A Tutorial on Approximate Bayesian Computation for Parameter Estimation in Systems Biology

Xin Liu and Mahesan Niranjan

***Abstract*—Intensively used inference algorithms such as Expectation-Maximization (EM), Markov chain Monte Carlo (MCMC) and Particle Filter (PF) require the evaluation of the likelihood, which might be intractable in some cases due to high dimensions. This incapability therefore motivates the need for likelihood-free approaches that are known as the approximate Bayesian computation (ABC) methods.**

Here we introduce various existing ABC algorithms in chronological order as they were developed and explain the advantages and potential drawbacks of each of them. By studying toy examples and a biological system, investigations show that all ABC algorithms perform inference tasks bypassing the likelihood evaluation, however the priority of methods depends on the complexity of the system of interest.

***Index Terms*—ABC, likelihood-free, chronological development and heat shock response model.**

I. Introduction

MANY biological phenomena are characterized by a set of ordinary differential equations (ODEs) in systems biology. Thorough understanding of such systems is achieved in a quantitative way via the solutions of these equations. Some unknown parameter values in them are tweaked by hand so that the system is best explained. In some scenarios, performances of such hand-tuning methods depend on the user experience, thus this elaboration may result inefficiency. Several methods have been proposed to tackle the parameter estimation problem in a systematic or probabilistic scheme. In this paper, we focus on introducing a set of likelihood-free methods and take a competitive look at the capabilities of seven members of this family.

The unknown system parameters have different definitions, such as reaction rate in biochemistry, mutation rate of coalescent tree in genetics and inflation rate in econometrics problem, etc. Several probabilistic methods have been used intensively, coupled with state-space model to solve the parameter estimation problem. Such methods claim to be capable of estimating as well as being able to infer the latent state via the noisy observation.

The techniques using Bayesian framework coupled with state-space model have to define two models, known as process model and output model. Generally, the process model describes the dynamics of system along with time, the output model is to quantify the observations related to the state of the system. In the Bayesian methods, the unknown parameter is always regarded as a state of the system with imposing zero dynamics so that this estimation could be involved with the evolution of the system. The information of the unknown parameter can be captured by the probability density function (pdf) of the artificial dynamics.

The popular estimation methods can be split into batch and sequential fashions. Batch methods include maximum likelihood (ML), Markov chain Monte Carlo (MCMC) etc., which infer unknown parameter by using the entire observations, when a new observation is received. Examples of work along this line are presented in [1]–[7].

As alternative, methods such as extended Kalman filter (EKF), unscented Kalman filter (UKF) and particle filtering (PF) are considered as efficient solutions by one-pass processing of the data if a new output is available rather than store and reprocess the existing data. Works presented in [8]–[13].

Aforementioned methods rely on the computation of likelihood to underpin the inference where the likelihood indicates how probability of observed output would be determined by a parameter value. This can be easily achieved via the output model when the additive noise distribution is Gaussian. However, this is a loose assumption to illustrate the capabilities of these methods, while variety of applications such as biological systems or genetics problems in reality are unable to provide the explicit form of noise. As when the dynamics of the system of interest is stochastic, the formulation of likelihood becomes intractable and therefore impossible to compute. A class of techniques using likelihood-free (or likelihood approximation) is known as approximate Bayesian computation (ABC) serving an alternative to the traditional likelihood-based methods.

ABC was originally designed to infer the coalescent time of DNA sequences from the human Y chromosome [14]. However, the simplest form of ABC method is concerned as an extremely inefficient algorithm. Therefore, authors proposed an inspirational concept in [15], which made use of summary statistics of data, instead of involving the raw dataset. As the illustration, authors applied their method to estimate the age of the most recent common ancestor of samples of DNA sequence from human Y chromosomes and human *Alu* sequences.

Even though the summary statistics can effectively extract the information about dataset and therefore opens the way to investigate problems in large scale. However, the method is still speed down by the condition that requires to produce the identical synthetic data. As the solution, authors [16] addressed such problem by introducing a brand new criterion. This criterion includes a metric function and a tolerance, where the metric function acts as the measurement for quantifying the similarity of the synthetic and data. If such quantity is less than the tolerance, then the current samples will be accepted

X. Liu and M. Niranjan are with the Department of Electronics and Computer Science, University of Southampton, Southampton, SO17 1GG UK (e-mail: xl1v09@ecs.soton.ac.uk; M.Niranjan@Southampton.ac.uk).

to represent posterior. In the work [16], authors successfully applied their method to analyze the demographic history of three human populations, which are the Basques, the Nuu-Chah-Nulth and the Biaka pygmies. Works [17]–[19] have done the inference problems in demographic history, population recombination rate and population history by employing this method and its similar variants.

In order to overcome the disadvantages of simple ABC methods, the sampling methods (e.g. MCMC) have been involved in the ABC framework [20], then a regression adjustment was further incorporated to ABC-MCMC so that more accurate inference can be achieved [21]. Authors successfully utilized ABC-MCMC associated with the regression adjustment method to probe the demographic problem of Chimpanzees [22].

The history of combining the ABC framework into sequential Monte Carlo method can track back to 2007, Sisson et al. [23] merged the partially rejection control method into ABC setting. From onward, ABC in the context of SMC becomes a very popular inference method for a lot of applications, such as the wireless network problem [24], model selection [25] and model inference [26], spatial extremes [27], inference based on epidemic model [28], etc,.

This article is organized as follows: We mathematically state the parameter estimation problem in section II. Three popular parametric/non-parametric inference methods including EKF and particle filter are introduction in section III. In section IV, we describe the family of ABC methods from the simplest form to the most recent ABC-SMC in the chronological order. Advantages and disadvantages of all considered ABC methods are illustrated via several toy examples. Finally, we comparative study the effectiveness of all mentioned ABC algorithms on parameter estimation by applying to a biological systems, which is known as the heat shock response model. The comparisons are carried out in terms of accuracy and computational expense. Summary of considered ABC methods are listed in the discussion section.

II. Parameter Estimation

To define the procedure of parameter estimation in the context of state-space model, we consider the state extension which regards the unknown parameter as the state of system and imposes the artificial dynamics on it [29]. By doing so, the evolution of extended state can be described as

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \boldsymbol{\theta}_t) + \boldsymbol{v}_t \tag{1}$$

where the $f(\cdot)$ forms the dynamics of interest, $\mathbf{x}_t$ is the previous sequence of state and $\boldsymbol{v}_t$ is the i.i.d noise sequence. Vector $\boldsymbol{\theta}$ consists of parameters in evolution. Dimensions of state and parameter are n and m respectively, thus, $\mathbf{x}_t$ and $\boldsymbol{\theta}$ can be further denoted as $\mathbf{x}_t = [x_{t,1}, \ldots, x_{t,n}]$ and $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_M]$. The likelihood-based algorithms estimate parameter $\boldsymbol{\theta}$ via the output of system.

$$\mathbf{y}_{t+1} = h(\mathbf{x}_{t+1}) + \boldsymbol{\omega}_{t+1} \tag{2}$$

where $h(\cdot)$ is the function to quantify how the current outputs $\mathbf{y}_{t+1}$ are related to the state sequence $\mathbf{x}_{t+1}$. Output, $\mathbf{y}_{t+1}$ is further corrupted by additive noise $\boldsymbol{\omega}_{t+1}$ which is generally assumed as a i.i.d noise sequence (the distribution is Gaussian). In particular algorithms such as MCMC and ML, the parameters are estimated by using the entire output sequence (i.e. $\mathbf{Y}_{1:t+1} = [y_1 \ldots y_{t+1}]$), while other likelihood-based methods recursively estimate parameter by only using current output $\mathbf{y}_{t+1}$.

Even though variety of likelihood-based algorithms are capable of simultaneously estimating the state and parameters, here we focus only on the parameter estimation and the state is considered as known. The parameter estimation in context of Bayesian is to construct a posterior by giving the sequences of state and output up to time index $t+1$, denoted as $p(\boldsymbol{\theta}_{t+1}|\mathbf{X}_{1:t+1}, \mathbf{Y}_{1:t+1})$. Note that in sequential paradigms, only current state and output are required for pdf as $p(\boldsymbol{\theta}_{t+1}|\mathbf{x}_{t+1}, \mathbf{y}_{t+1})$. Initially, the arbitrary guess $p(\boldsymbol{\theta}_0)$ needs to be made, which is generally called prior. Then, the computation of posterior $p(\boldsymbol{\theta}_{t+1}|\mathbf{x}_{t+1}, \mathbf{y}_{t+1})$ is consist of two steps as prediction and update.

The posterior distribution $p(\boldsymbol{\theta}_t, \mathbf{x}_t|\mathbf{y}_{1:t})$ at time index t is supposed to be available. Then, the prediction of pdf at time index $t+1$ can be obtained by following Chapman-Kolmogorov, i.e. equation. (3) .

Due to the evolution of system (1) follows Markov process, thus, in (3), term $p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{Y}_{1:t})$ is defined as $p(\mathbf{x}_{t+1}|\mathbf{x}_t)$, making use of dynamics equation (1) to compute. As the unknown parameters $\boldsymbol{\theta}$ are defined as variables of system, therefore, $p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t, \mathbf{Y}_{1:t}) = p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t)$ is equivalent to an imposed artificial dynamics of parameter, given as

$$p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t) = k(\boldsymbol{\theta}_t) \tag{4}$$

where the $k(\cdot)$ is the transition kernel for the parameter. In the update step, the prior distribution $p(\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1}|\mathbf{Y}_{1:t})$ obtained from prediction step will be used to calculate the the posterior distribution $p(\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1}|\mathbf{Y}_{1:t+1})$, such updating follows the Bayes's rule and is shown as:

$$p(\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1}|\mathbf{Y}_{1:t+1}) = \frac{p(\mathbf{y}_{t+1}|\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1}) p(\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1}|\mathbf{Y}_{1:t})}{p(\mathbf{y}_{t+1}|\mathbf{Y}_{1:t})} \tag{5}$$

where $p(\mathbf{y}_{t+1}|\mathbf{Y}_{1:t})$ is known as the normalizing constant and is usually ignored in the estimation problem. $p(\mathbf{y}_{t+1}|\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1})$ in (5) is known as likelihood function. With the output function (2) and the additive Gaussian noise $\boldsymbol{\omega}$, likelihood is computed as:

$$p(\mathbf{y}_{t+1}|\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1}) \propto \mathcal{N}(\mathbf{y}_{t+1} - h(\mathbf{x}_{t+1}, \boldsymbol{\theta}_{t+1}); 0, \boldsymbol{\sigma}^2\mathbf{I}) \tag{6}$$

Consequently, the joint posterior distribution of parameter and state is able to be obtained by using the current output $\mathbf{y}_{t+1}$ and the prior distribution.

Notice that almost all systems are continuous-time dynamics in reality, for the sake of implementation, here, we assume that the system of interest is observed at discrete times. One possible way to discretization is to numerically integrate the system process with respect to the time phase that observations are made, associated with the previous posterior distribution

$$p(\boldsymbol{\theta}_{t+1}, \mathbf{x}_{t+1}|\mathbf{y}_{1:t}) = \int p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{y}_{1:t}) p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t, \mathbf{y}_{1:t}) p(\mathbf{x}_t, \boldsymbol{\theta}_t|\mathbf{y}_{1:t}) d\mathbf{x}_t d\boldsymbol{\theta}_t \tag{3}$$

of parameter [29]:

$$\mathbf{x}_{t+1|t} = \mathbf{x}_{t|t} + \int_t^{t+1} f(\mathbf{x}(\tau), \boldsymbol{\theta}_{t|t}) d\tau \tag{7}$$

where $\mathbf{x}_{t+1|t}$ and $\boldsymbol{\theta}_{t+1|t}$ are termed as the prior state and parameter distributions, obtained by propagating the previous posterior estimates (i.e. $\mathbf{x}_{t|t}$ and $\boldsymbol{\theta}_{t|t}$) through the evolutions of system (1) and parameter (4).

The classical Kalman filter is considered as the optimal Bayesian solution to tackle this problem, however, its effectiveness is restricted by Gaussian process. Implausible estimation will be made with nonlinear and non-Gaussian system. As the alternative solutions, extended Kalman filter and particle filter, two popular algorithms will be described in the next section.

III. Suboptimal Algorithms

If the system is linear or Gaussian, then evolution of system (1) and output function (2) can be rewritten as

$$\mathbf{x}_{t+1} = \mathbf{F}\mathbf{x}_t + \boldsymbol{v}_t \tag{8}$$

$$\mathbf{y}_{t+1} = \mathbf{H}\mathbf{x}_{t+1} + \boldsymbol{\omega}_{t+1} \tag{9}$$

Linear functions of process and output are matrices $\mathbf{F}$ and $\mathbf{H}$. Noises $\boldsymbol{v}_t$ and $\boldsymbol{\omega}_{t+1}$ follow Gaussian distributions with zero means, while covariance matrices are $\boldsymbol{Q}_t$ and $\boldsymbol{R}_{t+1}$. Kalman filter provides an equivalent expressions to (3) and (5) in probabilistic notation as

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t+1}; \mathbf{F}\boldsymbol{x}_t, \boldsymbol{Q}_t) \tag{10}$$

$$p(\mathbf{y}_{t+1}|\mathbf{x}_{t+1}) = \mathcal{N}(\mathbf{y}_{t+1}; \mathbf{H}\mathbf{x}_{t+1}, \boldsymbol{R}_{t+1}) \tag{11}$$

Kalman filter holds the optimality if the system of interest is linear or Gaussian. However, when either of these two conditions are violated, Kalman filter will poorly perform. We consider two popular solutions in this section, one is a derivative-based algorithm, known as the extended Kalman filter. Another is more widely used and also likelihood-based method called particle filter.

A. Extended Kalman Filter

The extended Kalman filter, approximates non-linearities of dynamics (1) and output (2) via local linearization through Taylor series expansion. Close form of update equations are obtained if the truncation is to first- and second-order terms. Nevertheless, the most widespread use is to truncate as first-order (gradient) term, also representing here.

In order to recursively estimate, assumption of the initial conditions $\mathbf{x}_0$, $\boldsymbol{\theta}_0$ and error covariance matrix $\mathbf{P}_0$ need to be defined.

And the prior distributions of state and parameter are described as $\mathbf{x}_{t+1|t}$ and $\boldsymbol{\theta}_{t+1|t}$, the ways to calculate are shown in (7) and (4), respectively. Notice that a simple scheme, known as the random walk is always used as transition kernel for moving parameter (4), denoting as:

$$\begin{aligned} p(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t) &= k(\boldsymbol{\theta}_t) \\ &= \boldsymbol{\theta}_t + \boldsymbol{\omega}_t \end{aligned} \tag{12}$$

where $\boldsymbol{\omega}_t$ is the Gaussian noise, such a random walk scheme helps to move the arbitrary initial guesses to achieve the true values of interest. Several extensive discussions for choosing this move scheme can be found in literature [30]–[32].

