

---

# Bayesian Inference for Doubly Intractable Ordinary Differential Equations

---

**Lena Podina\***

Department of Computer Science  
The University of British Columbia  
Vancouver, BC, V6T 1Z2  
lena.podina@gmail.com

**Rachel Han\***

Department of Mathematics  
The University of British Columbia  
Vancouver, BC, V6T 1Z2  
hanrach@alumni.ubc.ca

**Tae Yoon (Harry) Lee \***

Faculty of Pharmaceutical Sciences  
The University of British Columbia  
Vancouver, BC, V6T 1Z3  
dlxodbs@student.ubc.ca

## Abstract

Uncertainty quantification is a growing field in applied mathematics to deal with noisy data in differential equation models. Approximate Bayesian computation is a class of Bayesian methods originating from statistics that can be easily applied to such deterministic models to cope with uncertainty as well as intractable likelihoods. In this project, we implement three variants of approximate Bayesian methods and apply them to estimate the parameters of the Susceptible-Infected-Recovered (SIR) epidemiological model and of the Lorenz system. We rigorously validate the methods via a simulation-based approach called Bayesian calibration, and analyze the COVID data using the SIR model.

## 1 Introduction

Ordinary differential equations (ODEs), or equations of the form

$$y^{(n)} = F(\theta, x, y^{(1)}, \dots, y^{(n-1)})$$

where  $y = f(x)$ , are used to model various natural processes, such as disease spread and lumber strength. Here the function  $f$  is the solution to the ODE. Usually there are several parameters  $\theta$  in  $F$  which must be provided (or estimated) before the ODE can be used for predictions. In other cases, such as in the modelling of epidemics, the parameters themselves have a physical meaning, such as the level of infectiousness of the illness. Hence, finding the parameters that govern the model is an important inference task.

A traditional way of estimating ODE parameters involves deterministic methods (e.g., using a numerical solver). In the era of big data, uncertainty quantification is ever more important. Bayesian methods offer a natural extension to the deterministic methods to incorporate uncertainty arising from measurement errors (Chkrebtii et al., 2016). Using the Bayes rule, the posterior of the parameters given the data can be written as:

$$\pi(\theta|D) = \frac{\pi(D|\theta)\pi(\theta)}{\pi(D)}$$

---

\*Equally contributed

where  $\pi(D)$  is the normalization constant,  $\pi(\theta)$  is the prior over the parameters, and  $\pi(D|\theta)$  is the likelihood. In cases where only the normalizing constant is intractable, classical Bayesian methods, notably Markov Chain Monte Carlo (MCMC), can be used for inference. However, when the likelihood is also intractable or difficult to compute, these “doubly intractable” problems (Murray et al., 2006) require other techniques. For instance, some nonlinear ODEs may only admit an implicit solution, such as in Yang et al. (2018). In this case, computing a likelihood is intractable. Well-known and studied doubly-intractable problems include the Ising model and Gaussian Markov random field models (Lyne et al., 2015).

## 1.1 Approximate Bayesian Computation

Approximate Bayesian Computation (ABC) is a class of *approximate* Bayesian inference methods which avoids computing the likelihood and normalizing constant altogether. The ABC philosophy was established by Rubin (1984), asserting that simulation-based inference provides the freedom to explore models without being constrained to those that are analytically convenient. Suppose that we are given data  $y_0$  from a process governed by an ODE and wish to infer its parameters. The key idea in all of the ABC methods is to sample parameters  $\theta^*$  from a prior distribution  $\pi(\theta)$  and compute the distance  $d(y_0, y^*)$  between the observed data  $y_0$  and simulated data  $y^*$  using the sampled parameters. If the distance is within some tolerance level  $\epsilon > 0$ , then we retain that sample and repeat the procedure until we have a desired number of posterior samples of the parameters to approximate the posterior distribution. The method is approximate for continuous  $y_0$  (but is exact for discrete cases), because the probability of drawing the exactly the same observed sample is zero. Thus, the approximation  $\pi(\theta|d(y_0, y^*) \leq \epsilon)$  to the posterior  $\pi(\theta|y_0)$  depends on  $\epsilon$ . If the value  $\epsilon$  is small, then the ABC methods will approximate the posterior well but at potentially high computational cost.

We will investigate ABC-rejection (ABC-Rej; Tavaré et al. (1997)), ABC Markov Chain Monte Carlo (ABC-MCMC; Marjoram et al. (2003)), and ABC Sequential Monte Carlo (ABC-SMC; Toni et al. (2009)). ABC-Rej is a simple algorithm which rejects samples  $\theta^*$  where  $d(y_0, y^*) > \epsilon$ , but it suffers from the curse of dimensionality. ABC-MCMC generates samples similarly to Metropolis-Hastings, but can have poor mixing and high correlation of samples. ABC-SMC fares better than the previous algorithms. The algorithm generates  $N$  samples (particles) of  $\theta$ , which it propagates through a series of distributions. See Appendices 1, 2, and 3 for further details and pseudocode.

## 1.2 Contribution

As part of this project we have implemented ABC-Rej, ABC-MCMC and ABC-SMC in Julia. We rigorously validated them using Bayesian calibration, which were not done in related works. Additionally, we compared their performance under two different simulation studies using measures of effective sample size and CPU-time. These measures were not used in related works, and we found them to provide valuable insights. Furthermore, we clearly showed the limitation of the Susceptible-Infected-Recovered model for analyzing COVID-19 data. All the codes for this report can be found in our [GitHub repository](#).

