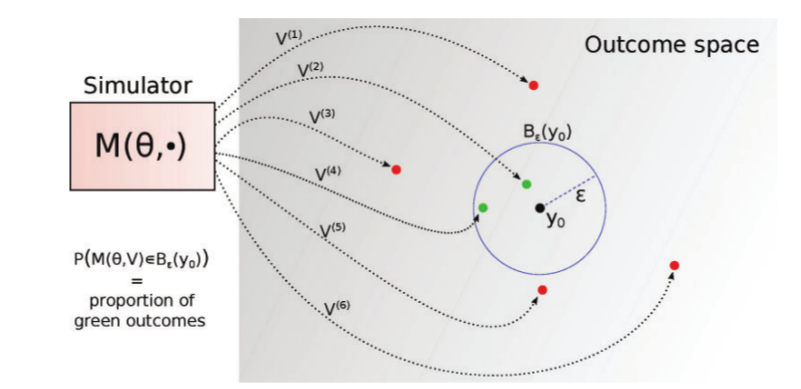


Figure 1 : Rejection ABC(Lintusaari)

From the Bayesian framework, estimation of the posterior distribution depends on the prior distribution and the likelihood. The posterior is defined as

Since the denominator , which is the marginal probability of the data does not depend on , the posterior can be expressed as proportional to the numerator (equation …)

|  |  |  |
| --- | --- | --- |
|  |  |  |

Where is a vector of parameter values, *x* is the observed data, is the posterior distribution, is the likelihood and is the prior distribution. ABC techniques use this same knowledge in the approximation of the posterior. The distribution of the retained parameter values is expected to converge to the posterior distribution for arbitrarily small values of the tolerance (ϵ) without the explicit calculation of the likelihood, such that

**2.1.2. Sequential ABC**

Sequential ABC is a class of ABC methods that approximates the posterior progressively by drawing samples from the prior sequentially (Lenormand). The prior for a particular sampling step depends on the previous retained sample except for the first sampling step which draws from the prior parameter space provided. Thus, the tolerance of the initial sampling step is less restrictive compared to the subsequent ones (Trevelyan). The sample at the current sampling step () is derived from the previous sample () using a decreasing sequence of tolerance levels. These methods determine by themselves the tolerance level used at each sampling step and provide a stopping criterion. This choice of tolerance for the current sampling step is determined as a function of the distances simulated in the previous sampling step (McKinley). Figure 2 gives an illustration of how the sequential ABC algorithm works. The first step of sequential ABC is the same as running rejection ABC; simulator (M) is run with parameter values () sampled from the prior distribution with tolerance obtaining a simulated sample and retained parameter values (). In the second step, tolerance is decreased compared to tolerance and the prior is determined by the retained parameter combinations () in the first step. A second sample of parameter values is obtained at tolerance and this process if repeated until a stopping criterion is reached. At each sampling step, a decision is made whether to retain a particular parameter value or discard it. If a simulated summary statistic at that step is further from the target statistic () than the tolerance level () of that step, that particular parameter value is discarded, otherwise it is retained. The final sample approximates the posterior distribution.

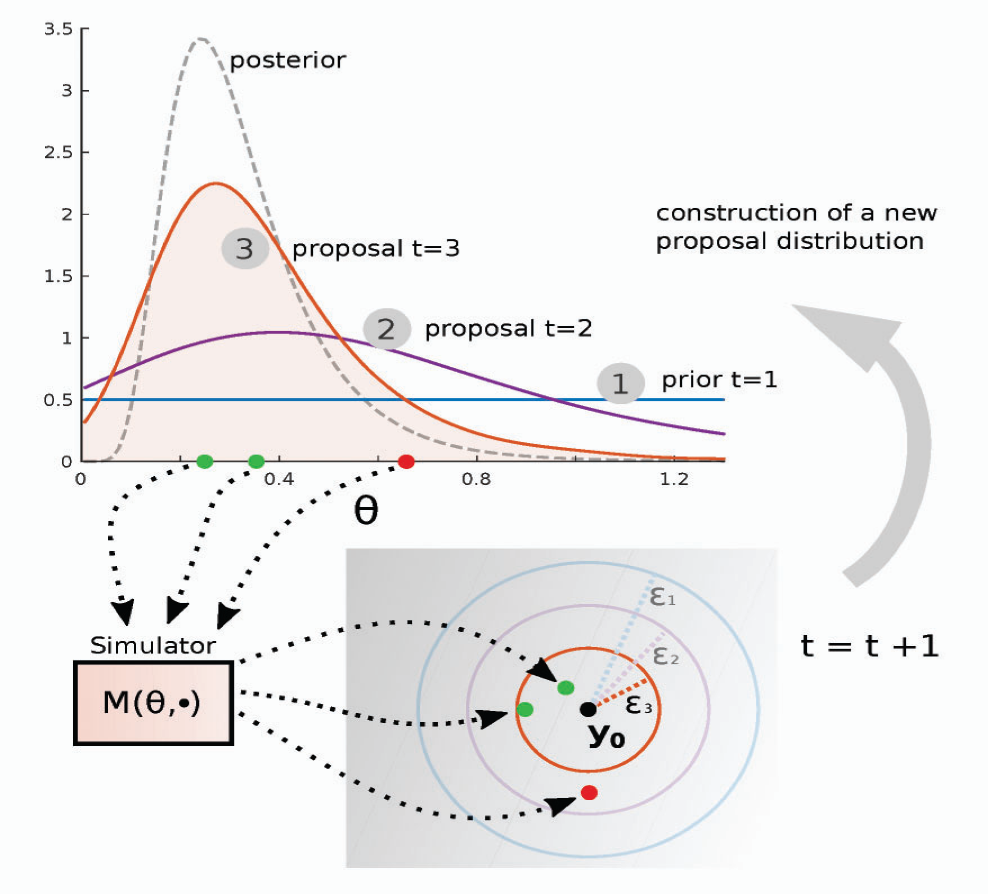


Figure 2 : Sequential ABC(Lintusaari)