Moreover, the current prior error covariance matrix termed $\mathbf{P}_{t+1|t}$, obtained by integrating a differential Lyapunov equation whose initial condition is the posterior error covariance of previous time instant $\mathbf{P}_{t|t}$, while time interval of the integral is $[t\ t+1]$. Algorithm presented here was proposed in [8] and [33].

$$\mathbf{x}_{t+1|t} = f(\mathbf{x}_{t|t}) + \boldsymbol{v}_t \tag{13}$$

$$\boldsymbol{\theta}_{t+1|t} = \boldsymbol{\theta}_{t|t} + \boldsymbol{\omega}_t \tag{14}$$

$$\mathbf{P}_{t+1|t} = \int_t^{t+1} (\boldsymbol{A}_{t+1}\mathbf{P}_{t|t} + \boldsymbol{P}_{t|t}\boldsymbol{A}_{t+1}^T + \boldsymbol{Q}_t) dt \tag{15}$$

where $\boldsymbol{Q}_t$ is the covariance matrix of noise in evolution of system, defined in (10). $\boldsymbol{A}_{t+1}$ is the Jacobian matrix of system dynamics $f(\cdot)$, obtained as:

$$\boldsymbol{A}_{t+1} = \begin{bmatrix} \frac{\partial f_1}{\partial_{x_1}} & \frac{\partial f_1}{\partial_{x_2}} & \cdots & \frac{\partial f_1}{\partial_{x_n}} & \frac{\partial f_1}{\partial_{\theta_1}} & \frac{\partial f_1}{\partial_{\theta_2}} & \cdots & \frac{\partial f_1}{\partial_{\theta_M}} \\ \vdots & & \ddots & \vdots & \vdots & & \ddots & \vdots \\ \frac{\partial f_n}{\partial_{x_1}} & \frac{\partial f_n}{\partial_{x_2}} & \cdots & \frac{\partial f_n}{\partial_{x_n}} & \frac{\partial f_n}{\partial_{\theta_1}} & \frac{\partial f_n}{\partial_{\theta_2}} & \cdots & \frac{\partial f_n}{\partial_{\theta_M}}. \end{bmatrix} \tag{16}$$

For updating the prediction (evolutions (19), (20) and (21)), Kalman gain termed $\boldsymbol{L}_{t+1}$ need be computed:

$$\boldsymbol{L}_{t+1} = \mathbf{P}_{t+1|t}\boldsymbol{H}_{t+1}^T(\boldsymbol{H}_{t+1}\mathbf{P}_{t+1|t}\boldsymbol{H}_{t+1}^T + \boldsymbol{R})^{-1} \tag{17}$$

where $\boldsymbol{R}_t$ is the covariance matrix of noise in output model, defined in (11) and $\boldsymbol{H}_{t+1}$ is the Jacobian matrix of the output model $h(\cdot)$, defined as:

$$\boldsymbol{H}_{t+1} = \begin{bmatrix} \frac{\partial h_1}{\partial_{x_1}} & \frac{\partial h_1}{\partial_{x_2}} & \cdots & \frac{\partial h_1}{\partial_{x_n}} & \frac{\partial h_1}{\partial_{\theta_1}} & \frac{\partial h_1}{\partial_{\theta_2}} & \cdots & \frac{\partial h_1}{\partial_{\theta_M}} \\ \vdots & & \ddots & \vdots & \vdots & & \ddots & \vdots \\ \frac{\partial h_n}{\partial_{x_1}} & \frac{\partial h_n}{\partial_{x_2}} & \cdots & \frac{\partial h_n}{\partial_{x_n}} & \frac{\partial h_n}{\partial_{\theta_1}} & \frac{\partial h_n}{\partial_{\theta_2}} & \cdots & \frac{\partial h_n}{\partial_{\theta_M}}. \end{bmatrix} \tag{18}$$

The posterior state, parameter and error covariances are updated as:

$$\mathbf{x}_{t+1|t+1} = \mathbf{x}_{t+1|t} + \boldsymbol{L}_{t+1}(\mathbf{y}_{t+1} - h(\mathbf{x}_{t+1|t}, \boldsymbol{\theta}_{t+1|t})) \tag{19}$$

$$\boldsymbol{\theta}_{t+1|t+1} = \boldsymbol{\theta}_{t+1|t} + \boldsymbol{L}_{t+1}(\mathbf{y}_{t+1} - h(\mathbf{x}_{t+1|t}, \boldsymbol{\theta}_{t+1|t})) \tag{20}$$

$$\begin{aligned} \mathbf{P}_{t+1|t+1} &= (\boldsymbol{I} - \boldsymbol{L}_{t+1}\boldsymbol{H}_{t+1})\boldsymbol{P}_{t+1|t}(\boldsymbol{I} - \boldsymbol{L}_{t+1}\boldsymbol{H}_{t+1})^T \\ &\quad + \boldsymbol{L}_{t+1}\boldsymbol{R}_{t+1}\boldsymbol{L}_{t+1}^T \end{aligned} \tag{21}$$

The EKF algorithm is summarized as Algorithm 1:

Algorithm 1 Extended Kalman filter

- Initialize $\mathbf{x}_0$, $\boldsymbol{\theta}_0$ and $\mathbf{P}_0^+$
- FOR $t = 1, \ldots, T$
  - Compute $\mathbf{x}_{t+1|t}$ using (13)
  - Compute $\boldsymbol{\theta}_{t+1|t}$ using (14)
  - Compute $\mathbf{P}_{t+1|t}$ using (15)
  - Calculate the Kalman gain $\boldsymbol{L}_{t+1}$ according to (17)
  - Update the state posterior $\mathbf{x}_{t+1|t+1}$ by (19)
  - Update the parameter posterior $\boldsymbol{\theta}_{t+1|t+1}$ by (20)
  - Update the covariance posterior $\mathbf{P}_{t+1|t+1}$ according to (21)
- END FOR

B. Particle Filter

Even though EKF utilizes first-order partial derivative as linearization for non-linear functions $f(\cdot)$ and $h(\cdot)$, however, EKF poorly performs when the interest is bimodal or heavily skewed. Such incapability may be due to the fact that EKF still approximates the posterior density via Gaussian distribution. Moreover, Taylor expansion cannot capture all important information of system via the transformation coefficient matrices (Jacobian matrices with first order filters and Hessian matrices with second order filters). For example, the system might be jump-linear, in which the parameters can change abruptly [34]. Thirdly, in some cases, the calculation for an analytical solution of first- or second-order partial derivative may contain human errors. Unscented Kalman filter (UKF), another well-known variant of Kalman filter, has been proposed as a timely and natural extension of EKF, which deterministically selects a minimum number of particles in the prior distribution and uses such set of points to represent the dynamics. Although UKF still makes use of Gaussian for the approximation, however, it does not need to calculate the partial derivative to linearize. In the work [10], authors instantiated UKF on transcriptional regulatory networks with Hill equation [35] and JAK-STAT signaling pathways [36]. Details and implementations of UKF can be found in [10], [37], [38]. We therefore review a Monte Carlo based technique, known as particle filter (PF).

Nevertheless, UKF is still incapable of tackling high dimension estimation task. Thus, sequential Monte Carlo methods, more widely known as Particle Filters (PF), offer a more powerful way to estimate parameter and infer the latent state of high dimensional systems. Several applications and reviews are represented in [39], [40] and [41]. The family of Monte Carlo algorithms make the approximation by averaging all particles from the target distribution, generally, such distribution is impossible to compute the analytical solution. The core algorithmic step in particle filtering is importance sampling, that of generating identically and independently distributed particles from a prior density of convenience, with associated weighting of these particles. In the sequential setting, PF offers recursive ways for updating these weights and then propagates particles through the dynamical system. A crucial issue in implementation is the problem of sample degeneracy, that is all samples collapse into one place after several iterations, and several tricks to circumvent these exist in the literature. A concise summary of these algorithms can be found in the tutorial paper by [40]. A pseudo-code of the PF is given in Algorithm 2.

Algorithm 2 Particle Filter

Initialization, t = 0

- FOR $i = 1, \ldots, N$
  - Sample $\mathbf{x}_0^i \sim p(\boldsymbol{x}_0)$, $\boldsymbol{\theta}_0^i \sim p(\boldsymbol{\theta}_0)$ and set t = 1
- END FOR

Importance sampling

- FOR $i = 1, \ldots, N$
  - Sample $\tilde{\mathbf{x}}_t^i \sim p(\mathbf{x}_t|\mathbf{x}_{t-1}^i)$ and $\tilde{\boldsymbol{\theta}}_t^i \sim p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1}^i)$
  - Evaluate the importance weights $\tilde{\boldsymbol{w}}_t^i = p(\mathbf{y}_t|\tilde{\mathbf{x}}_t^i, \tilde{\boldsymbol{\theta}}_t^i)$
- END FOR

Normalize the importance weights

- FOR $i = 1, \ldots, N$
  - $\tilde{\mathbf{w}}_t^i = \tilde{\mathbf{w}}_t^i / \sum_{i=1}^N \tilde{\mathbf{w}}_t^i$
- END FOR

Selection step

- Resample with replacement N particles $(\mathbf{x}_t^i; i = 1, \ldots, N)$ and $(\boldsymbol{\theta}_t^i; i = 1, \ldots, N)$ from the set $(\tilde{\mathbf{x}}_t^i; i = 1, \ldots, N)$ and $(\tilde{\boldsymbol{\theta}}_t^i; i = 1, \ldots, N)$ according to the importance weights

Set t $\rightarrow$ t+1 and go to Importance sampling step.

As the non-parametric method, PF has been intensively used to solve the problems such as robotic tracking, financial prediction and systems biology. Since Kingman [42] attributed the definitive formalization of coalescent theory, population genetics became a hot issue in the community, however, methods as EKF and PF are incapable to solve the problems in this area, therefore, motivating a new type method, known as the approximate Bayesian computation.

IV. Approximate Bayesian Computation Algorithms

PF approaches the target density by constructing distribution with fittest samples which are randomly drawn, while the fitnesses of which are determined by likelihood. As mentioned in section II, when the output is corrupted with an additive Gaussian noise, it is able to derive the explicit likelihood function (6).

However, several processes in real world [43]–[45] are characterized in a stochastic manner, therefore their likelihoods are impossible to compute. Moreover, even the deterministic systems, the computation of likelihoods are still intractable if the probability density function of corrupted noise is unknown.

Such incapabilities motivate the likelihood-free concept, referred to the approximate Bayesian computation (ABC). ABC is the data-driven method, inferred the parameter of interest by only using the data generated from the given model, instead of calculating likelihood.

A. Basic ABC methods

The basic idea in ABC algorithms is to sample the unknown from a prior distribution, $\theta \sim \pi(\theta)$, synthesize data from the model under study, $\mathbf{X}^* \sim f(\mathbf{x}_0, \theta^*)$, where $\mathbf{x}_0$ is the initial condition and $f(\cdot,\cdot)$ is the model, and accept θ^* as a sample for the posterior if the synthesized data $\mathbf{X}^*$ is close enough in some sense to the observations $\mathbf{X}$. In our discussion with Systems Biology models, we will focus on $f(\cdot,\cdot)$ being a set of ordinary differential equations which can be numerically integrated. We will also use Euclidean distance between the synthesized and the observations as measure of discrepancy. Variety of the ABC frameworks are derived by considering different sampling strategies and adapting the threshold at which acceptance decisions are made.

In order to retain the consistency of notation throughout the descriptions in this paper, we assume that the dimension of parameter in system is one, i.e. $D_p = 1$, therefore, the scalar θ_t denotes a sample of the parameter at the time instant t and the collection of samples is described as $\boldsymbol{\theta}$. However, in most real world situations, the dimension of parameters in system is always greater than one, as a result, the scalar θ_t is naturally extended to a vector, denoted as $\boldsymbol{\theta}_t$, and the collection of samples becomes a matrix, given as $\boldsymbol{\Theta}$.

1) ABC-Rejection: In its earliest form [14], the generated particle θ^* was accepted only if $\mathbf{X}^*$ was identical to the observations $\mathbf{X}$. It became immediately evident that this is an inefficient procedure because thousands of trails needed to be performed before accepting one of the generated particles. A modification to the scheme, introduced by [46] was to define a threshold ϵ and accept particles when the discrepancy between $\mathbf{X}^*$ and $\mathbf{X}$ was within this. This variant of the method is normally referred to as the ABC-rejection algorithm shown in Algorithm 3.

Algorithm 3 ABC-rejection

`Input`: $\pi(\theta)$, ϵ, $\mathbf{x}_0$, N_{rej}, $\mathbf{X}$, $\rho(\cdot,\cdot)$ and $f(\cdot,\cdot)$.
`Output`: $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_{N_{reg}}\}$
n=1
Repeat
1. Draw $\theta^* \sim \pi(\theta)$
2. Synthesize $\mathbf{X}^* \sim f(\mathbf{x}_0, \theta^*)$
3. Evaluate discrepancy $d = \rho(\mathbf{X}^*, \mathbf{X})$
4. **if** $d \leq \epsilon$ **then**
5. $\quad \theta_n = \theta^*$
6. $\quad n = n + 1$
7. **end if**

until $n = N_{rej}$

The procedure of ABC-rejection is illustrated in Figure 1, which shows the steps taken by ABC-rejection to approach the posterior distribution.

The function $\rho(\cdot,\cdot)$ in Algorithm 3 quantifies a distance between synthesized the true observations. The Euclidean distance is often chosen for this metric function in the context of systems biology, which is given by

$$\rho(\mathbf{X}^*, \mathbf{X}) = \sum_{i=1}^{N_{OT}} \|\mathbf{x}_i^* - \mathbf{x}_i\|, \tag{22}$$

where $\|\mathbf{x}_i^* - \mathbf{x}_i\|$ denotes the *norm* of error between $\mathbf{x}_i^*$ and $\mathbf{x}_i$, and N_{OT} is the number of data points of observations. Intuitively, tolerance ϵ plays a crucial role in performing the ABC-rejection. If ϵ approaches zero, the approximated posterior distribution becomes infinitesimally close to the target distribution. This is achieved at the expense of more computation, therefore, efficiency becomes an issue. Likewise high precision cannot be expected with tolerance ϵ increasing considerably.

Example 1 *Take the deterministic Lotka-Volterra model [47], [48] as an example The ODEs of this Lotka-Volterra model are described as*

$$\dot{\mathbf{x}}_1 = f(x_1) = \alpha x_1 - x_1 x_2, \quad \dot{\mathbf{x}}_2 = f(x_2) = x_1 x_2 - \beta x_2, \tag{23}$$

where x_1 and x_2 are two species in a system. In particular, observations of the system are corrupted by Gaussian noise which is generated from $\mathcal{N}(0, 0.05)$. The time length for synthesizing the pseudo-observations is 100 min, sampling at regular intervals of 0.2 minutes, that is the number of data for representing the pseudo-observations is 500. We generate the observations with the true model: $\alpha = \beta = 0.5$. Starting from the same prior distribution for picking particles, i.e $\pi(\boldsymbol{\theta}) \sim \mathcal{U}(0.2, 0.9)$, we examine the performance of ABC-rejection under different epsilons: $\epsilon = 300, 100, 20, 10$. The inference under each epsilon is carried out 10 times. The results of inferences are shown in Figure 2 where the ability of ABC-rejection to estimate parameters is varying considerably with respect to the epsilon value. The performance of ABC-rejection in terms of efficiency under different tolerances is summarized and listed in Table IV-A1. We can conclude that if the coarse acceptance criterion is employed, the algorithm appears as an efficient method with high acceptance rate, however, a few undesired particles are also accepted to represent the posterior. In contrast, the particles are able to narrowly circle around the true point when the harsh tolerance is taken; yet, this high precision requires more computational expense and the decline of acceptance rate is evident.

ϵ	Computational time (second)	Acceptance rate
300	4.434±0.11	100%±0
100	4.493±0.14	95.6%±1.6
20	154.4±18.1	2.43%±0.2
10	727.6±35.3	0.05%±0.02

This example illustrates the capability of ABC-rejection on parameter estimation without likelihood evaluation, however, it can be challenging to strike a balance between computational efficiency and accuracy.