## 2 Related work (applications of ABC to ODEs)

Toni et al. (2009) is one of the first to apply ABC to epidemiological models. They use an algorithm based on ABC-SMC, and compare its parameter estimation performance to that of other algorithms. Applying this algorithm in several different case studies, such as predator-prey dynamics, gene expression and disease spread, they also generate several models for each case study, and use Bayesian model selection to rank them. For example, in the disease spread application, Bayesian model selection was used to rank four different SIR-based epidemiological models. The parameters of these models was estimated during Bayesian model selection, using the ABC-SMC algorithm. This paper shows the flexibility of ABC in being applied, without change, in many different scenarios. Results show that ABC-MCMC greatly reduces the computational cost of rejection-ABC, but ABC-SMC needs about the same number of data generation steps as MCMC. We have based our implementation of ABC-SMC on this work and aim to extend it by applying it to novel scenarios not covered in the paper, such as the Lorenz system, and using Bayesian calibration to compare the three algorithms.

Picchini (2014) applies ABC-MCMC to models defined by stochastic differential equations (SDEs). Even though these models capture uncertainty better than their deterministic counterparts, likelihoods may have no closed form. Separately from this, traditional MCMC methods may suffer from issues such as poor mixing. A flexible and general method for inferring parameters is therefore useful. Picchini (2014) avoids simulating when the proposal is to be rejected with 100% certainty, which saves computation time. The application they considered is drug pharmacokinetics, which models the dynamics of drug absorption into the blood stream. When building the model, they assumed and accounted for Gaussian noise in the measurement data, and then compared the performance of exact inference (particle MCMC) and ABC-MCMC. The results showed that ABC-MCMC and particle MCMC obtained similar results, although exact Bayesian inference had a slightly higher effective sample size. The ABC algorithm was tested in several other scenarios, and overall results showed that exact and approximate Bayesian inference performed comparably, with a 40-50% reduction in computational effort for ABC methods.

Minter and Retkute (2019) is a tutorial for how to apply ABC (in several different forms) to various ODEs that govern epidemic spread. Two different epidemic models are investigated using ABC: parameters of the SIR model were found using rejection-ABC. Parameters for the stochastic Susceptible-Exposed-Infected-Recovered (SEIR) model were found using ABC-SMC. In the first case study, results show narrower posteriors around the true value when the daily counts of infected and recovered individuals were known, and more uniform posteriors when the number of infected and the size of the epidemic was only known at the end. However, the two ABC methods were run on different case studies, making comparison between them difficult. Also, no Bayesian calibration was done to compare the methods. We aim to extend their work by running all three algorithms on each case study, and measuring both the Bayesian calibration, and effective sample size of each algorithm.

### 3 ODE models

#### 3.1 Susceptible-Infected-Recovered (SIR) model

The SIR model (Kermack and McKendrick, 1927) is characterized by three quantities: the number of susceptible people  $S(t)$ , the number of active cases  $I(t)$  and the total number of recovered  $R(t)$  across time  $t$ . The model assumes a closed population without new births or deaths. Hence the number of population  $N = S(t) + I(t) + R(t)$  is fixed for all  $t$ . It is described by the following ODE:

$$\frac{dS(t)}{dt} = -\beta \frac{S(t)I(t)}{N}, \quad (1)$$

$$\frac{dI(t)}{dt} = \beta \frac{S(t)I(t)}{N} - \gamma I(t), \quad (2)$$

$$\frac{dR(t)}{dt} = \gamma I(t), \quad (3)$$

where  $\gamma$  represents how many infected people will recover per unit time (recovery rate) and  $\beta$  is the transmission rate at which susceptible people get infected. Another useful quantity that can be inferred from the parameters is the basic reproduction number  $R_0$ , which is the number of people an infected person can infect. For SIR models,  $R_0$  is given by the ratio  $\beta/\gamma$ , as derived in Jones (2007).

#### 3.2 Lorenz system

We consider applying ABC on a deterministic chaotic system where the likelihood of the solution given parameters is difficult to compute owing to its chaotic behaviour. One such example is the Lorenz system (Lorenz, 1963), characterized by three parameters,  $\sigma$ ,  $\rho$ , and  $b$ :

$$\frac{dx}{dt} = \sigma(y - x), \quad (4)$$

$$\frac{dy}{dt} = \rho x - y - xz, \quad (5)$$

$$\frac{dz}{dt} = xy - bz. \quad (6)$$

The system exhibits chaotic behaviour in which the trajectory of the solution is non-periodic. However, the trajectory will still lie on some invariant set, called the ‘attractor’ (which looks like a butterfly).

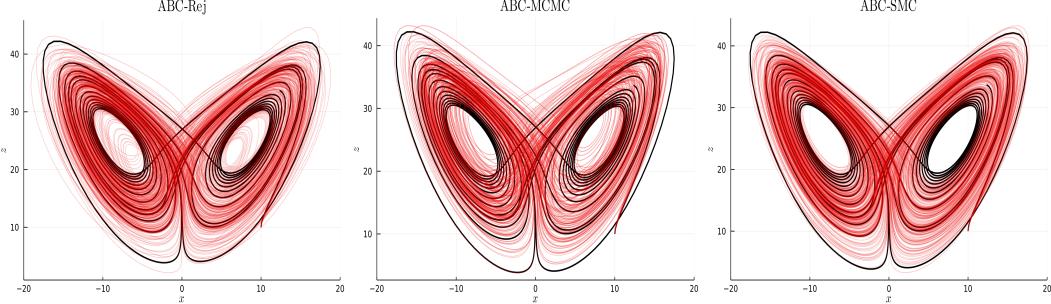


Figure 1: The trajectories of Lorenz system based on the estimates  $\sigma$ ,  $\rho$ , and  $b$  on  $xz$  plane. The red lines represent solution from 20 random samples out of 100. The black lines denote  $y_0$ .