**2.1.3. Bayesian Maximum Likelihood Estimation (BMLE)**

Bayesian Maximum Likelihood Estimation (BMLE) termed as “sampling from the posterior distribution” in (Menzies), approximates the posterior by applying sampling importance resampling.

The Steps below describe the algorithm of BMLE and how the method is implemented.

1. Draw a large number of parameter combinations from the prior distribution
2. For each parameter combination, run the model and estimate model outcomes
3. Using these model outcomes, estimate the likelihood for the parameter combination and retain this value (log-likelihood)
4. Resample from the original parameter sample with replacement, using the likelihood values as sampling weights.

Parameter combinations with high values of the likelihood are more consistent with the target supplied. This property allows the assessment of how the data supports one parameter combination compared with another.

**2.2. The Simulation Model**

**2.2.1. The SIR model**

The simulation model used in this study was a simple stochastic SIR (Susceptible - Infectious - Recovered) model, which we used to generate the observed data and also applied to the methods. The SIR model is an epidemiological and compartmental model that computes the number of infectious individuals with an infectious disease in a closed population over time. A closed population implies that the population size remains constant over time, that is, there are no births and deaths. The population is divide into three compartments (i.e. health states) – susceptible, Infectious and recovered. The rates in between the compartments determine how many individuals move from one compartment to another.

**Susceptible**

**Recovered**

**Infectious**

**β**  **γ**

Figure 3: Structure of the SIR compartmental model

This model involves a system of three non-linear ordinary differential equations (ODEs) that relates the number of susceptible, number of infectious, and number of recovered individuals (Weiss). The SIR model functions under several assumptions such as: the population under study is homogeneous (all individuals behave the same), the mode of transmission of the disease from infectious to susceptible individuals is through direct contact between infectious and susceptible individuals, recovered individuals gain permanent immunity to the disease. The following system of ODEs governs the dynamics of the SIR model

Where is the disease transmission rate, is the recovery rate, is the duration of infection and the basic reproductive number. is the proportion of individuals in the population that are susceptible to the disease and represents the proportion of infectious individuals. Susceptible individuals become infectious at a rate. At a rate , infectious individuals recover from the disease (gain permanent immunity to the disease) (LStone).

The population stays constant throughout the transmission dynamics over the set time such that

Figure 3 illustrates the dynamics of a stochastic SIR model run in the R software over time of 75 days for a population of 1000 individuals. The blue curve indicates the Susceptible compartment, the red curve indicates the Infectious compartment and the green curve indicates the Recovered individuals. The susceptible compartment reduces to zero as the infectious compartment gradually picks up.

**2.3. Performance Measures**

To compare the performance of the methods, we gave them equal amounts of time to run simulations and compared the resulting posterior to the reference posterior by computing percentage overlaps.

* + 1. **Running Simulations at equal amount of time**

To give each of the calibration methods an equal amount of computation time, we took the following approach:

1. Run 10000 simulations for each method and recorded the time each method took to run. We found out that Sequential ABC took longer to run the 10000 simulations compared to Rejection ABC.
2. Estimate the number of simulations Rejection ABC could run in Sequential time as
3. At tolerance of 1, run these estimated number of simulations using Rejection ABC and record the computation time.
4. Estimate the number of simulations run by Rejection ABC at the amount of time used by Sequential ABC as

This step fixes the time for the two methods

1. Calculate the tolerance for Rejection ABC such that the number of accepted parameter combinations is equal to that of sequential ABC as follows;
   * 1. **Percentage overlap**

To compare the posterior densities of the methods to the reference posterior density, we created a raster using the raster function from the raster library in R, which we used to compute percentage overlaps. A raster consists of a matrix of cells or pixels arranged into rows and columns to form a grid. Each cell contains a value which represents stored information. The raster was created by considering the minimum and maximum values of beta and gamma retained by the calibration methods to be compared as well as the reference. This was done so that the same raster could be applied to all the methods and reference. The resulting parameter space was divided into equally sized bins with beta values on the x-axis and gamma values on the y-axis (see Figure *4*). This formed a grid in which the posterior densities laid. We applied the grid to each posterior density to quantify the density of each cell or pixel.



Figure 4: 4x4 raster applied to a posterior density

The percentage overlap for each method was computed by summing the within cell density differences between the calibration method and the reference for that particular scenario (see equation… below).

Where is the percentage overlap for method and scenario , represents the matrix

form of the raster applied on calibration method and scenario , represents the matrix

form of the raster applied on the reference for scenario , *n* is the number of parameter combinations retained by each calibration method.