2) ABC-Regression: The tradeoff made for acceptance rate and precision limits the widespread use of ABC-rejection. [49] introduced the modification of standard rejection ABC method associated with a local regression adjustment. This so-called ABC-regression method and ABC-rejection appear

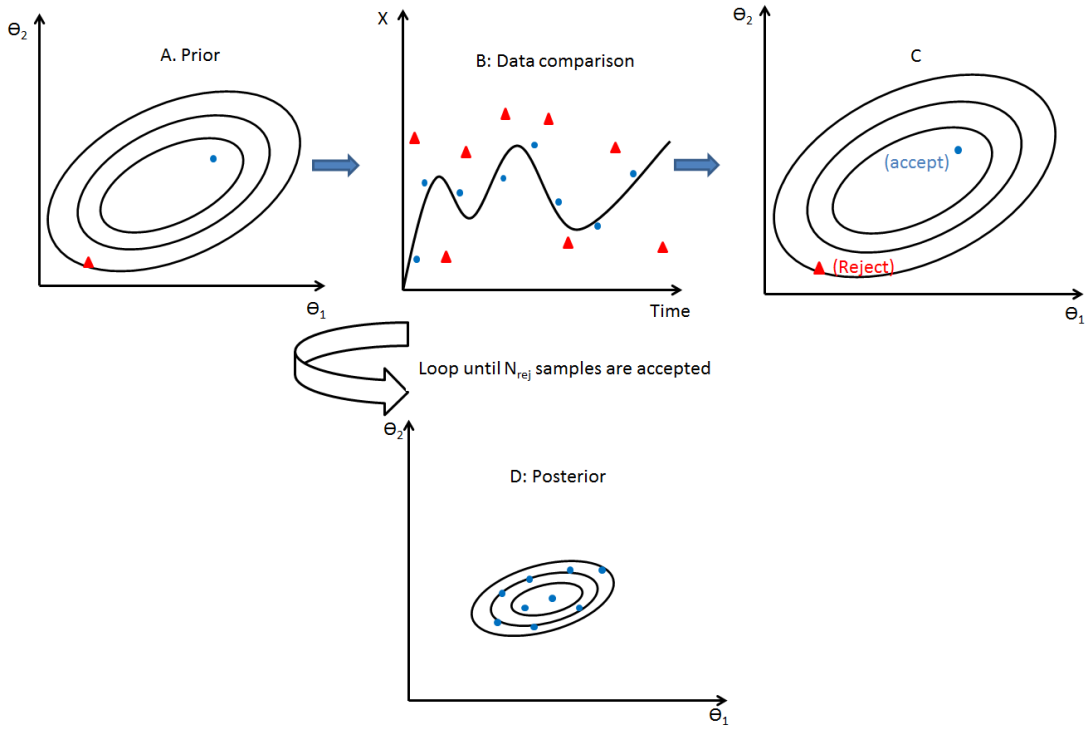


Fig. 1. The procedure of ABC-Rejection algorithm: The graph.A shows the prior distribution of parameters (there are two parameters in this illustration i.e. $\boldsymbol{\theta} = [\theta_1, \theta_2]$) where the dot and triangle are samples drawn from the prior distribution. It is easy to see from the graph.B that is the simulation yielded from the dot parameter set is sufficiently close to the true data, thus the dot sample for parameter is accepted. In contrast the simulation from the triangle set mimic poorly the true data, and are therefore rejected. By running through the process, the posterior distribution is obtained and shown in the graph.D, and turns out to be narrower than the prior distribution.

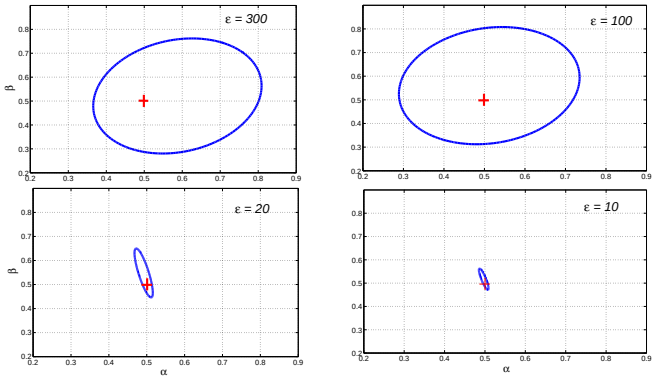


Fig. 2. Illustration of the posterior distribution $p(\boldsymbol{\theta}|\mathbf{X})$ *(blue ellipses)* obtained under different tolerances ϵ, where the small *red cross* implies the point of true parameters.

fundamentally in collecting samples for approximating the posterior distribution. With making use of the local correction, ABC-regression is allowed to afford a relatively large tolerance ϵ. The innovation of this algorithm is in characterizing the relationship between the collection of accepted samples and their corresponding discrepancies via a linear regression model, given as

$$\widehat{\boldsymbol{\theta}} = \mathbf{D}\boldsymbol{\beta} + \boldsymbol{\omega}, \tag{24}$$

where $\widehat{\boldsymbol{\theta}} \in \mathbb{R}^{N_{reg} \times D_p}$ are the accepted samples from the rejection scheme. $\boldsymbol{\beta} \in \mathbb{R}^{q \times D_p}$ are the regression coefficients and q is the dimension of the summary statistics which is introduced below. $\boldsymbol{\omega} \in \mathbb{R}^{N_{reg} \times D_p}$ denotes the unobserved random variables and $\mathbf{D} \in \mathbb{R}^{N_{reg} \times q}$ is the matrix indicating the discrepancy between the pseudo-observations and the true dataset, defined as

$$\mathbf{D} = \begin{bmatrix} s^*_{1,1} - s_1 & s^*_{1,2} - s_2 & \dots & s^*_{1,q} - s_q \\ s^*_{2,1} - s_1 & s^*_{2,2} - s_2 & \dots & s^*_{2,q} - s_q \\ s^*_{3,1} - s_1 & s^*_{3,2} - s_2 & \dots & s^*_{3,q} - s_q \\ \vdots & \vdots & \vdots & \vdots \\ s^*_{n,1} - s_1 & s^*_{n,2} - s_2 & \dots & s^*_{n,q} - s_q \end{bmatrix}. \tag{25}$$

We note that this discrepancy matrix is calculated using the summary statistics $\boldsymbol{s} \in \mathbb{R}^{q \times 1}$, instead of the raw data $\mathbf{x}$. This is because the ABC methods were motivated initially to tackle the inference problem in population genetics, a field often making use of very large datasets. When each individual point in dataset is used to evaluate the discrepancy matrix, the computational complexity becomes unaffordable. Summary statistics[1], capturing dynamical features as much as possible with minimum effort, are instead used to calculate the matrix $\mathbf{D}$.

The linear regression model shown in equation 24 was taken by [49] to correct the result from the rejection scheme, in

[1]In systems biology, statistical measures such as mean, median, mode, standard deviation and skewness are widely used. In population genetics, more information can be used as summary statistics. For example, the number and frequency of segregating sites, number of population pairs, Shanon's index and variance of allele length in each population.

which the updated inference is adopted for the term $\boldsymbol{\omega}$ in model and the previous result is corrected by

$$\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}} - \mathbf{D}^{\mathrm{T}}\boldsymbol{\beta} \tag{26}$$

$$= \widehat{\boldsymbol{\theta}} - (\mathbf{S}^* - \mathbf{S})^{\mathrm{T}}\boldsymbol{\beta}, \tag{27}$$

where $\mathbf{S} \in \mathbb{R}^{q \times N_{reg}}$ is the summary statistics matrix of the true dataset, which is produced by replicating from the vector **s**. The value of the regression coefficients matrix $\boldsymbol{\beta}$ is often determined via the least-squares estimator, in which the sum of squared residuals (i.e. the unobserved random variables $\boldsymbol{\omega}$) is given as

$$SS(\boldsymbol{\beta}) = \sum_{n=1}^{N_{reg}} \hat{\theta}_n - (\boldsymbol{s}^* - \boldsymbol{s})^{'}\boldsymbol{\beta}$$

$$= (\widehat{\boldsymbol{\theta}} - \mathbf{D}\boldsymbol{\beta})^{\mathrm{T}}(\widehat{\boldsymbol{\theta}} - \mathbf{D}\boldsymbol{\beta}), \tag{28}$$

then the regression coefficients $\boldsymbol{\beta}$ can be estimated by first differentiating $SS(\boldsymbol{\beta})$ with respect to $\boldsymbol{\beta}$, given as

$$\frac{\partial SS(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -2(\widehat{\boldsymbol{\theta}} - \mathbf{D}\widehat{\boldsymbol{\theta}})\mathbf{D}^{\mathrm{T}} = 0. \tag{29}$$

Then setting the equation 29 to zero, we have

$$\boldsymbol{\beta} = (\mathbf{D}^{\mathrm{T}}\mathbf{D})^{-1}\mathbf{D}^{\mathrm{T}}\widehat{\boldsymbol{\theta}}, \tag{30}$$

solution of $\boldsymbol{\beta}$ is obtained.

Moreover, in practice, errors (components in **D**) are not always equally allocated around the 'best-fitting line' for the entire $\widehat{\boldsymbol{\theta}}$, violating model assumptions of homoscedasticity[2]. To handle heteroscedasticity[3], in ABC-regression, the sample from rejection-sampling method $\widehat{\theta}_i$ is weighted by the Epanechnikov kernel, defined as

$$w_i = K_\epsilon(d_i) = \begin{cases} c\epsilon^{-1}(1 - (d_i/\epsilon)^2) & d_i \leq \epsilon \\ 0 & d_i \geq \epsilon \end{cases} \tag{31}$$

where d_i is the discrepancy between $\boldsymbol{s}_i^*$ and $\boldsymbol{s}_s$, c is the normalizing constant and ϵ is the tolerance value. Other kernel functions, for instance, the Gaussian kernel could be adopted for weighting samples. The advantage of Epanechnikov kernel is due to an initial smooth decrease, falling sharply to zero as d_i increases, therefore, few small non-zero values are assigned to weights [49], [50].

The weighted least squares is naturally extended from the original least squares estimator, which determines the regression coefficients matrix $\boldsymbol{\beta}$ by minimizing the sum of squared residuals associated with their corresponding weights, given as

$$\boldsymbol{\beta} = (\mathbf{D}^{\mathrm{T}}\mathbf{W}\mathbf{D})^{-1}\mathbf{D}^{\mathrm{T}}\mathbf{W}\boldsymbol{\theta}, \tag{32}$$

where the i^{th} diagonal element of the weights matrix **W** is taken by $K_\epsilon(d_i)$. ABC-regression can be carried out with the steps shown in Algorithm 4.

Example 2 *Let ABC-regression work through the deterministic Lotka-Volterra model as a fair comparison, to illustrate its advantage over ABC-rejection. We examine the performances of these two algorithms with different values of of ϵ and combinations of summary statistics. Mean, variance and median are captured as the summary statistics for the dataset; By making use ϵ as 20, 2 and 0.2, we progressively make the acceptance of particle harder. The results obtained using summary statistics are shown in Figure 3. It is easy to observe that the estimates from ABC-rejection roughly center around the true values, and when the tolerance becomes tighter the precision of inference from rejection scheme is increased greatly. Moreover, considering the identical tolerance $\epsilon = 20$, the accuracy of inference from ABC-rejection is lower than the results shown in Figure 2. This is due to the discrepancy being evaluated based on the summary statistics in this example, instead of the raw dataset adopted in the previous study. Surprisingly, the expected improvement from regression adjustment is only evident in the case where mean is taken as the summary statistics. When the mean, variance and median are adopted simultaneously for summary statistics, ABC-regression performs worse than ABC-rejection, regardless of the value of tolerance ϵ. This is known as the curse of dimensionality [51]. The accuracy and reliability of ABC-regression decreases rapidly with increasing number of summary statistics, therefore, such problem is the major hindrance for the successful inference by ABC-regression [52], [53]. In addition, the informative prior distribution is considered, where samples are generated from $\mathcal{U}(0.2, 0.9)$, while the true values of parameters are 0.5. We also note that, in Lotka-Volterra model, the periodicity of species is irrelevant to parameter values and dependence only appears on the magnitude. Consequently, the gain of choosing mean as summary statistics can not be expected generally.*

3) ABC-MCMC: Although the local regression process has the effectiveness for correcting estimate with only a negligible computational cost, an informative prior must be set for ABC-rejection and ABC-regression. When samples are generated

Algorithm 4 ABC-regression

Input: $\pi(\theta)$, ϵ, $\mathbf{x}_0$, N_{reg}, **s**, $\eta(\cdot)$, $\rho(\cdot,\cdot)$ and $f(\cdot,\cdot)$.
Output: $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_{N_{reg}}\}$

n=1
Repeat
1. Draw $\theta^* \sim \pi(\theta)$
2. Synthesize $\mathbf{X}^* \sim f(\mathbf{x}_0, \theta^*)$
3. Calculate summary statistics $\mathbf{s}^* = \eta(\mathbf{X}^*)$
4. Evaluate discrepancy $d^* = \rho(\mathbf{s}^*, \mathbf{s})$
5. **if** $d^* \leq \epsilon$ **then**
6. $\widehat{\theta}_n = \theta^*$, $\mathbf{S}_n^* = \mathbf{s}^*$ and $\mathbf{d}_n = d^*$
7. $n = n + 1$
8. **end if**

until $n = N_{reg}$

9. Evaluate weight vector $\boldsymbol{w} : 1 \times N_{reg}$ using equation 31
10. Calculate regression coefficient matrix $\boldsymbol{\beta} : q \times D_p$ according to equation 32
11. Replicate $\mathbf{s} : q \times 1 \rightarrow \mathbf{S} : q \times N_{reg}$
12. Compute $\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}} - (\mathbf{S}^* - \mathbf{S})^{\mathrm{T}}\boldsymbol{\beta}$

[2]Homoscedasticity means that the distance between each data point and 'best-fitting line' is identical.

[3]Heteroscedasticity means that the distance between each data point and 'best-fitting line' is subject to change.

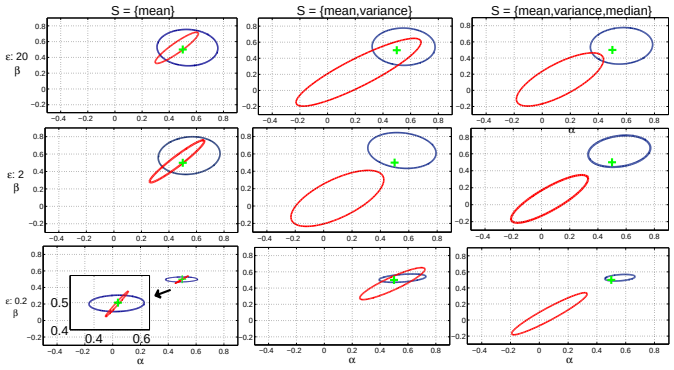


Fig. 3. Illustrations of the the posterior distribution $p(\boldsymbol{\theta}|\mathbf{X})$ from ABC-rejection and ABC-regression. The panels at *columns* show $p(\boldsymbol{\theta}|\mathbf{X})$ under three combinations of summary statistics: $\mathbf{s} = \{\text{mean}\}$, $\{\text{mean, variance}\}$ and $\{\text{mean, variance, median}\}$. The panels at *rows* denote the estimations under three tolerance levels: $\epsilon = \{20,\ 2,\ 0.2\}$. The results from ABC-rejection are shown in *(blue ellipse)* and *(red ellipse)* contours denote ABC-regression, point of the true parameter values is indicated as the *green cross*.

from a non-informative prior distribution, thousands of attempts will be rejected leading to inefficiency. As a solution of this difficulty, [20] performed MCMC algorithm in the ABC setting, namely the ABC-MCMC.

In ABC-MCMC, a Markov chain transition kernel $q(\theta^*|\theta_t)$ of invariant distribution $p(\theta_t|\mathbf{X})$ is allowed to apply on each sample so that $\int q(\theta^*|\theta_t)p(\theta_t|\mathbf{X}) = p(\theta^*|\mathbf{X})$, leading to the samples still being distributed according to the posterior of interest [54]. When the current distance between pseudo-observations and true dataset is less than the tolerance ϵ, the proposal is taken as the sample for parameter with an acceptance probability, defined as

$$h(\theta^*, \theta) = 1 \wedge \frac{p(\theta^*)q(\theta|\theta^*)}{p(\theta)q(\theta^*|\theta)} = \min\left(1, \frac{p(\theta^*)q(\theta|\theta^*)}{p(\theta)q(\theta^*|\theta)}\right). \tag{33}$$

The steps for executing ABC-MCMC algorithm are given in Algorithm 5.