## 4 Empirical studies

We conduct a simulation study on each of the ODE models to validate our implementation of the three ABC algorithms. For validation, we perform Bayesian calibration (see Appendix E for details; Gelman et al. (2020)). In addition, we compare performance of the algorithms based on effective sample size (ESS; Flegal et al. (2010)), CPU-time and ESS per CPU-time. Since the Lorenz system is a harder ODE problem than the SIR model, we set aside simulation study results of the SIR model in the appendix (see Section D).

Next, we apply ABC methods to analyze the COVID-19 data with the SIR model. We extract COVID-19 data on British Columbia, Canada from the [Government of Canada](#). We compute the posterior distribution of the SIR model parameters for each of the ABC methods and show fitted trajectories through a functional version of a (univariate) box plot (Sun and Genton, 2011).

To generate the simulated data  $y^*$  given the sampled parameter  $\theta^*$ , we use the solution from a deterministic ODE solver. We assume that the data generating process has no error and the initial conditions are known. We use truncated normal distributions with range  $(0, \infty)$  for all ODE parameters under consideration, as they are all positively-valued.

### 4.1 Simulation study: Lorenz system

For each experiment, we sample the true parameters  $\theta_0 = (\sigma_0, \rho_0, b_0)$  as follows:

$$\begin{aligned}\sigma_0 &\sim \text{Truncated}(\mathcal{N}(10, 0.01)), \\ \rho_0 &\sim \text{Truncated}(\mathcal{N}(26, 0.01)), \\ b_0 &\sim \text{Truncated}(\mathcal{N}(8/3, 0.01)).\end{aligned}$$

Using  $\theta_0$ , we find the solution  $y_0$  to Lorenz system up to  $T = 20$  time periods using an ODE solver. We use the same distribution but larger standard deviations for the prior of each parameter for the ABC methods:

$$\begin{aligned}\sigma &\sim \text{Truncated}(\mathcal{N}(10, 5)), \\ \rho &\sim \text{Truncated}(\mathcal{N}(26, 10)), \\ b &\sim \text{Truncated}(\mathcal{N}(8/3, 1)).\end{aligned}$$

The distance function between the true solution  $y_0$  and approximate solution  $y^*$  is the discrete form of

$$d(y_0, y^*) = \left( \int_0^T |y_0(t) - y^*(t)|^2 e^{-at} dt \right)^{1/2},$$

where  $a$  is set to 0.001. The time-dependent exponential decay term  $e^{-at}$  is introduced to place less weight on the difference at later times and thus accepting more samples in the ABC algorithms, as the distance is likely to become exponentially large as the chaotic system evolves for longer period of time.

For the simulations, we choose the following hyper-parameters to obtain a good fit. The distance tolerance for ABC-MCMC is set higher than others with  $\epsilon = 50$ . Additionally, we take samples

every 20-th step (thinning) and discard the first 100 (burn-in). For ABC-SMC, we choose  $\epsilon = [70, 65, 60, 55, 50, 44, 40]$ . Lastly, we set  $\epsilon = 40$  for ABC-Rej. We sample 250 parameters for each method.

Figure 1 shows some of the fitted solutions for each method. Visually ABC-SMC appears to have the best fitted trajectories. Figure 2 illustrates that ABC-Rej and ABC-SMC achieve a desired calibration level of 0.9, whilst ABC-MCMC does not. A very low value of ESS in Table 1 suggests that ABC-MCMC does not appear to accept new proposals. On the other hand, as expected, the ESS of ABC-Rej is equal to the sample size of 250 on average. ABC-SMC has the ESS of 150. In terms of ESS per CPU-time, ABC-Rej was the best with an average value of 627, followed by ABC-SMC with an average value of 104. ABC-MCMC is the worst performing one, with the lowest value of ESS and highest value of CPU-time.

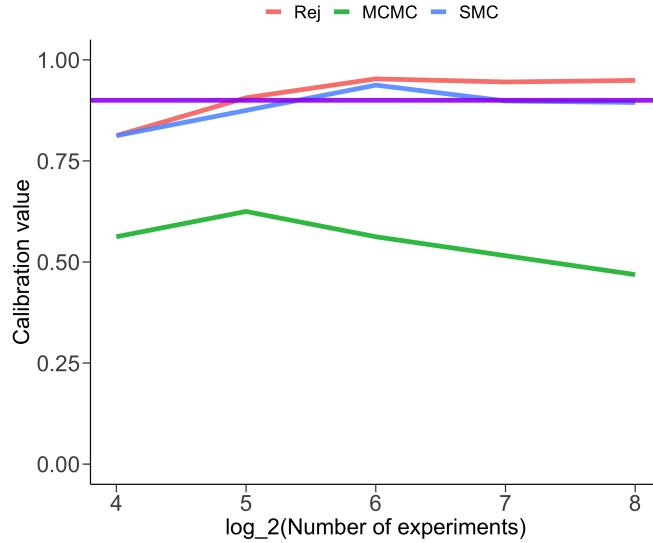


Figure 2: Bayesian calibration of ABC-Rej (red), ABC-MCMC (green), and ABC-SMC (blue) for a desired level of 0.9 (purple) with the Lorenz system.

Table 1: Simulation results for Lorenz model (best value bold-faced)

	ESS			CPU time (s)			ESS/time (1/s)		
	Rej	MCMC	SMC	Rej	MCMC	SMC	Rej	MCMC	SMC
Median	<b>250</b>	12	126	<b>0.42</b>	6.56	1.49	<b>622</b>	2	84
Mean	<b>268</b>	13	156	<b>0.45</b>	6.65	1.52	<b>627</b>	2	104

## 4.2 Analysis of the COVID-19 data

Estimating parameters of epidemic models such as SIR could provide useful insight to a pandemic. For example, if a given epidemic model closely describes the actual data, one could predict how the number of infected people would evolve over time given the parameters. The ABC methods can potentially provide a flexible framework of generating samples from the posterior distribution of parameters if the prior is reasonable.