It can be seen that, apart from the acceptance criterion conducted by ABC, the proposed samples must additionally be accepted according to the MH acceptance probability. The use of this acceptance probability makes the Markov chain satisfy the detailed balance condition, and guarantees its convergence to the stationary distribution $p(\theta|\mathbf{X})$. We prove this in what follows, and these standard derivations are described in [20].

Taking the notations previously mentioned, the detailed balance can be described

$$p(\theta|\mathbf{X})r(\theta \to \theta^*) = p(\theta^*|\mathbf{X})r(\theta^* \to \theta), \tag{34}$$

where $r(\theta \to \theta^*)$ is the MH transition mechanism, given as

$$r(\theta \to \theta^*) = k(\theta^*|\theta)p(\mathbf{X}|\theta^*)h(\theta, \theta^*). \tag{35}$$

Substituting equation 35 into the left-hand side of the detailed balance condition, we have

$$\begin{aligned} p(\theta|\mathbf{X})r(\theta \to \theta^*) &= p(\theta|\mathbf{X})k(\theta^*|\theta)p(\mathbf{X}|\theta^*)h(\theta, \theta^*) \\ &= \frac{p(\mathbf{X}|\theta)\pi(\theta)}{p(\mathbf{X})}k(\theta^*|\theta)p(\mathbf{X}|\theta^*)h(\theta, \theta^*) \end{aligned}$$

Algorithm 5 ABC-MCMC

Input: $\theta_1 \sim \pi(\theta)$, ϵ, $\mathbf{x}_0$, N_{mcmc}, $\mathbf{X}$, $k(\cdot)$, $\rho(\cdot,\cdot)$ and $f(\cdot,\cdot)$.
Output: $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_{N_{mcmc}}\}$
t=1
Repeat
1. Move $\theta_t \to \theta^*$: $\theta^* \sim k(\theta_t)$
2. Synthesize $\mathbf{X}^* \sim f(\mathbf{x}_0, \theta^*)$
3. Calculate distance $d = \rho(\mathbf{X}^*, \mathbf{X})$
4. **if** $d \leq \epsilon$ **then**
5. generate indicator $u \sim \mathcal{U}(0, 1)$
6. **if** $u < \min\left(1, \frac{\pi(\theta^*)k(\theta_t|\theta^*)}{\pi(\theta_t)k(\theta^*|\theta_t)}\right)$ **then**
7. $\theta_{t+1} = \theta^*$
8. $t = t + 1$
9. **else**
10. $\theta_{t+1} = \theta_t$
11. $t = t + 1$
12. **end if**
13. **end if**

until $t = N_{mcmc}$

$$\begin{aligned} &= \frac{p(\mathbf{X}|\theta)\pi(\theta)}{p(\mathbf{X})}k(\theta^*|\theta)p(\mathbf{X}|\theta^*)\frac{\pi(\theta^*)k(\theta|\theta^*)}{\pi(\theta)k(\theta^*|\theta)} \\ &= \frac{p(\mathbf{X}|\theta^*)\pi(\theta^*)}{p(\mathbf{X})}p(\mathbf{X}|\theta)k(\theta|\theta^*) \\ &= p(\theta^*|\mathbf{X})p(\mathbf{X}|\theta)k(\theta|\theta^*) \\ &= p(\theta^*|\mathbf{X})p(\mathbf{X}|\theta)k(\theta|\theta^*)h(\theta^*, \theta) \\ &= p(\theta^*|\mathbf{X})r(\theta^* \to \theta), \end{aligned} \tag{36}$$

which holds the equality.

Example 3 *Deploying all aforementioned methods to the Lokta-Volterra model, starting from a relatively non-informative prior, we attempt to illustrate the advantage of the ABC-MCMC. In this case, the initial $\boldsymbol{\theta}_1$ is generated from a Gaussian distribution $\boldsymbol{\theta}_1 \sim \mathcal{N}(0.7, 0.01^2)$ and the tolerance ϵ is set to 20. For illustrating the outperformance of ABC-MCMC, ABC-rejection draws the samples from the identical prior. As shown in Figure 4, the posterior estimate from ABC-MCMC converges to the true value from a relatively far place after a few iterations. Moreover, it can be seen from the contour of the posterior shown in the right panel of Figure 4 that ABC-rejection performs similarly in terms of accuracy. The gain of Markov transition kernel $k(\cdot)$ is evident in the computational efficiency, where ABC-MCMC takes approximately 1300 iterations with an acceptance rate of 7.69% to collect 100 samples, whereas 24000 iterations are taken by ABC-rejection to collect 100 samples, yielding an acceptance rate of 0.42%. We also note that even though the tolerance is set to a value that is identical to* **Example 1** *($\epsilon = 20$), however, ABC-rejection performed worse in this study in comparison to the results shown in Figure 2. This is because the prior distribution is non-informative in this simulation, while it is set to a relatively favorable prior (easier to generate samples that is close to the true values) in the previous study.*

Interestingly, as shown in graphs, even though two param-

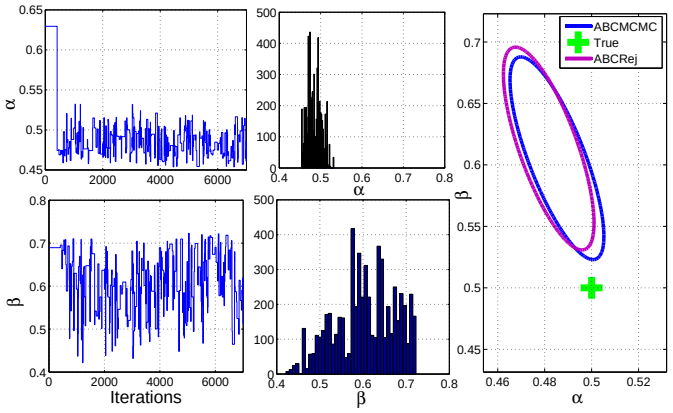


Fig. 4. *First column:* the trajectories of estimations for α and β from the ABC-MCMC algorithm. *Second column:* the histograms of the inferred samples, notice that the estimates obtained in the *burn-in* phase are not included in the histograms. *Last column:* The comparative study of performances between ABC-MCMC and ABC-rejection in term of the contour of particles.

eters are simultaneously estimated, it becomes immediately apparent that the variance of samples for α is considerably lower than the one for β. This phenomenon is known as *sloppiness/stiffness*, and such property gives a rise to the concept of *sensitivity analysis*, which motivates our innovative ABC-based inference algorithm. This algorithm will be introduced in another paper.

B. Advanced ABC methods

A variety of deficiencies, such as the simple sampling method, the curse of dimensionality and the slow Markov chain mixing, limit the widespread use of basic ABC methods on parameter estimation of biological systems [53]. The advanced ABC methods were developed by focusing on determining the acceptance criterion, either deterministically or automatically. In addition, the adaptive transition kernel is also merged in the SMC based ABC methods.

1) ABC-partial rejection control (ABC-PRC) algorithm: ABC-MCMC is able to partially alleviate the constraint on informativeness of prior, however, it suffers from the slow Markov chain mixing. A toy example was considered by [23], [55], [56], and we took this problem to illustrate this particular issue with ABC-MCMC, which is represented below.

Example 4 *Suppose it is of interest to estimate the mean μ of a mixture Gaussian model with a prior distribution $\pi(\cdot)$, given as*

$$f(x^*|\mu) = \frac{1}{2}\mathcal{N}(x^*;\mu,1) + \frac{1}{2}\mathcal{N}(x^*;\mu,0.01), \quad (37)$$

$$\pi(\mu) \sim \mathcal{U}(-10,10), \quad (38)$$

where $\mathcal{N}(x^*;\mu,\sigma^2)$ is the one-dimensional normal probability density function of mean μ and variance σ^2, evaluated at x^*, and $\mathcal{U}(a,b)$ is the uniform distribution on the interval $[a,b]$. The true observation x is assumed to be zero (i.e. $x=0$), therefore, the true posterior distribution is given as

$$p(\mu|x) \propto p(x|\mu)p(\mu) = \{\mathcal{N}(\mu;0,1) + \mathcal{N}(\mu;0,0.01)\}\mathcal{I}_{[-10,10]}(\mu), \quad (39)$$

where $\mathcal{I}_{[-10,10]}(\mu)$ is the indicator function, returning one if μ is in the interval $[-10,10]$, otherwise, zero. In this toy example, the $\mathbb{L}_1$ distance is used as the function to measure the discrepancy between x^* and x, since $x=0$, $\rho(x^*,x) = |x^*-x| = |x^*|$. In the ABC setting, the posterior distribution is approximated by the tolerance, given as

$$\begin{aligned} p_\epsilon(\mu|x) &\propto p(\rho(x,x^*)\le\epsilon|\mu)p(\mu) \\ &= p(|x^*|\le\epsilon|\mu)\mathcal{I}_{[-10,10]}(\mu) \\ &= p(-\epsilon\le x^*\le\epsilon|\mu)\mathcal{I}_{[-10,10]}(\mu), \end{aligned} \quad (40)$$

where $p(-\epsilon\le x^*\le\epsilon|\mu)$ indicates the density $p(x|\mu)$ that is between $p(-\epsilon|\mu)$ and $p(\epsilon|\mu)$, and this region can be calculated as

$$\begin{aligned} p(-\epsilon\le x^*\le\epsilon|\mu) &= \Phi(\epsilon-\mu) - \Phi(-(\epsilon+\mu)) \\ &\quad +\Phi(10(\epsilon-\mu)) - \Phi(-10(\epsilon+\mu)) \end{aligned} \quad (41)$$

where $\Phi(\cdot)$ is the cumulative distribution function of the normal Gaussian distribution. , the posterior distribution approximated For moving the samples, the MH transition kernel following the random is given as

$$k(\mu^*|\mu_n) \sim \mathcal{N}(\mu_n, 0.15^2). \quad (42)$$

Results shown in Figure 5 were obtained by setting tolerance as $\epsilon = 0.025$ and the number of MCMC iterations as $N_{mcmc} = 20000$. It can be easily observed from the graphs

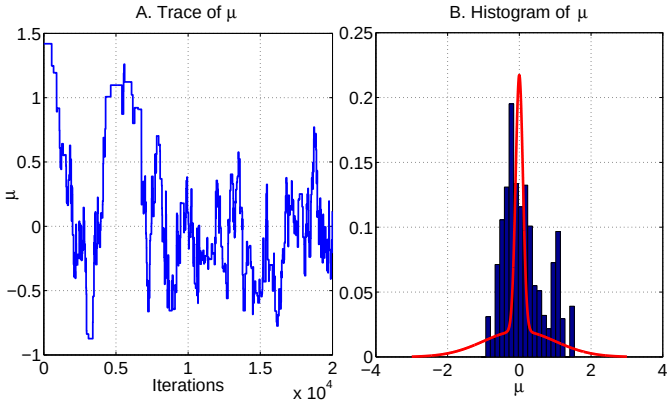


Fig. 5. A: Trajectory of samples for μ obtained from 20000 ABC-MCMC iterations. B: Histogram of the samples, and the true target distribution $p_\epsilon(\mu|x)$ is shown as the red solid line.

that, after converging to the true value, the samples again move to the place in space away from the true value leading to the multi-modal distribution. Moreover, no samples visit the tails of distributions within 20000 ABC-MCMC iterations. As this toy example is deliberately designed for illustrative purpose, the true distribution $p(\mu|x)$ can be approximated by $p_\epsilon(\mu|x)$ whose analytical solution is given as equation 41. The target distribution is shown as the red curve in Figure 5, which provides evidence to illustrate that ABC-MCMC suffers from the slow-mixing chain problem in this simple example.

To overcome this problem, SMC method has been recently merged into the ABC framework and details are introduced below. In brief, the philosophy of SMC is to gradually approach the posterior of interest via a series of intermediary

distributions, which are constituted by a collection of particles $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_{N_{smc}}\}$. In the intermediary phase, each particle will be perturbed around the space through the transition kernel and its importance is weighted based on how well it can represent the posterior. Those fittest realizations are 'encouraged' to characterize the target distribution by frequent selection, whilst those with negligible weights are discarded. Moreover, within the ABC framework, through bypassing the evaluation of the likelihood, the target distribution $p(\theta|\boldsymbol{X})$ is approximated as $p_\epsilon(\theta|\rho(\boldsymbol{X}^*, \boldsymbol{X}) \leq \epsilon)$. Intuitively, when the prior distribution is non-informative and the tolerance ϵ is small, the computational complexity immediately becomes an issue.

In the earliest form of SMC sampling ABC methods, [23] derived an innovative ABC approach by combine the previously proposed SMC sampler [57] with a partial rejection control scheme (ABC-PRC) at which the acceptance criterion is specified as a sequence of tolerances $\boldsymbol{\epsilon} = \{\epsilon_1, \ldots, \epsilon_T\}$. A smooth approach to the target posterior can be expected with this tolerance path, rather than a jump caused with a specific value of ϵ. Basically, ABC-PRC draws the particles from the previous population by considering their weights, and perturbs those particles around the space using the transition kernel, $\theta^{**} \sim k(\theta^*)$. The pseudo-observations are synthesized from the underlying model, $\boldsymbol{X}^* \sim f(\boldsymbol{x}_0, \theta^{**})$, where $\boldsymbol{x}_0$ is the initial condition and $f(\cdot,\cdot)$ is the dynamics. Particle θ^{**} is accepted and weighted if the discrepancy between synthetic data $\boldsymbol{X}^*$ and true dataset $\boldsymbol{X}$ is lower than the current tolerance ϵ_t. When the diversity issue appears, i.e. $\{\sum_{i=1}^{N_{smc}} (w_t^i)^2\}^{-1} \leq \frac{N_{smc}}{2}$, particles are resampled associated with their corresponding weights. The procedure of ABC-PRC is diagrammatically illustrated in Figure 6.

A variety of SMC-based ABC methods were developed with different weighting processes for pursuing unbiasedness and adaptivity. The earliest sequential sampling ABC method, ABC-PRC, however, can be shown to have inappropriate weight evaluation leading to biased estimation. The proof is given below. Considering the SMC sampler [57], the weight evaluation is given as

$$w_t^i \propto \frac{\pi(\theta_t^i) L_{t-1}(\theta^*|\theta_t^i)}{\pi(\theta^*) k_t(\theta_t^i|\theta^*)}, \tag{43}$$

where $\pi(\cdot)$ is the prior distribution for generating the initial particles and $L_{t-1}(\cdot)$ is known as the backward Markov transition kernel defining the probability for moving particles from θ_t^i to θ^*. Following the suggestion from [23], the Gaussian random walk kernel is adopted for $L_{t-1}(\theta^*|\theta_t^i)$ and $k_t(\theta_t^i|\theta^*)$.

[55] pointed out that the threshold ϵ_t for accepting particles violates the condition of using this weighting scheme from the previous SMC sampler [57][4].

[4]As the weight is defined as $w_t = \frac{p_t(\boldsymbol{\theta}_t)}{q_t(\boldsymbol{\theta}_t)}$, for approximating the target distribution at the time t, in the original SMC sampler, [57] achieves the approximation by marginalizing the particles coupled with an artificial constructed backward transition kernel L_{t-1}. That is $p_t(\theta_{1:t}) = p_t(\theta_t) \prod_{j=1}^{t-1} L_j(\theta_j|\theta_{j+1})$. However, when using ϵ_t, the distribution of current particles θ_t is approximated as $p_{\epsilon_t}(\theta_t) = \pi_{\epsilon_t}(\theta_t, \mathbf{X}^*|\mathbf{X}) \propto \pi(\theta_t) f(\mathbf{X}^*|\theta^*) \mathbb{I}(\rho \mathbf{X}^*, \mathbf{X})$, so it can not be described as the marginal distribution in θ_t.