We note that the SIR model assumes all individuals are equally likely to contract the disease. Furthermore, it assumes that if a person recovers from the disease, they are permanently immune. We assume a closed population of 4.5 million people (the total population in BC as of 2020) and take  $S(0) = 4,500,000$ . We also treat the number of deaths as the number of recovered since they cannot infect the susceptible. The COVID-19 Canada dataset includes the number of active cases, recovered and deaths in each province per day. We extract data for BC only and split the data into susceptible,

infected and recovered in the following way:

$$R(t) = \text{number of people recovered} + \text{number of deceased}, \quad (7)$$

$$I(t) = \text{number of active cases}, \quad (8)$$

$$S(t) = 4500000 - (I(t) + R(t)) \quad (9)$$

where  $t$  is in the unit of one week. The dates of our interest span from 2020-08-10 to 2021-04-10. Because the vaccine was introduced early January of 2021, we further split the data into pre-vaccine and post-vaccine periods to estimate the posterior distribution of parameters for each period separately.

#### 4.2.1 Hyper-parameter set-up

For priors for the parameters  $\beta$  and  $\gamma$ , we choose the truncated normal distribution with range  $(0, \infty)$  since the parameters are positive,

$$\begin{aligned} \beta &\sim \text{Truncated}(\mathcal{N}(0.1, 0.5)), \\ \gamma &\sim \text{Truncated}(\mathcal{N}(0.2, 0.2)). \end{aligned}$$

We use the Euclidean norm distance function:

$$d = \frac{1}{T} \sum_{t=1}^T \sqrt{(I_{sim}^t - I_{real}^t)^2 + (R_{sim}^t - R_{real}^t)^2}.$$

For ABC-rejection, we set a small tolerance level  $\epsilon = 0.1$ . We give a range of values  $\epsilon = [10, 5, 1, 0.5, 0.1]$  for ABC-SMC allowing the algorithm to gradually estimate the posterior. For ABC-MCMC,  $\epsilon$  that is too small will prevent algorithm from accepting the proposed samples. Therefore, we choose  $\epsilon = 1$ .

Another hyper-parameter to consider for ABC-MCMC and ABC-SMC is the standard deviation ( $\sigma^2$ ) in the Gaussian random walk kernel used in the algorithms. As in the Metropolis-Hastings algorithm,  $\sigma^2$  controls the acceptance of the samples. If it is too high, many proposed samples will be rejected which results in no variation in the samples. This is especially apparent in ABC-MCMC algorithm. On the other hand if  $\sigma^2$  is too low, we have highly correlated samples. Considering these factors and after trial and error, we choose  $\sigma^2 = 0.5$  for ABC-SMC. To account for the relatively large  $\epsilon$  for ABC-MCMC, we choose  $\sigma^2 = 0.25$ . Additionally, we use thinning of 10000 and burn in of 100 to produce less correlated samples for ABC-MCMC.

#### 4.2.2 Results

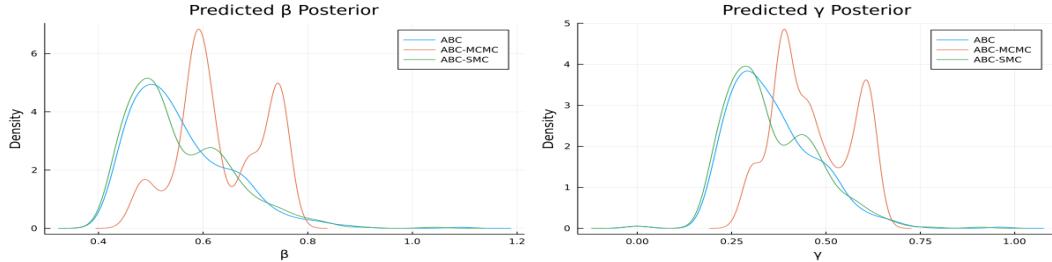


Figure 3: Density plots of estimated parameters  $\beta$  and  $\gamma$  in the post-vaccine period.

For each method, we collected 500 samples of the parameters. Figure 3 shows the density plots of the estimated distribution before the vaccine. ABC-rejection and ABC-SMC give similar distributions with the mode at  $\beta \approx 0.5, \gamma \approx 0.3$ . ABC-MCMC is localized at two sets of values with narrow dispersion:  $\beta \approx \{0.59, 0.75\}$  and  $\gamma \approx \{0.4, 0.6\}$ . The narrow dispersion reflects how ABC-MCMC may not be as effective in exploring the sample space.

We observe a similar trend in Figure 4 but notice a larger dispersion in the posterior density for  $\beta$  for ABC-MCMC. The transmission rate predicted by the mode of the estimated posterior is  $\beta \approx 0.45$  for ABC-Rej and ABC-SMC. ABC-MCMC is spread over similar values,  $\beta \approx (0.4, 0.55)$ . We note that

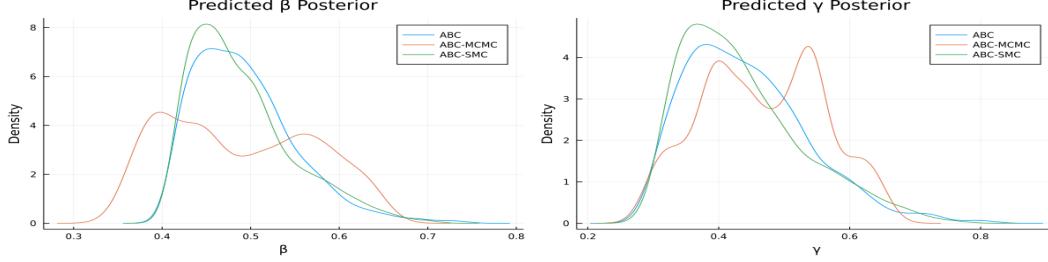


Figure 4: Density plots of estimated parameters  $\beta$  and  $\gamma$  in the post-vaccine period.

the predicted transmission rate in the post-vaccine period is slightly lower than the pre-vaccine. The point estimate for the recovery rate is  $\gamma \approx 0.28$  and  $\gamma \approx 0.6$  respectively for ABC-Rej, ABC-SMC and ABC-MCMC. We also note here that the predicted recovery rate during post-vaccine is similar to pre-vaccine period.

We give a physical interpretation of the parameters. For example, in the SIR setting, the  $\beta = 0.5$  can be interpreted as the number of people an individual can infect per week. The recovery rate  $\gamma = 0.4$  means the infectious period is  $1/0.4 = 2.5$  weeks.

For each of the pre and post vaccine periods, using the estimated posterior distribution of the parameters, we produce 500 predicted trajectories of  $S(t)$ ,  $I(t)$ , and  $R(t)$ . We visualize  $I(t)$  and  $R(t)$  through functional box plots. Figures 5 and 6 show the actual observed data in black ( $I(t)$ ) and purple ( $R(t)$ ) along with its median predicted trajectory in red and green, respectively, along with their 50% regions. We observe that predicted trajectories from ABC-MCMC better encompass the data because the estimated posterior is not as skewed as those computed by ABC and ABC-SMC. We think that this behaviour is likely caused by higher  $\epsilon$  set for ABC-MCMC ( $\epsilon = 1$ ).

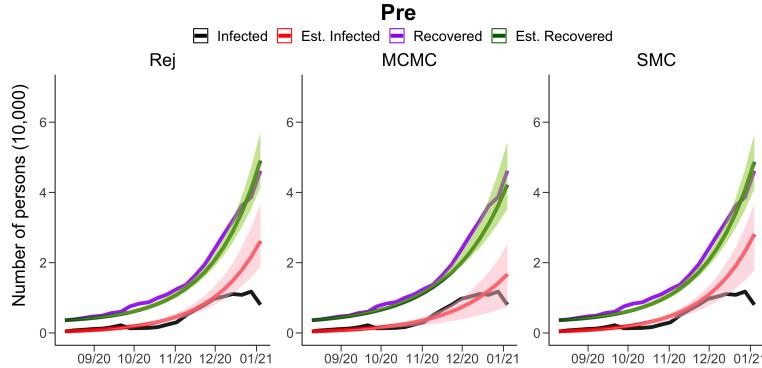


Figure 5: Functional box plot of the predicted curves for pre-vaccine period

## 5 Discussion

We have successfully implemented and validated three ABC methods, ABC-Rej, ABC-MCMC and ABC-SMC, using Bayesian calibration on two deterministic ODEs. For the Lorenz system where the sensitivity to parameters is significant, the ABC methods gave reasonable estimates of the true chaotic trajectory. We conclude the following distinctions between the methods from the simulation study: ABC-Rej generates samples with the most variation. This is due to the fact that accepted samples are sampled from the prior independently every iteration. In contrast to ABC-Rej, the accepted samples in ABC-MCMC are prone to move only a little from the initial sample because the acceptance rate is either too high or too low, resulting in low ESS. We found that the proposed samples were accepted only a handful of times despite the bigger tolerance, thinning and burn-in. We postulate that ABC-MCMC was nailing down the *perturbed* posterior distribution in the simulation study, since we can view ABC as an exact inference method for a perturbed problem as a function of  $\epsilon$  (Fearnhead, 2018). We expect we need smaller tolerance and more thinning and burning to generate

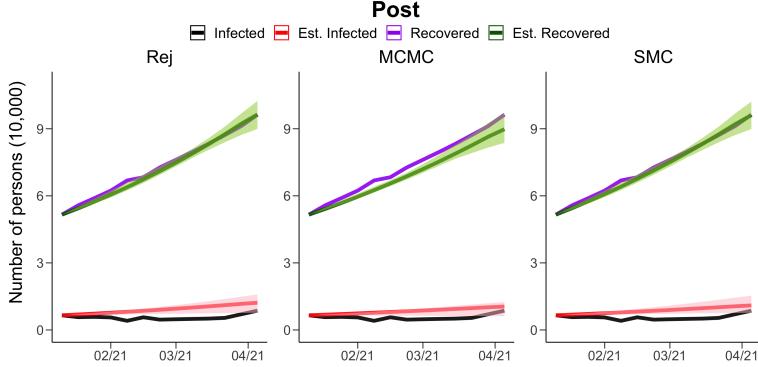


Figure 6: Functional box plot of the predicted curves for post-vaccine period

more independence or variations in samples although that would significantly increase computation time. ABC-SMC also uses the Gaussian kernel to propose samples but the weighting and resampling schemes avoid the problem of samples being stuck in one region. Additionally, because the samples are propagated gradually through decreasing tolerance  $\epsilon_1 > \epsilon_2 > \dots > \epsilon_T$ , it gives tighter bounds for the estimation. Overall, we recommend ABC-SMC for its efficiency and flexibility with caution.

One crucial weakness of our simulation study was lack of exploration of important ABC factors, namely summary statistic and distance function. These important factors are well-known to be very crucial in both estimation and efficiency for ABC methods (Aeschbacher et al., 2012; Burr and Skurikhin, 2013; Prangle et al., 2017). We relied on the summary statistic (which was just the input itself) and distance functions that worked well in relevant studies. Moreover, we did not test the scenario where there was noise in data generating process.