Let us assume an extreme case in which ϵ is set to zero, then only if $\boldsymbol{X}^* = \boldsymbol{X}$ the particle θ_{t-1} would be taken as θ_t (for elaborating, here, θ^{**} is instead denoted as θ_{t-1}). To show the biasedness, we first define the joint density $p(\theta_t, \theta_{t-1})$ as

$$\begin{aligned} p(\theta_t, \theta_{t-1}) &= p(\theta_{t-1}) p(\theta_t|\theta_{t-1}) \\ &= \pi(\theta_{t-1}|\boldsymbol{X}) p(\theta_t|\theta_{t-1}, \boldsymbol{X}) \\ &= \pi(\theta_{t-1}|\boldsymbol{X}) \frac{k_t(\theta_t|\theta_{t-1}) f(\boldsymbol{X}^*|\theta_t) \mathbb{I}(\boldsymbol{X}^* = \boldsymbol{X})}{\int k_t(\theta_t|\theta_{t-1}) f(\boldsymbol{X}^*|\theta_t) \mathbb{I}(\boldsymbol{X}^* = \boldsymbol{X}) d\theta_t} \\ &\propto \pi(\theta_{t-1}|\boldsymbol{X}) k_t(\theta_t|\theta_{t-1}) f(\boldsymbol{X}|\theta_t). \end{aligned} \tag{44}$$

We further consider an arbitrary integrable function $r(\cdot)$, having variable θ_t with weights w_t, and expectation of $r(\cdot)$ is written as

$$\begin{aligned} &\mathbb{E}[r(\theta_t) w_t] \\ &\propto \iint r(\theta_t) p(\theta_t, \theta_{t-1}) w_t d\theta_t d\theta_{t-1} \\ &\propto \iint r(\theta_t) \pi(\theta_{t-1}|\boldsymbol{X}) k_t(\theta_t|\theta_{t-1}) f(\boldsymbol{X}|\theta_t) \frac{\pi(\theta_t) L_{t-1}(\theta_{t-1}|\theta_t)}{\pi(\theta_{t-1}) k_t(\theta_t|\theta_{t-1})} d\theta_t d\theta_{t-1} \\ &\propto \iint r(\theta_t) f(\boldsymbol{X}|\theta_{t-1}) f(\boldsymbol{X}|\theta_t) \pi(\theta_t) L_{t-1}(\theta_{t-1}|\theta_t) d\theta_t d\theta_{t-1} \\ &\propto \iint r(\theta_t) \pi(\theta_t|\boldsymbol{X}) f(\boldsymbol{X}|\theta_{t-1}) L_{t-1}(\theta_{t-1}|\theta_t) d\theta_t d\theta_{t-1} \\ &\propto \int r(\theta_t) \pi(\theta_t|\boldsymbol{X}) \times \left\{ \int f(\boldsymbol{X}|\theta_{t-1}) L_{t-1}(\theta_{t-1}|\theta_t) d\theta_{t-1} \right\} d\theta_t. \end{aligned} \tag{45}$$

Principally , the unbiased result should be denoted as

$$\mathbb{E}[r(\theta_t) w_t] \propto \int r(\theta_t) \pi(\theta_t|\boldsymbol{X}) d\theta_t, \tag{46}$$

where only the particles at t time instant θ_t are involved. If the backward transition kernel $L_{t-1}(\theta_{t-1}|\theta_t)$ is irrelevant to the current particles θ_t, i.e. the integral $\int f(\boldsymbol{X}|\theta_{t-1}) L_{t-1}(\theta_{t-1}|\theta_t) d\theta_{t-1}$ is always a constant, then the expectation of function $r(\theta_t)$ could be unbiased as shown in equation 45. However, the backward transition kernel adopts the random walk scheme, that is $L_{t-1}(\theta_{t-1}|\theta_t) = k_t(\theta_{t-1}|\theta_t) \sim \mathcal{N}(\theta_t, \sigma_k^2)$, which violates the condition. Consequently, the weights from equation 43 certainly produce biased estimates.

Example 5 In order to illustrate the biased estimate made by ABC-PRC, we consider the previously studied Gaussian mixture model. By using the identical algorithmic settings such as $x = 0$, $\sigma_k^2 = 0.15^2$ and $\rho(x^*, y) = |x^*|$, ABC-PRC is carried out with 10 consecutive iterations, associated with the sequence of tolerances starting from $\epsilon_1 = 2$ down to $\epsilon_{10} = 0.01$. The number of particles N_{smc} is set to 1000. Results are given in Figure 7. The posterior distribution approximated by tolerance $p_{\epsilon_t}(\theta|\mathbf{X})$ can be solved analytically as given in equation 41, and simulations from different tolerance values ϵ_t are shown as the purple curves. We also draw the exact posterior distribution by considering $\epsilon_t = 0$ as the red dashed line. As seen in the graph, only in the first iteration particles can roughly represent the posterior distribution. However, in successive iterations a failure to cover the distributional tail is clearly observed. Consequently, we

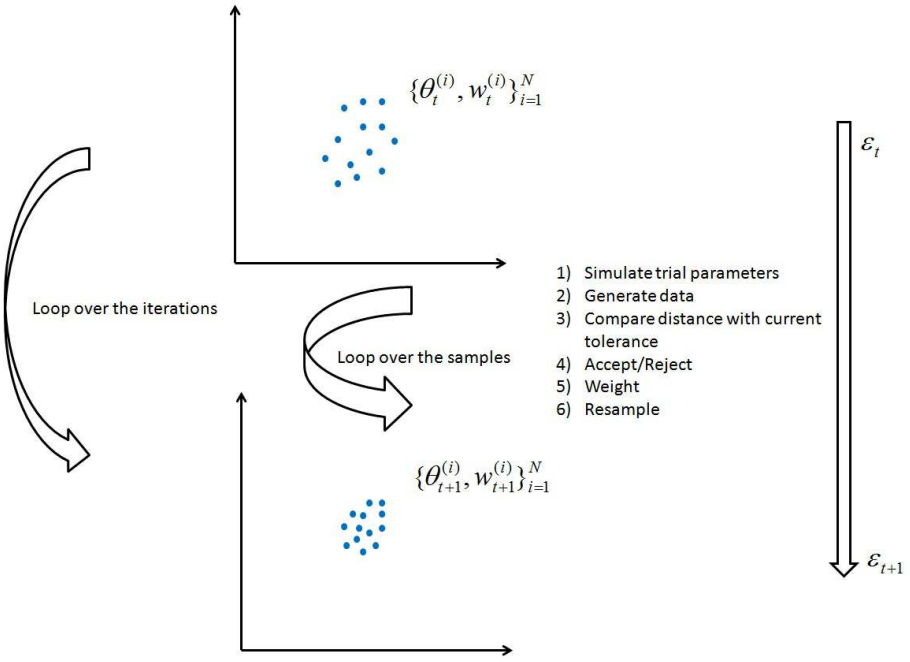


Fig. 6. In the beginning of each iteration, particles are picked from the previous population associated with weights, subsequently, those particles are perturbed by the transition kernel. The pseudo-observations are synthesized by solving the ODEs of system under study, if the discrepancy between synthetic dataset and true observation is lower than the current acceptance criterion, then the particles are retained. When the effective sample size N_{eff} is smaller than a threshold, i.e. $\{\sum_{i=1}^{N_{\text{smc}}} (w_t^i)^2\}^{-1} \leq \frac{N_{\text{smc}}}{2}$, a resampling step will be carried out on the particles and reset the weights to $1/N_{\text{smc}}$.

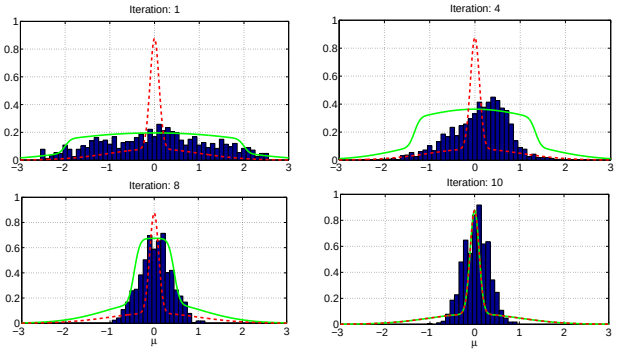


Fig. 7. Histograms of particles obtained from the ABC-PRC in 1^{st}, 4^{th}, 8^{th} and 10^{th} iterations for estimating mean μ of a Gaussian mixture model. The red dash line represents the true posterior distribution $p(\mu|x)$ whose explicit expression is given as equation 39, while the green solid line shows the posterior distribution approximated by the current tolerance $p_{\epsilon_t}(\mu|x)$ and of which the expression is given as equation 40.

discern that ABC-PRC yields biased estimations. One positive outcome of this study is that, unlike ABC-MCMC, ABC-PRC has the appealing property of being capable of avoiding the multi-modal estimation.

2) ABC-sequential importance sampling (ABC-SIS) algorithm: Since the weighting scheme adopted in the original ABC-PRC leads to biased estimates, two similar methods, population Monte Carlo *[58] based method, namely ABC-PMC, and* sequential Monte Carlo *based algorithm known as ABC-SIS were proposed by [55] and [59]. [60] issued a correction to the original ABC-PRC algorithm. Due to the high similarity among these algorithms, we only mention ABC-SIS as a representative paradigm of ABC-PMC and corrected ABC-PRC.*

Aiming to yield unbiased estimation of the target distribution, ABC-SIS abandons the constructed backward transition kernel for approximating the intermediary distributions. Alternatively, it carries out this approximation by straightforwardly using the prior distribution and the current tolerance ϵ_t, denoted as

$$p_t(\theta_t) = \frac{\pi(\theta_t)}{B} \sum_{j=1}^{B} \mathbb{I}(\rho(\boldsymbol{X}_j^*, \boldsymbol{X}) \leq \epsilon_t), \tag{47}$$

where $\pi(\theta_t)$ is the prior distribution for generating the initial particles for parameter; B is the number of sets of pseudo-observations synthesized; $\mathbb{I}(\cdot)$ is the indicator function; ϵ_t is the threshold for accepting particles to represent the intermediary distribution $p_t(\theta_t)$.

In the importance sampling *based methods, the weight is defined as*

$$w_t(\theta_t) = \frac{p_t(\theta_t)}{q_t(\theta_t)}, \tag{48}$$

where $q_t(\theta_t)$ is called the proposal distribution and the numerator is approximated by equation 47. The fundamental concept in importance sampling is to encourage the frequent selection of crucial samples, which is accomplished by choosing an appropriate proposal distribution. Following the derivation introduced by [54], the proposal distribution is formulated as

$$q_t(\theta_t) = \int q_{t-1}(\theta_{t-1}) k(\theta_t|\theta_{t-1}) d\theta_{t-1}, \tag{49}$$

here $q_t(\theta_t)$ is adopted as the marginal distribution of the previous proposal distribution. Since the previous proposal distribution $q_{t-1}(\theta_{t-1})$ is the approximation of the intermediary distribution at time $t-1$, i.e. $p_{t-1}(\theta_{t-1})$, $q_t(\theta_t)$ can be regarded as the distribution obtained from perturbing $p_{t-1}(\theta_{t-1})$. Replacing the $q_{t-1}(\theta_{t-1})$ as $p_{t-1}(\theta_{t-1})$ in equation 49, the proposal distribution is rewritten as

$$q_t(\theta_t) = \int p_{t-1}(\theta_{t-1}) k(\theta_t|\theta_{t-1}) d\theta_{t-1}. \tag{50}$$

Here the integral $\int p_{t-1}(\theta_{t-1})d\theta_{t-1}$ is always analytically intractable. To address this, by applying the Monte Carlo approximation [61], we have

$$\begin{aligned}\int p_{t-1}(\theta^i_{t-1})d\theta^i_{t-1} &\approx \frac{1}{N}\sum_{j=1}^{N} \delta_{\widehat{\theta^j}\sim p_{t-1}(\theta_{t-1})}(\theta^j_{t-1}) w(\theta^i) \\ &\approx \frac{1}{N}\sum_{j=1}^{N} w(\theta^j_{t-1}),\end{aligned} \tag{51}$$

N is the number of particles and $\{\theta^i_{t-1},\ i=1,\ldots,N\}$ are realizations from the distribution $p_{t-1}(\theta_{t-1})$. Since $\{\widehat{\theta^j},\ j=1,\ldots,N\}$ are taken from $p_{t-1}(\theta_{t-1})$, therefore, each $\widehat{\boldsymbol{\theta}^j}$ belongs to θ^i_{t-1}. Consequently, $\delta_{\widehat{\theta^j}\sim p_{t-1}(\theta_{t-1})}(\theta^i_{t-1})$ always equals one. By making use of this approximation, the proposal distribution at time t can therefore be given as

$$q_t(\theta^i_t) \approx \frac{1}{N}\sum_{j=1}^{N} w(\theta^j_{t-1}) k(\theta^i_t|\theta^j_{t-1}). \tag{52}$$

Defining $b_t(\theta_t) = \sum_{j=1}^{B} \mathbb{I}(\rho(\boldsymbol{X}^*_j, \boldsymbol{X}) \le \epsilon_t)$, and substituting equation 47 and 52 into the weight calculation, we have

$$\begin{aligned} w_t(\theta^i_t) &= \frac{p_t(\theta^i_t)}{q_t(\theta^i_t)} \\ &= \frac{\pi(\theta^i_t) b_t(\theta^i_t)}{\frac{B}{N}\sum_{j=1}^{N} w(\theta^j_{t-1}) k(\theta^i_t|\theta^j_{t-1})} \\ &\propto \frac{\pi(\theta^i_t)}{\sum_{j=1}^{N} w(\theta^j_{t-1}) k(\theta^i_t|\theta^j_{t-1})}. \end{aligned} \tag{53}$$

If the system of interest is formulated deterministically, the particle θ^i_t is used to generate the pseudo-observations only once and therefore B is set to 1. When the target system is stochastic, θ^i_t is utilized for synthesizing the pseudo-observations more than once, which means that $B > 1$.

Notice that these three algorithms only differ in the resampling process triggered by the effective sample size, but not in kind. In the beginning of each iteration $t \neq 0$, all algorithms pick particles from the previous population with their corresponding weights before moving these particles by the transition kernel. In ABC-PRC, beyond this picking strategy, if a severe degeneracy is observed (i.e. the effective sample size is lower than a threshold), particles will be resampled according to the current weights. However, in the other two methods, ABC-PMC and ABC-SIS, this additional resampling is canceled, since [55], [59] claimed that the resampling in response to degeneracy is unnecessary as it is already performed at the beginning of each iteration. The block for describing ABC-SIS steps is given in Algorithm 6.