From the analysis of the COVID-19 data, we observed that ABC-MCMC was able to explore the parameter space better with high  $\epsilon$  value, thinning and burn-in at the cost of more computation time. With the same  $\epsilon$  for ABC-Rej and ABC-SMC, the predicted posteriors were very similar. We also note that the waves in the infection curve was hard to capture with the SIR model as we saw in Figures 5 and 6. Other models such as Susceptible-Exposed-Infectious-Recovered, Susceptible-Exposed-Infectious-Susceptible, and their stochastic variations may better fit to the actual data. For these models, we can easily apply the framework of ABC to do inference.

In summary, ABC provides a flexible framework to incorporate uncertainty in differential equations models. There is overflowing literature of applications of ABC to various differential equations models, such as stochastic and partial differential equations. We advocate the use of Bayesian calibration and various measures, such as ESS (if correlation is introduced in the method of interest) and CPU-time, for validation of new methods for methodology development or of existing methods for new applications.

## References

- Aeschbacher, S., M. A. Beaumont, and A. Futschik (2012). A novel approach for choosing summary statistics in approximate bayesian computation. *Genetics* 192(3), 1027–1047.
- Burr, T. and A. Skurikhin (2013). Selecting summary statistics in approximate bayesian computation for calibrating stochastic models. *BioMed research international* 2013.
- Chkrebtii, O. A., D. A. Campbell, B. Calderhead, M. A. Girolami, et al. (2016). Bayesian solution uncertainty quantification for differential equations. *Bayesian Analysis* 11(4), 1239–1267.
- Del Moral, P., A. Doucet, and A. Jasra (2012). An adaptive sequential monte carlo method for approximate bayesian computation. *Statistics and computing* 22(5), 1009–1020.
- Fearnhead, P. (2018). Asymptotics of abc. *Handbook of Approximate Bayesian Computation*, 269–288.

- Flegal, J. M., G. L. Jones, et al. (2010). Batch means and spectral variance estimators in markov chain monte carlo. *The Annals of Statistics* 38(2), 1034–1070.
- Fu, Y.-X. and W.-H. Li (1997). Estimating the age of the common ancestor of a sample of dna sequences. *Molecular biology and evolution* 14(2), 195–199.
- Gelman, A., A. Vehtari, D. Simpson, C. C. Margossian, B. Carpenter, Y. Yao, L. Kennedy, J. Gabry, P.-C. Bürkner, and M. Modrák (2020). Bayesian workflow. *arXiv preprint arXiv:2011.01808*.
- Gordon, N. J., D. J. Salmond, and A. F. Smith (1993). Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE proceedings F (radar and signal processing)*, Volume 140, pp. 107–113. IET.
- Jones, J. H. (2007). Notes on r0. *California: Department of Anthropological Sciences* 323.
- Kermack, W. O. and A. G. McKendrick (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character* 115(772), 700–721.
- Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of atmospheric sciences* 20(2), 130–141.
- Lyne, A.-M., M. Girolami, Y. Atchadé, H. Strathmann, D. Simpson, et al. (2015). On russian roulette estimates for bayesian inference with doubly-intractable likelihoods. *Statistical science* 30(4), 443–467.
- Marjoram, P., J. Molitor, V. Plagnol, and S. Tavaré (2003). Markov chain monte carlo without likelihoods. *Proceedings of the National Academy of Sciences* 100(26), 15324–15328.
- Minter, A. and R. Retkute (2019). Approximate bayesian computation for infectious disease modelling. *Epidemics* 29, 100368.
- Murray, I., Z. Ghahramani, and D. MacKay (2006). MCMC for doubly-intractable distributions. *arXiv preprint arXiv:1206.6848*.
- Picchini, U. (2014). Inference for sde models via approximate bayesian computation. *Journal of Computational and Graphical Statistics* 23(4), 1080–1100.
- Prangle, D. et al. (2017). Adapting the abc distance function. *Bayesian Analysis* 12(1), 289–309.
- Rubin, D. B. (1984). Bayesianly justifiable and relevant frequency calculations for the applies statistician. *The Annals of Statistics*, 1151–1172.
- Sisson, S. A., Y. Fan, and M. M. Tanaka (2007). Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences* 104(6), 1760–1765.
- Sun, Y. and M. G. Genton (2011). Functional boxplots. *Journal of Computational and Graphical Statistics* 20(2), 316–334.
- Tavaré, S., D. J. Balding, R. C. Griffiths, and P. Donnelly (1997). Inferring coalescence times from dna sequence data. *Genetics* 145(2), 505–518.
- Toni, T., D. Welch, N. Strelkowa, A. Ipsen, and M. P. Stumpf (2009). Approximate bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface* 6(31), 187–202.
- Yang, C.-H., J. V. Zidek, and S. W. Wong (2018). Bayesian analysis of accumulated damage models in lumber reliability. *Technometrics*.

## Appendix

### A ABC-rejection algorithm

ABC-rejection (ABC-Rej) is the simplest ABC algorithm, proposed first by Tavaré et al. (1997) and later improved on by Fu and Li (1997). ABC-rej involves repeatedly sampling  $\theta$  from its prior, simulating  $y^*$  given  $\theta$ , and accepting the sampled  $\theta$  if it is similar enough to the original dataset. If  $\theta$  has a high dimension, it may be difficult to efficiently explore the space of relevant  $\theta$ , as a result of the “curse of dimensionality”. Hence the acceptance probability is likely to be low.

---

**Algorithm 1:** ABC for a continuous space

---

```
input : prior distribution  $\pi(\cdot)$ 
input : likelihood  $f(\cdot|\theta)$ 
input :  $y$  in  $\mathbb{R}^d$ 
input : summary statistic  $\eta : \mathbb{R}^d \rightarrow Z$ 
input : measure  $d : (Z, Z) \rightarrow \mathbb{R}$ 
input : tolerance  $\epsilon > 0$ 
    initialize samples = {}
    for  $i = 1 \dots N$  do
        repeat
            sample  $\theta'$  from the prior distribution  $\pi(\cdot)$ 
            sample  $y'$  from the likelihood  $f(\cdot|\theta')$ 
        until  $d(\eta(y), \eta(y')) \leq \epsilon$ ;
        samples = samples  $\cup \theta'$ 
    end
output : samples
```

---

## B ABC-MCMC algorithm

ABC Markov Chain Monte Carlo (ABC-MCMC), introduced by Marjoram et al. (2003), explores the parameter space more efficiently than ABC-rej. It also involves generating parameter samples  $\theta^{(1)} \dots \theta^{(t)}$  to create a posterior on  $\theta$ . The key steps of the algorithm are as follows: 1) propose  $\theta^{(i)} \sim q(\theta^{(i)} | \theta^{(i-1)})$  for a proposal distribution or kernel  $q$ , 2) simulate  $y^{(i)}$  from  $\theta^{(i)}$  until  $y^{(i)}$  is sufficiently similar to the data, and 3) accept  $\theta^{(i-1)}$  in a Metropolis-Hastings fashion proportionally to  $\Pr(\theta^{(i)}) / \Pr(\theta^{(i-1)})$ , correcting for asymmetry of  $q$ . A disadvantage of ABC-MCMC is potential for very high correlation of samples, and tendency for the chain to get stuck in regions of low probability for long periods of time, as described in Toni et al. (2009).

---

**Algorithm 2:** ABC-MCMC

---

```

input : prior distribution  $\pi(\cdot)$ 
input : likelihood  $f(\cdot | \theta)$ 
input :  $y$  in  $\mathbb{R}^d$ 
input : summary statistic  $\eta : \mathbb{R}^d \rightarrow Z$ 
input : measure  $d : (Z, Z) \rightarrow \mathbb{R}$ 
input : tolerance  $\epsilon > 0$ 
input : proposal function  $q$ 
    initialize samples =  $\{(\theta^0, y^0)\}$  using Algorithm 1
    for  $i = 1 \dots N - 1$  do
        sample  $\theta'$  from  $q(\cdot | \theta^{i-1})$ 
        sample  $y'$  from the likelihood  $f(\cdot | \theta')$ 
        sample  $u$  from Uniform[0,1]
        let  $\alpha = \frac{\pi(\theta')q(\theta^{(i-1)} | \theta')}{\pi(\theta^{(i-1)})q(\theta' | \theta^{(i-1)})}$ 
        if  $u \leq \alpha$  and  $d(\eta(y'), \eta(y)) \leq \epsilon$  then
            samples = samples  $\cup (\theta', y')$ 
        else
            samples = samples  $\cup (\theta^{(i-1)}, y^{(i-1)})$ 
        end
    end
output : samples

```

---

## C ABC-SMC algorithm

Lastly we consider ABC Sequential Monte Carlo (ABC-SMC), which was first proposed by Gordon et al. (1993). This method extends a particle filter method to an ABC setting, and partially avoids the problems of the previous two algorithms. This was shown by Sisson et al. (2007). The SMC algorithm we use is based on Toni et al. (2009). As described in detail in that paper, this algorithm involves propagating a set of sampled parameter values (or particles)  $\{\theta^{(1)}, \dots, \theta^{(N)}\}$  through a set of intermediate distributions,  $\pi(\theta|d(y_0, y^*) \leq \epsilon_i)$ ,  $i \in [1 \dots T]$ . Each  $\epsilon_i$  is chosen such that  $\epsilon_1 > \dots > \epsilon_T$ , so that  $\pi(\theta|d(y_0, y^*) \leq \epsilon_i)$  gradually converges to the posterior.

In the algorithm below, we use the notation  $\theta_t, \epsilon_t$  and  $y_t$  with subscript  $t$  where  $t \in \{0, \dots, T\}$  to denote the  $t$ -th evolution time. The superscript  $i \in \{0, \dots, N\}$  indicates the  $i$ -th sample. The observed data  $y_0$  is denoted  $y_e$  to avoid confusion. The effective sample size for the SMC algorithm is taken as  $ESS = \frac{1}{\sum_{j=1}^N (W_t^j)^2}$  as suggested by Del Moral et al. (2012). The systematic resampling was used in the resampling step.

---

**Algorithm 3:** ABC-SMC (Toni et al., 2009)

---

**input** : prior distribution  $\pi(\cdot)$   
**input** : likelihood  $f(\cdot|\theta)$   
**input** : data  $y \in \mathcal{D}$   
**input** : summary statistic  $\eta : \mathbb{R}^d \rightarrow Z$   
**input** : measure  $d : (Z, Z) \rightarrow \mathbb{R}$   
**input** : MCMC kernel  $q$   
**input** : Number of samples  $N$   
**input** : ESS Threshold  
**input** :  $\epsilon_1 > \epsilon_2 > \dots > \epsilon_T$ , where  $T$  is number of evolution