Algorithm 6 ABC-SIS

Input: $\boldsymbol{\theta}_0 = \{\theta^1_0, \ldots, \theta^{N_{\text{smc}}}_0\} \sim \pi(\boldsymbol{\theta})$, $\mathbf{x}_0$, $\mathbf{X}$, $k(\cdot)$, $\rho(\cdot,\cdot)$, $f(\cdot,\cdot)$, T and $\boldsymbol{\epsilon} = \{\epsilon_1, \ldots, \epsilon_T\}$
Output: $\boldsymbol{\theta}_T = \{\theta^1_T, \ldots, \theta^{N_{\text{smc}}}_T\}$
for $t = 1, \ldots, T$ **do**
  for $i = 1, \ldots, N_{\text{smc}}$ **do**
    1. Draw $\boldsymbol{\theta}^*$ from $\boldsymbol{\theta}_{t-1}$ according to weights $\boldsymbol{w}_{t-1}$
    2. Move $\boldsymbol{\theta}^* \rightarrow \boldsymbol{\theta}^{**}$: $\boldsymbol{\theta}^{**} \sim k(\boldsymbol{\theta}^*)$.
    3. Synthesize $\mathbf{X}^* \sim f(\mathbf{x}_0, \theta^{**,i})$
    4. Calculate distance $d = \rho(\mathbf{X}^*, \mathbf{X})$
    5. **if** $d \le \epsilon_t$ **then**
    6. $\quad \theta^i_t = \theta^{**,i}$
    7. $\quad$ compute weight $w^i_t = \frac{\pi(\theta^i_t)}{\sum_{j=1}^{N} w^j_{t-1} k(\theta^i_t|\theta^j_{t-1})}$
    8. **end if**
  end for
end for

Example 6 To illustrate how ABC-SIS overcomes biased estimation which is encountered by the ABC-PRC, we examine the performance of the ABC-SIS on the same problem, as well as the algorithmic settings. Outputs from ABC-SIS are plotted in Figure 8. As shown in the graphs, ABC-SIS fully covers the target distributions in all iterations, even the distributional tails where ABC-PRC fails to explore. Consequently, the modification of weight calculation successfully removes the bias of the ABC sampling approach.

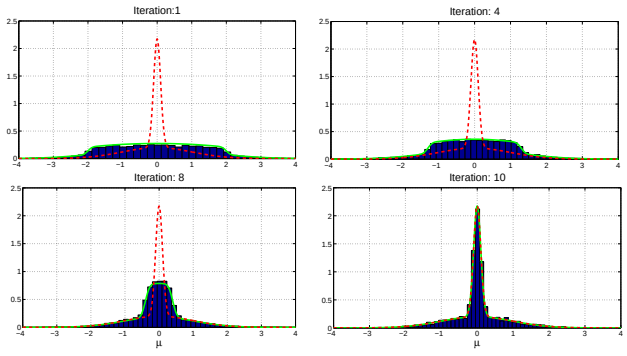


Fig. 8. Histograms of particles obtained from the ABC-SIS in 1^{st}, 4^{th}, 8^{th} and 10^{th} iterations for estimating mean μ of a Gaussian mixture model. The red dash line represents the true posterior distribution, that is $p(\mu|x)$, while the green solid line shows the approximated posterior distribution, i.e. $p_{\epsilon_t}(\mu|\rho(x^*, x) \le \epsilon_t)$.

3) ABC-sequential Monte Carlo sampler (ABC-SMC) algorithm: Apparently, the transition kernel $k(\cdot)$ and the tolerance ϵ play significantly crucial roles in determining the performance of all ABC methods. For the SMC-based ABC methods, the schedule of tolerances $\boldsymbol{\epsilon} = \{\epsilon_1, \ldots, \epsilon_T\}$ needs to be carefully designed, as large decreases lead to low acceptance rate, and conversely, small step sizes in the schedule require more iterations. Moreover, the computational complexity of weight calculation is quadratic in the number of particles, which may become an issue with large number of particles.

To address these problems, [56] proposed the innovative SMC-based ABC method which reduces the computational complexity to linear in the number of particles. The idea of the 'linear' weight calculation also provides an automatic way of finding the path of tolerance.

This adaptive ABC-SMC algorithm is theoretically underpinned by the previous SMC sampler [57], in which weights are evaluated as

$$w_t^i \propto w_{t-1}^i \frac{p_t(\theta_t^i) L_{t-1}(\theta_{t-1}^i|\theta_t^i)}{p_{t-1}(\theta_{t-1}^i) k_t(\theta_t^i|\theta_{t-1}^i)}, \tag{54}$$

similarly, $L_{t-1}(\cdot)$ is the backward Markov kernel, and following the suggestion from [57], an optimal choice for this backward kernel is chosen as

$$L_t^{opt}(\theta_t|\theta_{t+1}) = \frac{p_t(\theta_t) k_{t+1}(\theta_{t+1}|\theta_t)}{\int p_t(u) k_{t+1}(u|\theta_t) du}. \tag{55}$$

Unfortunately, the analytical solution of the integral in equation 55 is intractable. As one of the possible solutions, an MCMC kernel of invariant distribution $p_{t+1}(\cdot)$ for the transition kernel $k_{t+1}(\cdot)$ is considered for approximating the optimal backward kernel $L_t^{opt}(\cdot)$. Consequently, the weight can be approximated as following

$$\begin{aligned} w_t^i &\propto w_{t-1}^i \frac{p_t(\theta_t^i) L_t(\theta_{t-1}^i|\theta_t^i)}{p_{t-1}(\theta_{t-1}^i) k_t(\theta_t^i|\theta_{t-1}^i)} \\ &\propto w_{t-1}^i \frac{p_t(\theta_t^i)}{p_{t-1}(\theta_{t-1}^i)} \times \frac{p_t(\theta_{t-1}^i) k_t(\theta_t^i|\theta_{t-1}^{i)})}{p_t(\theta_n^i)} \\ &\times \frac{1}{k_t(\theta_t^i|\theta_{t-1}^i)} \\ &\propto w_{t-1}^i \frac{p_t(\theta_{t-1}^i)}{p_{t-1}(\theta_{t-1}^i)} \\ &\propto w_{t-1}^i \frac{\sum_{m=1}^{M_{smc}} \mathcal{I}_{\epsilon_t}(\boldsymbol{X}_{m,t-1}^{*,i}, \boldsymbol{X})}{\sum_{m=1}^{M_{smc}} \mathcal{I}_{\epsilon_{t-1}}(\boldsymbol{X}_{m,t-1}^{*,i}, \boldsymbol{X})} \end{aligned} \tag{56}$$

where $\boldsymbol{X}^* \in \mathbb{R}^{M_{smc} \times N_{smc} \times D_s \times N_{OT}}$ are the pseudo-observations, and $\boldsymbol{X}_{m,t-1}^{*,i}$ can be interpreted as the m^{th} synthetic outputs generated by the i^{th} parameter particle at t time instant θ_{t-1}^i. $\mathcal{I}_{\epsilon_t}(\boldsymbol{X}_{m,t-1}^{*,i}, \boldsymbol{X})$ is an indicator function that returns one if the discrepancy between pseudo-observation $\boldsymbol{X}_{m,t-1}^{*i}$ and data $\boldsymbol{X}$ is less than the tolerance ϵ_n, zero otherwise. Symbol M_{smc} here is the integer factor, functioning to operate M_{smc} SMC filters in parallel [62].

Benefiting from the weight calculation, ABC-SMC is able to adaptively select the current tolerance level ϵ_t. The idea behind this automatic scheme is to determine an appropriate reduction of the tolerance level based on the proportion of particles surviving under the current tolerance. If a large amount of particles remain 'alive', it implies the acceptance criterion is relatively loose and it is safe to make a jump for the next tolerance level. In contrast, if the ratio of 'alive' particles is low, this means particles are less likely to describe the posterior, therefore, a tiny movement should be considered. Such process is mathematically described as $PA(\boldsymbol{X}_t, \epsilon_{t+1}) \leq \alpha PA(\boldsymbol{X}_t, \epsilon_t)$.

For intentionally moving particles toward the target distribution, ABC-SMC determines the diagonal elements of covariance matrix for the random walk as the variances of the previous population, which is denoted as

$$var(\boldsymbol{\theta}) = \mathbb{E}[\boldsymbol{\theta}_t^2] - (\mathbb{E}[\boldsymbol{\theta}_t])^2, \tag{57}$$

where the expectation expands as

$$\mathbb{E}[\boldsymbol{\theta}_t] = \sum_{i=1}^{N_{smc}} w_t^i \theta_t^i. \tag{58}$$

Substituting equation 58 into equation 57, the variance is formulated as

$$\begin{aligned} var(\boldsymbol{\theta}) &= \mathbb{E}[\boldsymbol{\theta}_t^2] - (\mathbb{E}[\boldsymbol{\theta}_t])^2 \\ &= \sum_{i=1}^{N_{smc}} (w_t^i \theta_t^i)^2 - (\sum_{i=1}^{N_{smc}} w_t^i \theta_t^i)^2 \end{aligned} \tag{59}$$

Consequently, the algorithm for ABC-SMC is shown in Algorithm 7 where the notations are clarified in Table II. A comparative study of ABC-SMC against other methods is introduced in section V-A, particularly, a competition between ABC-SMC and ABC-SIS is represented in section V-B.

V. Quantitative performance comparison

In this section, we examine the performances of ABC approaches on parameter estimation to the heat shock response model in increasing dimensionality of the unknown parameters. This biological system is used as an example in [13] for quantifying how an initialization influences the performance of the three sequential inference methods. In this study, this system is used for exploring how the tolerance schedule dominates the performance of the ABC algorithms.

A. Two unknown parameters case

We first consider the relatively simple case in which two parameters are assumed unknown and the other four parameters are assigned to values from the literature [63]. The algorithmic settings for generating the synthetic dataset are the same in all simulations. As seen from the previous study described in [13], the difficulty of estimating the parameters rapidly grows with respect to the prior distribution and the dimensionality of unknown parameters. To ensure the identifiability of inference, only stiff parameters are assumed unknown (analysis of sensitivity of the heat shock model is given in our publishing paper). Hence the inference task is focused on two unknown parameters space (k_d and α_d), while the remaining four parameters are fixed to their true values, i.e. $K_d = 3$ and $\alpha_d = 0.015$. In addition, each unknown parameter is assigned the flat non-informative prior. All simulations were carried out

TABLE I
DEFINITIONS OF INPUTS

Method	Input
ABC-SMC	1. Number of particles N_{smc}; Initial tolerance ϵ_0, target tolerance ϵ_{Ta}; integer factor M_{smc}; tolerance reduction factor α_{smc}. 2. $\theta_0^i \sim \pi(\theta),\ i = 1, \ldots, N_{\text{smc}}$, initial particles for parameters. 3. $\mathbf{X}_0 : D_s \times N_{\text{smc}} \times M_{\text{smc}}$, initial condition for system states. 4. True system observations $\mathbf{X}$: $D_s \times \text{N}_{\text{OT}}$. 5. Resampling threshold $N_T = N_{\text{smc}}/2$

TABLE II
LIST OF NOTATIONS

ABC-SMC			
Description	Symbol	Function	Dimension
number of particles used for SMC	N_{smc}	represent prior/posterior distribution of parameter	scalar
integer factor	M_{smc}	number of SMC filters are used in parallel	scalar
initial particles for parameters	$\boldsymbol{\theta}_0$	initially use for generating solution of dynamics	$D_p \times N_{\text{smc}}$
initial condition of states	$\mathbf{x}_0$	states of dynamics at 1^{st} time instant	$D_{\text{s}} \times 1$
weights vector	$\boldsymbol{w}$	represent the importances of particles for parameters	$1 \times N_{\text{smc}}$
tolerance reduction factor	α_{smc}	calculate the next tolerance level	scalar
synthetic system output	$\mathbf{X}^*$	synthetic system output obtained by solving ODEs of dynamics associated with the current particles of parameters	$M_{\text{smc}} \times N_{\text{smc}} \times D_{\text{s}} \times N_{\text{OT}}$
distance matrix	$\mathbf{D}$	represent discrepancy between each synthetic and real datasets	$M_{\text{smc}} \times N_{\text{smc}}$
indicator matrix	$\boldsymbol{I}$	show if the underlying discrepancy is less than epsilon	$M_{\text{smc}} \times N_{\text{smc}}$
summation vector	$\boldsymbol{v}$	summation of elements in indicator matrix as column-wise	$1 \times N_{\text{smc}}$

with MATLAB®*on an Intel®Xeon™W3520 @ 2.67GHz with 12 GB RAM computer.*

Implementation details are following given

- ABC-rejection *generates samples from the uniform distribution* $\theta_{k_d} \sim \mathcal{U}(0, 10)$ *and* $\theta_{\alpha_d} \sim \mathcal{U}(0, 1)$. *The samples with a population of 1,000 are accepted with tolerance* $\epsilon = 0.7$.
- ABC-regression *uses the same prior for generating the samples as ABC-rejection. Mean of dataset is taken as the summary statistics and tolerance* $\epsilon = 0.7$. *The population of samples for representing posterior is 1,000.*
- ABC-MCMC *initializes the inference from the place where* $\boldsymbol{\theta}_0 = [3.5, 0.02]$. *The random walk proposals are utilized for parameters, in particular,* $\mathcal{N}(\theta_{t-1}, 0.0001^2)$ *for* k_d *and* $\mathcal{N}(\theta_{t-1}, 0.000007^2)$ *for* α_d. *We employ 2,000 MCMC iterations associated with the tolerance* ϵ *as 10.*
- ABC-PRC *generates particles from the uniform distribution* $\theta_{k_d} \sim \mathcal{U}(-10, 10)$ *and* $\theta_{\alpha_d} \sim \mathcal{U}(-1, 1)$ *with a population of 1,000. The path of tolerances starts from 4 and goes down to 0.7 within 10 iterations, reducing consistently in each iteration. The transition kernels for moving particles use the simplest random walk kernel, specifically,* $\mathcal{N}(\theta_{t-1}, 2^2)$ *for* k_d *and* $\mathcal{N}(\theta_{t-1}, 0.01^2)$ *for* α_d. *The threshold for triggering resampling is set to 500.*
- ABC-PMC *and* ABC-SIS *use the algorithmic settings which are identical to ABC-PRC.*
- ABC-SMC *uses the same prior distributions for synthesizing particles, for which population is 1,000 and* M_{smc} *is 10. The target tolerance* ϵ_T *is 10 and the discount factor* α_{smc} *for determining the next tolerance is 0.99 and the threshold for performing resampling is 500.*

The priors for generating the initial samples in ABC-rejection, ABC-regression and ABC-MCMC are tuned to strike a balance between accuracy and efficiency. In other words, to deliver comparable results with an affordable computational cost, the optimal initial conditions are chosen for these non-sophisticated approaches. In addition, tolerance for ABC-MCMC is adapted to a relatively larger value, because otherwise the proposed samples can barely survived under the tolerance considered in other non-sophisticated methods.

Figure 4.9 shows the estimates from seven ABC methods. Apparently, the non-sophisticated methods (ABC-rejection and ABC-regression) produce the estimations which successfully recover the unknowns and are in good agreement with the results obtained from the advanced approaches (ABC-PRC, ABC-SIS, ABC-PMC and ABC-SMC), while ABC-MCMC fails to make accurate inferences. As expected, all SMC based algorithms outperform the non-sophisticated methods in terms of accuracy, and the unknown parameters are inferred by the advanced algorithms with a high similarity.

As previously claimed, the computational complexity heavily depends on the acceptance criterion, and so we further assess the number of model evaluations required to fulfill the target tolerance. Since the evaluation of the complex system is usually expensive, therefore, the comparison is carried out after accounting for the cost of synthesizing pseudo-observations. In order to make a fair comparison, each approach is run 10 times and the results are shown in Figure 10.

Algorithm 7 ABC-SMC

Input: Details are listed in Table I and set $t = 1$

Output: $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_{N_{\text{smc}}}\}$

Set ϵ_1 to an arbitrary enough large value

Repeat

1. If $\{\sum_{i=1}^{N_{\text{smc}}} (w_t^i)^2\}^{-1} \leq N_T$, then resample $\boldsymbol{\theta}_t$ according to their weights $\boldsymbol{w}_t$;
   details can be found in [30];
2. Move $\boldsymbol{\theta}_t \rightarrow \boldsymbol{\theta}_{t+1}$: $\boldsymbol{\theta}_{t+1} \sim k(\boldsymbol{\theta}_t)$, where $\boldsymbol{\sigma}_k^2$ is determined by equation 59.
3. Synthesize $\mathbf{X}_{t+1}^* \sim f(\mathbf{x}_0, \boldsymbol{\theta}_{t+1})$
4. Compute the ratio of $\boldsymbol{\theta}_{t+1}$ remaining 'alive' under current tolerance ϵ_t,
   such function is defined as $\text{PA}(\mathbf{X}_{t+1}^*, \epsilon_t)$, details are given in A-D:
   A. compute $M_{\text{smc}} \times N_{\text{smc}}$ distance matrix $\mathbf{D} = \rho(\mathbf{X}_{t+1}^*, \mathbf{X})$ using equation 22
   B. compute $M_{\text{smc}} \times N_{\text{smc}}$ indicator matrix $\boldsymbol{I} : I_{m,n} = \mathcal{I}(d_{m,n}, \epsilon_t)$ where
   $\mathcal{I}(\cdot) = 1$ if $d_{m,n} \leq \epsilon_t$, otherwise, $\mathcal{I}(\cdot) = 0$.
   C. constitute $1 \times N_{\text{smc}}$ summation vector $\boldsymbol{v}$: $\boldsymbol{v}_n = \sum_{m=1}^{M_{\text{smc}}} I_{m,n}$
   D. $\text{PA}(\mathbf{X}_{t+1}^*, \epsilon_t) = \frac{\sum_{n=1}^{N_{\text{smc}}} (\boldsymbol{v}^n \neq 0)}{N_{\text{smc}}}$.
5. Determine ϵ_{t+1} by solving $\text{PA}(\mathbf{X}_{t+1}^*, \epsilon_{t+1}) \leq \alpha_{\text{SMC}} \text{PA}(\mathbf{X}_{t+1}^*, \epsilon_t)$.
6. **if** $\epsilon_{t+1} \geq \epsilon_{\text{Ta}}$ **then**
   Calculate weights $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t \frac{\text{PA}(\mathbf{X}_t^*, \epsilon_{t+1})}{\text{PA}(\mathbf{X}_t^*, \epsilon_t)}$ and $t = t + 1$
7. **else**
   $\boldsymbol{\theta} = \boldsymbol{\theta}_t$ and set $\epsilon_t = \epsilon_{\text{Ta}}$.

until $\epsilon_t \leq \epsilon_{\text{Ta}}$.

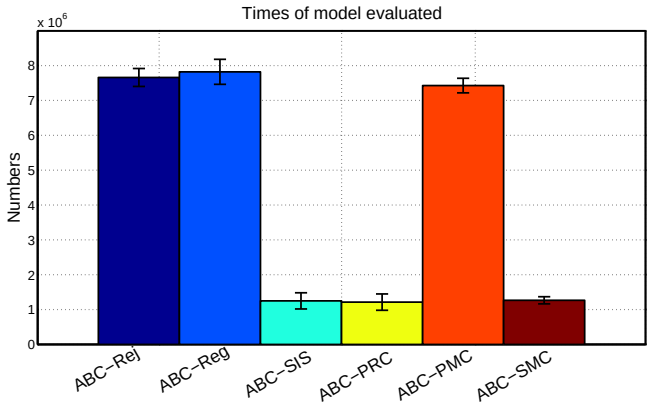


Fig. 10. Number of model evaluations required to achieve the target tolerance by six ABC methods. We have not included ABC-MCMC results because the estimations were poor quality.

As shown in the graph, more trials are attempted by the basic methods to find the fittest samples, while the majority of SMC based approaches consume less cost for simulating pseudo-observations. We note that ABC-PMC, however, is an exception. The significant increase of computational cost is caused by the transition kernel $k(\cdot)$ which utilizes twice the variance of the empirical population as the covariance matrix $\boldsymbol{\sigma}_k^2$. Such adaptivity negatively influences the rate of convergence when the prior (or posterior distributions in the early iterations) is widely distributed leading to the large $\boldsymbol{\sigma}_k^2$, subsequently, many proposals are rejected. ABC-SIS and ABC-PRC, benefiting from an appropriate $\boldsymbol{\sigma}_k^2$ of transition kernel, are clearly observed the outperformance in terms of computational efficiency. However, this cannot be expected in the case with high dimension, since finding an appropriate $\boldsymbol{\sigma}_k^2$ might be impossible. This claim is supported by the investigation stated in our publishing paper.

ABC-SMC seemingly accomplishes an attractive balance between adaptivity and efficiency in this example. Due to the 'no rejection' strategy of ABC-SMC, uninfluenced by the large transitions in the first few iterations, all particles are kept. In addition, most realizations are assigned non-zero weights by the importance evaluation coupled with the loose tolerances. As a result, the large amount of 'surviving' particles leads to a rapid reduction of tolerance in the early stage. Subsequently, due to the wide distribution of particles and decreased tolerance, only few particles are weighted as non-zero, which results in a serious drop of the effective sample size. Hence, a resampling step is triggered to enrich the diversity and it prevents the particles from collapsing. When a slow tolerance reduction is observed meaning that the particles almost achieved the convergence, the algorithm should be stopped by the fulfillment of tolerance. In other words, an appropriate target tolerance can greatly boost the computational efficiency, since less attempts will be made to satisfy the criterion with negligible movement. Although this determination is highly case dependent, a heuristic solution was suggested by [56] to stop the algorithm when the drop of tolerance is less than 1.5%.

B. Three unknown parameters case

In the previous example we found ABC-SIS and ABC-SMC perform similarly in both accuracy and computational efficiency, therefore, a comparison is further carried out to discriminate their abilities on parameter estimation by increasing the dimension of unknown parameters. The study introduced below is to identify parameters k_d, α_d and α_0 with assuming the remaining three are known. The method of generating the synthetic dataset is identical to the two unknown parameters case, and the prior and the transition kernel for α_0 are identical to those for α_d that are described in last section. Comparison is carried out by setting two tolerance paths for ABC-SIS, in which the initial tolerance ϵ_0 and the final tolerance ϵ_T are identical to ABC-SMC ($\epsilon_0 = 480$ and $\epsilon_T = 10$). Since ABC-SMC on average requires 240 iterations to reach ϵ_T in an automatic manner, for a fair comparison, ABC-SIS declines the tolerance from ϵ_0 down to ϵ_T in regular intervals (total number of intervals is 240). In addition, we further assess how ABC-SIS benefits from a manually-chosen tolerance sequence, via a tolerance schedule defined as $\boldsymbol{\epsilon} = [488, 440, 392, 344, 296, 249, 201, 153, 105, 57, 10]$ which is shown as the red line in Figure 11(c).

Posterior distributions of parameters are shown in Figure 11(a) and 11(b). As seen from the graphs, since ABC-SMC

adaptively utilizes the variance of the empirical distribution for the transition kernel, it greatly outstrips ABC-SIS in terms of the uncertainty of estimation. From the perspective of efficiency, the computational advantage of ABC-SMC cannot be expected in this higher dimensional example. The side effect from the adaptivity of ABC-SMC is clearly shown in Figure 11(d), in which substantially more model evaluations were taken by ABC-SMC. Nevertheless, benefiting from the 'no rejection' strategy, ABC-SMC can be ran in parallel which partially alleviates this computational complexity. By using 8 cores for the parallel computing, the time duration for simulating pseudo-observations 14,000,000 times was cut to one eighth of its original value. Even though the computational expense is still ten times greater than ABC-SIS with well designed tolerance schedule, efficiency can be further boosted by using more cores.

We note that the tolerance sequence used governs the computational performance of ABC-SIS, better performance is evident with a suitable tolerance path, whereas a negative influence is observed if the tolerances are inappropriately chosen.

In summary, from the comparisons carried out on the heat shock response system, it is clear that ABC-SMC has the superior performance in terms of accuracy, and the adaptivity further increases its appeal. The computational efficiency of ABC-SMC, however, substantially decreases in the problems with high dimensionality. Majority of SMC type ABC methods (except ABC-PMC) are more attractive in tackling simple problems, however, deterministically assigning tolerance sequences and the tuning of transition kernel limit their appeals for complex problems. The non-sophisticated methods are somewhat straightforward solutions in the inference problem, rather than the advanced methods.

From the study conducted in [13], we note that the particle filtering algorithm is capable of precisely estimating the parameters of heat shock model, regardless of the combination of the two unknowns. Additionally, the PF, benefiting from the one-pass data visiting scheme, can expect an improvement on the computational efficiency. The family of ABC algorithms, except ABC-MCMC, are reliable for accurately inferring two or three unknown parameters of heat shock system. The superior performance of ABC methods, however, is highly dependent on the algorithmic settings, e.g. the tolerance and the transition kernel. Moreover, similarly to the comparative study for discussing the advantage of sequential Methods in [13], ABC approaches might struggle with efficiency due to their batch nature, and the increase of computational complexity with respect to the growth of dataset volume is considerably greater than the PF. Since ABC methods bypass the evaluation of likelihood, therefore, these approaches alleviate the statistical assumption of additive noise and become suitable for the problems without sufficient prior knowledge.

VI. Discussion

In this chapter, we thoroughly investigated the approximate Bayesian computation methods both theoretically and empirically. Starting from the simplest ABC-rejection approach, we chronologically converge to the most powerful adaptive ABC-SMC algorithm. The features of all mentioned ABC methods are summarized in Table III.

Interestingly, in the study of ABC-regression method, the use of summary statistics heavily effects the performance of algorithm. When the mean of data is the only adopted statistics term, the method performs the best. Insights on the influence of summary statistics *represent an avenue for further study of ABC type methods and which may make it an appropriate treatment in the context of systems biology.*

References

[1] *M. Dewar, V. Kadirkamanathan, M. Opper, and G. Sanguinetti, "Parameter estimation and inference for stochastic reaction-diffusion systems: application to morphogenesis in D. melanogaster,"* BMC Systems Biology, *vol. 4, no. 1, p. 21, 2010.*

[2] *A. Golightly and D. J. Wilkinson, "Bayesian inference for stochastic kinetic models using a diffusion approximation,"* Biometrics, *vol. 61, pp. 781–788, 2005.*

[3] *M. Ashyraliyev, J. Jaeger, and J. G. Blom, "Parameter estimation and determinability analysis applied to Drosophila gap gene circuits,"* BMC Systems Biology, *vol. 2, 2008.*

[4] *C. Zhan and L. Yeung, "Parameter estimation in systems biology models using spline approximation,"* BMC Systems Biology, *vol. 5, no. 1, p. 14, 2011.*

[5] *M. Barenco, D. Tomescu, D. Brewer, R. Callard, J. Stark, and M. Hubank, "Ranked prediction of p53 targets using hidden variable dynamic modeling,"* Genome Biology, *vol. 7, no. 25, 2006.*

[6] *B. Jayawardhana, D. B. Kell, and M. Rattray, "Bayesian inference of the sites of perturbations in metabolic pathways via Markov chain Monte Carlo,"* Bioinformatics, *vol. 24, no. 9, pp. 1191–1197, 2008.*

[7] *V. Vyshemirsky and M. A. Girolami, "BioBayes: A software package for Bayesian inference in systems biology,"* Bioinformatics, *vol. 24, no. 17, pp. 1933–1934, 2008.*

[8] *G. Lillacci and M. Khammash, "Parameter estimation and model selection in computational biology."* PLoS Computational Biology, *vol. 6, no. 3, pp. 696–713, 2010.*

[9] *K. Nakamura, R. Yoshida, M. Nagasaki, S. Miyano, and T. Higuchi, "Parameter estimation of* in silico *biological pathways with particle filtering towards a petascale computing,"* Pacific Symposium on Biocomputing, *vol. 14, pp. 227–238, 2009.*

[10] *M. Quach, N. Brunel, and F. d'Alché Buc, "Estimating parameters and hidden variables in non-linear state-space models based on ODEs for biological networks inference."* Bioinformatics, *vol. 23, no. 23, pp. 3209–3216, 2007.*

[11] *X. Sun, L. Jin, and M. Xiong, "Extended Kalman filter for estimation of parameters in nonlinear state-space models of biochemical networks,"* PLoS ONE, *vol. 3, no. 11, p. e3758, 11 2008.*

[12] *J. Yang, V. Kadirkamanathan, and S. A. Billings, "In* vivo *intracellular metabolite dynamics estimation by sequential Monte Carlo filter," in* Computational Intelligence and Bioinformatics and Computational Biology, 2007. CIBCB '07. IEEE Symposium on, *april 2007, pp. 387 –394.*

[13] *X. Liu and M. Niranjan, "State and parameter estimation of the heat shock response system using Kalman and particle filters,"* Bioinformatics, *vol. 28, no. 11, pp. 1501–1507, 2012.*

[14] *S. Tavaré, D. J. Balding, R. C. Griffiths, and P. Donnelly, "Inferring coalescence times from dna sequence data."* Genetics, *vol. 145, pp. 505–518, 1997.*

[15] *Y. X. Fu and W. H. Li, "Estimating the age of the common ancestor of a sample of dna sequences."* Mol. Biol. Evol, *vol. 14, pp. 195–199, 1997.*

[16] *G. Weiss and A. V. Haeseler, "Inference of population history using a likelihood approach."* Genetic, *vol. 149, pp. 1539–1546, 1998.*

[17] *J. K. Pritchard, M. T. Seielstad, A. Perez-Lezaun, and M. W. Feldman, "Population growth of human y chromosomes: a study of y chromosome microsatellites."* Molecular Biology and Evolution, *vol. 16, pp. 1791–1798, 1999.*

[18] *J. D. Wall, "A comparison of estimators of the population recombination rate."* Mol. Biol. Evol, *vol. 17, pp. 156–163, 2000.*

TABLE III
ABC METHODS SUMMARY

Method	Time	Advantage	Disadvantage
ABC-Rejection	1997	Straightforward, simply compare the simulation with real dataset.	1. Sensitive to the choice of prior. 2. Performance completely depends on choice of tolerance ϵ.
ABC-Regression	2002	More accurate than ABC-Rejection.	1. Sensitive to the choice of prior. 2. Performance depends on choice of summary statistics $\boldsymbol{s}$.
ABC-MCMC	2003	Partially alleviates the dependence of prior distribution by using the transition kernel $k(\cdot)$ for perturbing samples.	The transition kernel needs to be tuned. Possibly trap in a region with rare opportunity to jump out if small σ_k^2 is used, while the acceptance rate will be low if the σ_k^2 is large.
ABC-PRC	2009	1. Candidate samples are drawn from the previous iteration with importance weights, efficiently eliminate the candidates which negligibly contribute to posterior. 2. The use of tolerance schedule makes algorithm gradually approach the target.	1. Tweak the covariance of transition kernel function $k(\cdot)$. 2. Superior performance can be only achieved by considering an appropriate tolerance path.
ABC-SIS	2009		
ABC-PMC	2009	The covariance matrix $\boldsymbol{\sigma}_k$ of the transition kernel $k(\cdot)$ is adaptively determined.	Computational complexity caused by simultaneously using the adaptive transition kernel and the fixed tolerance schedule.
ABC-SMC	2012	1. Adaptively select the tolerance level and transition kernel. 2. Weight computation depends on the ratio of survived particles.	Highest computation complexity among algorithms for the real biological systems.