Initialize  $W_0^{(i)} = \frac{1}{N}$ .  
Set  $i = 0$   
**while**  $i < N$  **do**  
    sample  $\theta_0^{(i)}$  from  $\pi(\cdot)$   
    sample  $y_0^{(i)}$  from  $f(\cdot|\theta_0^{(i)})$   
    **if**  $d(\eta(y_e) - \eta(y_0^{(i)}))$  **then**  
        samples = samples  $\cup \theta_0^{(i)}$   
    *i* = *i* + 1  
**end**  
Set  $t = 0$   
**while**  $t > T$  **do**  
    Set  $i = 0$   
    **while**  $i < N$  **do**  
        sample  $\theta^*$  from previous  $\{\theta_{t-1}^{(0,1,\dots,N)}\}$  with weights  $W_{t-1}^{(0,\dots,N)}$   
        Sample  $\theta^{**}$  from  $q(\cdot|\theta^*)$   
        **if**  $\pi(\theta^{**}) = 0$  **then**  
            start again with  $i = 0$   
        sample  $y_t^{(i)}$  from  $f(\cdot|\theta^{**})$   
        **if**  $d(\eta(y_e) - \eta(y_t^{(i)})) < \epsilon_t$  **then**  
             $\theta_t^{(i)} \leftarrow \theta^{**}$   
             $W_t^{(i)} = \frac{\pi(\theta^{(i)}_t)}{\sum_{j=1}^N W_{t-1}^{(j)} q(\theta_{t-1}^{(j)} | \theta_t^{(j)})}$   
        *i* = *i* + 1  
**end**  
Normalize weights  
**if**  $ESS < ESS$  Threshold **then**  
    resample  $\theta_t^{(0,\dots,N)}$  with weights  $W_t^{(0,\dots,N)}$   
     $\theta_{t-1}^{(0,\dots,N)} \leftarrow \theta_t^{(0,\dots,N)}$   $t = t + 1$   
**end**  
**output** :  $\theta_T^{(0:N)}$

---

## D Simulation study of the SIR model

We set up the experiments to emulate the COVID-19 data as closely as possible. We use the initial values of 1, 0.0001, and 0.07 for  $S_0$ ,  $I_0$ , and  $R_0$ , respectively. We assign the following *true* distributions to the model parameters:

$$\begin{aligned}\beta_0 &\sim \text{Truncated}(\mathcal{N}(2, 0.5)), \\ \gamma_0 &\sim \text{Truncated}(\mathcal{N}(1, 0.25)).\end{aligned}$$

We use those same distributions as the priors for the ABC methods. We set the number of experiments  $N$  to  $2^{10}$ , the number of samples  $n$  to 250, the total time period  $T$  to 20, and the tolerance level  $\epsilon$  to 0.1 for ABC-Rej, 1 for ABC-MCMC, and  $[10, 5, 1, 0.5, 0.1]$  for ABC-SMC. We employ the Gaussian random walk kernel for ABC-MCMC and ABC-SMC, with the standard deviation of 0.5 for  $\beta$  and 0.25 for  $\gamma$  (if a negative value is proposed, then we simply resample). We use the Euclidean norm for measuring the distance as follows:

$$d = \frac{1}{T} \sum_{t=1}^T \sqrt{(I_{sim}^t - I_{real}^t)^2 + (R_{sim}^t - R_{real}^t)^2}.$$

Since we are using the true priors for the ABC methods, we expect all of the methods to be very closely calibrated to the desired level of 0.9. Figure 7 shows that ABC-Rej and ABC-SMC are well-calibrated, but ABC-MCMC is not. As provided in Table 2, the ESS of ABC-MCMC is really tiny, indicating that samples are too correlated. ABC-Rej has the best average ESS value, followed by ABC-SMC. However, the median CPU-time was significantly higher for ABC-Rej by 4 times compared to that of ABC-SMC, implying it could take a long time for ABC-Rej to obtain samples. Overall, in terms of ESS per CPU-time, ABC-Rej performed the best.

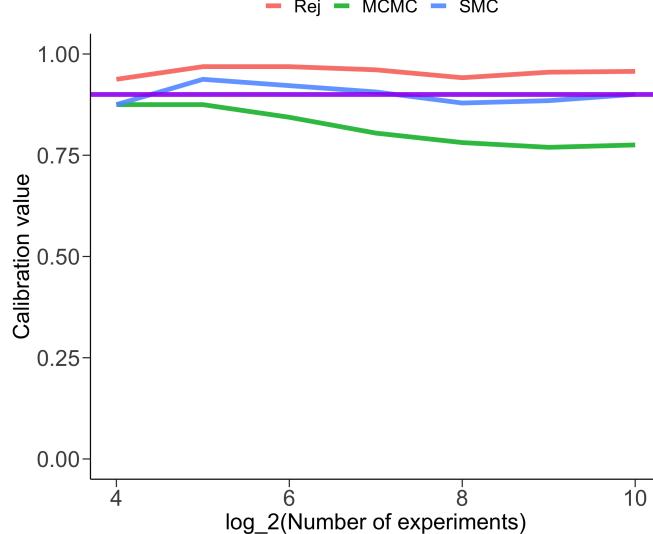


Figure 7: Bayesian calibration of ABC-Rej (red), ABC-MCMC (green), and ABC-SMC (blue) for a desired level of 0.9 (purple) with the SIR model.

Table 2: Simulation results for SIR model (best value bold-faced)

	ESS			CPU time (s)			ESS/time (1/s)		
	Rej	MCMC	SMC	Rej	MCMC	SMC	Rej	MCMC	SMC
Median	<b>250</b>	27	171	<b>0.33</b>	0.43	0.34	<b>749</b>	62	498
Mean	<b>361</b>	31	185	1.39	0.44	<b>0.35</b>	<b>868</b>	70	528

## E Bayesian calibration (Gelman et al., 2020)

Bayesian calibration is a simulation-based approach to validate Bayesian methods. After specifying a *true* distribution over the parameters, for each experimental, we draw a set of true parameters and then check whether an  $(1 - \alpha)\%$  credible interval of an estimated posterior distribution computed by a method contains the true parameters for  $\alpha \in (0, 1)$ . As the number of experimental runs increases, the average number of times that it does converges to  $(1 - \alpha)$  if the method is correctly implemented. This holds true for exact inference methods, such as MCMC. For ABC, this may not hold true since ABC is an approximate method for continuous data. However, if the approximation is good, then the calibration value should be close to the specified  $(1 - \alpha)$  level.

Bayesian calibration is a more rigorous way to check the validity of implemented Bayesian methods by exploring a myriad number of possible cases, as opposed to testing the methods on a small set of parameter values and measuring how well parameters are estimated.