[19] *A. Estoup, I. J. Wilson, C. Sullivan, J. M. Cornuet, and C. Moritz, "Inferring population history from microsatellite and enzyme data in serially introduced cane toads* bufo marinus." Genetics, *vol. 159, pp. 1671–1687, 2002.*

[20] *P. Marjoram, J. Molitor, V. Plagnol, and S. Tavaré, "Markov chain Monte Carlo without likelihoods."* Proc Natl Acad Sci, *vol. 100, no. 26, pp. 15 324–15 328, 2003.*

[21] *D. Wegmann, C. Leuenberger, and L. Excoffier, "Efficient approximate Bayesian computation coupled with Markov Chain Monte Carlo without likelihood,"* Genetics, *vol. 182, pp. 1207–1218, 2009.*

[22] *D. Wegmann and L. Excoffier, "Bayesian Inference of the Demographic History of Chippanzees."* Molecular Biology and Evolution, *vol. 27, no. 6, pp. 1425–1435, 2010.*

[23] *S. A. Sisson, Y. Fan, and M. M. Tanaka, "Sequential Monte Carlo without likelihoods."* Proc. Natl. Acad. Sci, *vol. 104, pp. 1760–1765, 2007.*

[24] *G. W. Peters, I. Nevat, S. A. Sisson, Y. Fan, and J. Yuan, "Bayesian Symbol Detection in Wireless Relay Networks via Likelihood Free Inference,"* IEEE Transactions on Signal Processing, *vol. 58, pp. 5206–5218, 2010.*

[25] *G. W. Peters, K. Balkrishnan, and B. Lasscock, "Model selection and Adaptive Markov Chain Monte Carlo for Bayesian Cointegrated VAR Models,"* Bayesian Analysis, *vol. 5, no. 3, pp. 465–491, 2010.*

[26] *G. W. Peters, S. A. Sisson, and Y. Fan, "Likelihood-free Bayesian inference for α-stable models,"* Computational Statistics and Data Analysis, *vol. 56, no. 11, pp. 3743–3759, 2010.*

[27] *R. Erhardt and R. Smith, "Approximate Bayesian Computing for Spatial Extremes,"* Computational Statistics and Data Analysis, *vol. 56, no. 6, pp. 1468–1481, 2012.*

[28] *T. Mckinley, A. Cook, and R. Deardon, "Inference in epidemic models without likelihoods,"* Int.J.Biostat, *vol. 5, no. 1, p. 24, 2009.*

[29] *A. Sitz, U. Schwarz, J. Kurths, and H. U. Voss, "Estimation of parameters and unobserved components for nonlinear systems from noisy time series."* Physical Review E, *vol. 66, no. 016210, 2002.*

[30] *G. Kitagawa, "A self-organizing state-space model,"* J. Am. Stat. Assoc, *vol. 93, no. 443, pp. 1203–1215, 1998.*

[31] *Y. Bar-Shalom, X. R. Li, and T. Kirubarajan,* Estimation, Tracking and Navigation: Theory, Algorithms and Software. *New York: John Wiley & Sons, 2001.*

[32] *J. S. Liu,* Monte Carlo strategies in scientific computing. *New York: Springer, 2001.*

[33] *D. Simon,* Optimal State Estimation. *Wiley, 2006.*

[34] *S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation."* In Proceedings of the IEEE, *vol. 92, no. 3, pp. 401–422, 2004.*

[35] *H. Chen, T. He and G. M. Church, "Modeling gene expression with differential equations,"* Pac. Symp. Biocomput., *vol. 4, pp. 29–40, 1999.*

[36] *I. Swameye, T. G. Müller, J. T. Timmer, O. Sandra, and U. Klingmüller, "Identification of nucleocytoplasmic cycling as a remote sensor in cellular signaling by databased modeling,"* Proceedings of the National Academy of Sciences, *vol. 100, no. 3, pp. 1028–1033, 2003.*

[37] *S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems."* Proceedings of the American Control Conference, *vol. D, no. 82, pp. 1628–1632, 1995.*

[38] *S. Julier, J. Uhlmann, and H. Durrant-White, "A new method for nonlinear transformation of means and covariances in filters and estimators,"* IEEE Transactions on Automatic Control, *vol. 45, pp. 477–482, 2000.*

[39] *A. Doucet, N. de Freitas, and N. Gordon,* Sequential Monte Carlo Methods in Practice. *Springer, 2001.*

[40] *M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking."* IEEE Transactions on Signal Processing, *vol. 50, no. 2, 2002.*

[41] *J. F. de Freitas, M. Niranjan, A. H. Gee, and A. Doucet, "Sequential Monte Carlo methods to train neural network models,"* Neural Computation, *vol. 12, no. 4, pp. 955–993, 2000.*

[42] *J. F. Kingman, "On the genealogy of large populations."* J. Appl. Prob, *vol. 19, pp. 27–43, 1982.*

[43] *M. Calder, S. Gilmore, and J. Hillston, "Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA," in* Transactions on Computational Systems Biology. *Springer, 2004, pp. 1–23.*

[44] *J. V. Rodríguez, J. A. Kaandorp, M. Dobrzyński, and J. G. Blom, "Spatial stochastic modelling of the phosphoenolpyruvate-dependent phosphotransferase (PTS) pathway in E*scherichia coli," Bioinformatics, *vol. 22, no. 15, pp. 1895–1901, 2006.*

[45] *H. E. Samad, M. Khammash, L. Petzold, and G. D., "Stochastic modelling of gene regulatory networks,"* J. Robust Nonlinear Control, *vol. 15, pp. 691–711, 2005.*

[46] *M. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters."* J. Amer. Statist. Assoc, *vol. 94, no. 446, pp. 590–599, 1999.*

[47] *A. J. Lotka,* Elements of physical biology. *Baltimore: MD: Williams & Wilkins Co, 1925.*

[48] *V. Volterra, "Variazioni e fluttuazioni del numero d'individui in soecie animali conviventi."* Men. R. Acad. Naz. dei Lincei, *vol. 2, pp. 31–113, 1926.*

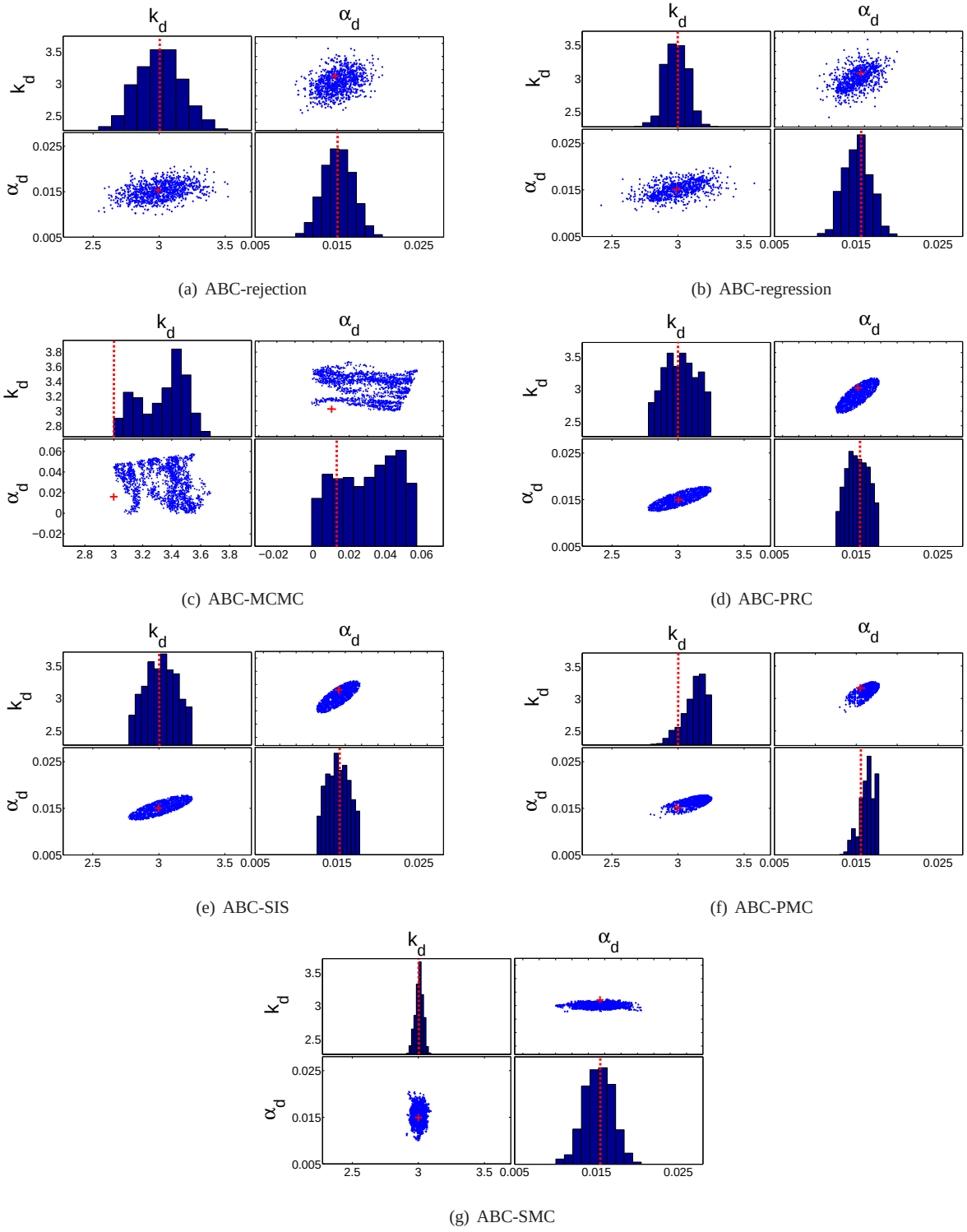
[49] *M. A. Beaumont, "Estimation of population growth or decline in genetically monitored populations."* Genetics, *vol. 164, pp. 1139–1160, 2003.*

[50] *J. Fan and W. Zhang, "Statistical estimation in varying coefficient models,"* Ann. Stat, *vol. 27, pp. 1491–1518, 1999.*
[51] *C. M. Bishop,* Pattern Recognition and Machine Learning. *New York: Springer-Verlag, 2006.*
[52] *P. Joyce and P. Marjoram, "Approximately sufficient statistics and Bayesian computation,"* Stat. Appl. Genet. Mol. Biol., *vol. 7, no. 1, p. 26, 2008.*
[53] *M. A. Beaumont, "Approximate Bayesian computation in evolution and ecology."* Annu. Rev. Ecol. Evol. Syst, *vol. 41, pp. 379–406, 2010.*
[54] *C. Andriue, N. De Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning,"* Machine Learning, *vol. 50, pp. 5–43, 2001.*
[55] *M. Beaumont, C. P. Robert, J. M. Marin, and J. M. Cornuet, "Adaptivity for abc algorithms: the abc-pmc scheme."* Biometrika, *vol. 94, no. 4, pp. 983–990, 2009.*
[56] *P. Del Moral, A. Doucet, and A. Jasra, "An adaptive sequential Monte Carlo method for approximate bayesian computation."* Stat. Comput, *vol. 22, no. 5, pp. 1009–1020, 2012.*
[57] ——, *"Sequential Monte Carlo samplers,"* J. R. Stat. Soc, *vol. 68, pp. 411–436, 2006.*
[58] *O. Cappé, A. Guillin, J.-M. Marin, and C. Robert, "Population Monte Carlo,"* J. Comput. Graph. Statist., *vol. 13, no. 4, pp. 907–929, 2004.*
[59] *T. Toni, D. Wlech, N. Strelkowa, A. Ipsen, and M. P. H. Stumpf, "Approximate Bayesian computation scheme for parameter inference and model seletion in dynamical systems."* J. R. Soc, *vol. 6, pp. 187–202, 2009.*
[60] *S. A. Sisson, Y. Fan, and M. M. Tanaka, "Correction: Sequential Monte Carlo without likelihoods."* Proc Natl Acad Sci, *vol. 54, p. e1760, 2009.*
[61] *A. Doucet, N. De Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods." in* Sequential Monte Carolo Methods in Practice, *2001, pp. 3–13.*
[62] *C. Andrieu and T. Johanes, "A tutorial on adaptive Markov chain Monte Carlo,"* Stat. Comput., *vol. 18, pp. 343–373, 2008.*
[63] *H. El-Samad, S. Prajna, A. Papachristodoulou, J. Doyle, and M. Khammash, "Advanced methods and algorithms for biological networks analysis."* Proceedings of the IEEE, *vol. 94, pp. 832–853, 2006.*

PLACE PHOTO HERE

Xin Liu Biography text here.

Mahesan Niranjan Biography text here.



(a) ABC-rejection

(b) ABC-regression

(c) ABC-MCMC

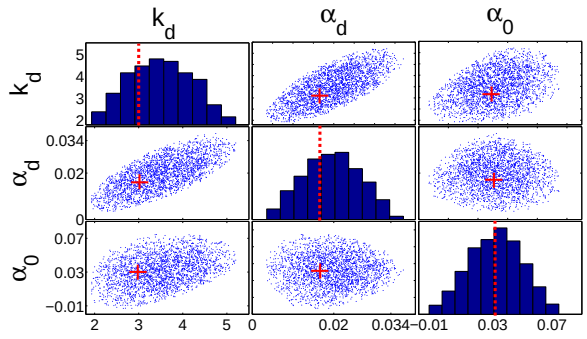
(d) ABC-PRC

(e) ABC-SIS

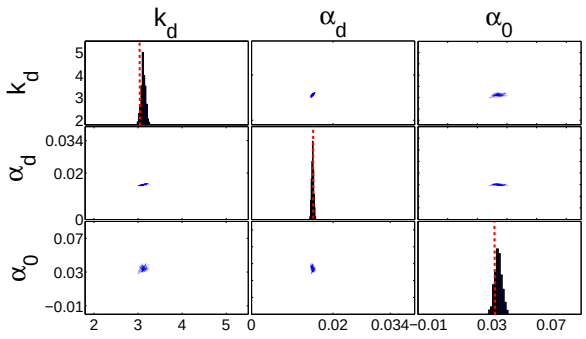
(f) ABC-PMC

(g) ABC-SMC

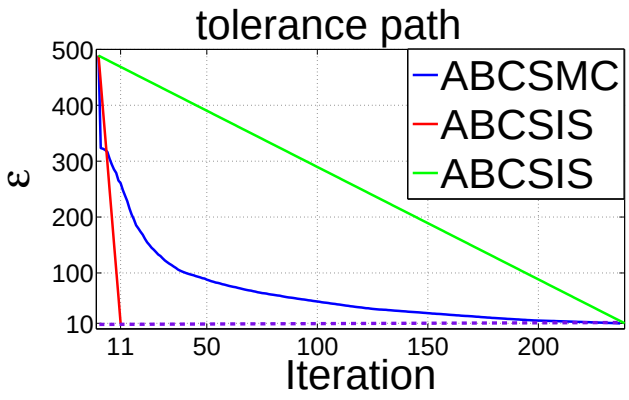
Fig. 9. Full posterior distributions of parameters k_d and α_d of the heat shock model obtained by seven methods, where the red dash lines and the red '+' both denote the true values of parameters. The scatter plots in each of the parameters are mirror images about the diagonal histograms.



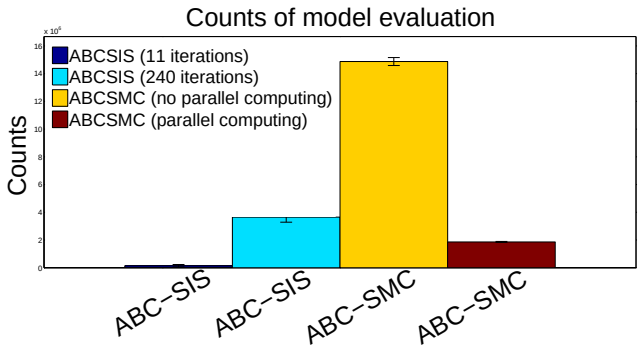
(a) ABC-SIS



(b) ABC-SMC



(c) path of tolerance



(d) counts of model evaluation

Fig. 11. (a) and (b): Posterior distributions of the parameters k_d, α_d and α_0 obtained by ABC-SIS and ABC-SMC, where the red dash lines are the true values and the red '+' is the location that particles should center. (c): Paths of tolerance, in which two schedules with different reductions are considered in ABC-SIS. (d): Counts of model evaluation that are carried out by ABC-SIS and ABC-SMC to reach the target tolerance.