

Comparing two sequential Monte Carlo samplers for exact and approximate Bayesian inference on biological models

Aidan C. Daly ^{1a}, Jonathan Cooper ^{2b}, David J. Gavaghan ^{3a}, and
Chris Holmes ^{4c}

^a*Department of Computer Science, University of Oxford*

^b*Research IT Services, University College London*

^c*Department of Statistics, University of Oxford*

August 21, 2017

¹Electronic address: aidan.daly@balliol.ox.ac.uk

²Electronic address: j.p.cooper@ucl.ac.uk

³Electronic address: david.gavaghan@dtc.ox.ac.uk

⁴Electronic address: cholmes@stats.ox.ac.uk

Abstract

Bayesian methods are advantageous for biological modeling studies due to their ability to quantify and characterize posterior variability in model parameters. When Bayesian methods cannot be applied, due either to non-determinism in the model or limitations on system observability, approximate Bayesian computation (ABC) methods can be used to similar effect, despite producing inflated estimates of the true posterior variance. Due to generally differing application domains, there are few studies comparing Bayesian and ABC methods, and thus there is little understanding of the properties and magnitude of this uncertainty inflation. To address this problem, we present two popular strategies for ABC sampling that we have adapted to perform exact Bayesian inference, and compare them on several model problems. We find that one sampler was impractical for exact inference due to its sensitivity to a key normalizing constant, and additionally highlight sensitivities of both samplers to various algorithmic parameters and model conditions. We conclude with a study of the O'Hara-Rudy cardiac action potential model to quantify the uncertainty amplification resulting from employing ABC using a set of clinically relevant biomarkers. We hope that this work serves to guide the implementation and comparative assessment of Bayesian and ABC sampling techniques in biological models.

1 Introduction

Parametric models have been hugely popular in the fields of systems and cellular biology since the inception of mathematical biological modeling [43]. Relative to phenomenological models, which simply seek to reproduce observations, parametric models boast advantages in interpretability, as the components and parameters of a well-designed model are meant to correspond directly to biological entities. Because of this correspondence, intuition into the target system can aid in formulation and parameterization of models, and conversely, the results obtained from the fitting of these models to biological data can give insights into the operation of the target system [39, 19, 25, 29]. The so-called “inverse problem” of determining model parameters from model observations, generally by automated means, can give insight into biological systems not only by determining the values of biologically relevant parameters that are most likely to reproduce the observed data, but by determining to what degree these parameters are unique [3].

Variation over model parameter estimates can come from one of several sources, and informs either our understanding of the biological system or our faith in the suitability of the model to explain the system. We expect measurement noise to contribute to model uncertainty in a well-defined manner proportional to its magnitude [4]. Greater uncertainty can be attributed either to inherent biological variability, such as that between individuals in a population, or, in cases of extreme variation, to “model unidentifiability” – the presence of potentially infinitely many equally valid parameterizations of a model given a set of observations [34, 6]. Such unidentifiabilities may arise from attempting to model portions of the system which are impossible to observe experimentally, and should lead either to a re-formulation of the model or a refinement of the data collected to fit it. Thus, while the form of a parametric model suggests a belief in a single set of “true” parameters, this is unrealistic when fitting to most biological data. Consequently, automated fitting methods such as maximum likelihood and least squares that output a single set of “optimal” values may lose a great deal of information from the data.

In the face of these concerns, Bayesian inference has become an increasingly attractive solution for parameter fitting in biological models [42]. Instead of single parameter values, Bayesian methods output a posterior *distribution* over the model parameters. This allows quantification of the spread over the parameter values as well as showing potential dependencies between model parameters. In addition to highlighting potentially unidentifiable model parameters, which would be characterized by extreme vari-

ance and/or correlation with another parameter [36], Bayesian methods can qualify and quantify inherent biological variation about parameters, such as the expected distribution of model parameters across a population of cells from which data were aggregated [42]. While other methods of uncertainty quantification exist, such as calculation of standard confidence intervals or regression-based sensitivity analyses, these methods can perform poorly in the face of large amounts of noise, large parameter spaces, and highly non-linear models [34]. Because Bayesian methods are more resistant to these conditions, they are a far more reliable and general approach for systems biology modeling [42].

One field that can benefit greatly from Bayesian modeling approaches is the field of cardiac action potential modeling. Over the last half century, these models have undergone an evolution in complexity commensurate to the increase in quality and quantity of cellular and sub-cellular data available [29, 35]. As a result, the days of Hodgkin and Huxley, where models were small enough to be fit by hand to data collected independently from a single source [16], are gone. The difficulty in gathering enough data to fit these larger models has driven many modelers to adapt parameters or even entire components from studies conducted under differing experimental conditions, or even from studies on cells from different species [27, 28]. While care is taken to adjust these models so that overall outputs match experimental data, these practices raise concerns as to model identifiability, or the degree to which the parameters for the model are unique [10]. Some modelers are addressing these concerns by the well-documented application of Bayesian inference techniques, which unambiguously link models to the data used to fit them, and provide a comprehensive characterization of posterior variability in the model parameters [17].

Studies involving cardiac models, however, may face restrictions on system observation that prevent the application of Bayesian techniques. Bayesian inference requires a calculable “likelihood function” – a function quantifying how likely it is that a given set of parameters would generate a given model output. When we cannot directly observe the quantity we are attempting to model, we cannot calculate such a likelihood. In cardiac and other biological systems, such observational constraints can be encountered when recording data *in vivo*. Such *in vivo* recordings are important for current efforts to create personalized heart models, which aim to advance clinical treatment by allowing for *in silico* patient-specific risk assessment for atrial fibrillation and other complications [26]. Although the output of a cardiac model is generally either the current passing through one or more ion channels or the voltage across the cell membrane over time, it is generally impossible to

record such time-dependent data *in vivo*, forcing experimentalists to instead record or report only lower dimensional biomarkers that attempt to characterize the action potential [9, 8]. Thus, data recorded or reported in these clinical studies may be limited to so-called “summary statistics” – data such as the time required for a cell to repolarize, the peak membrane potential, or the steepest change in membrane potential – rather than the full model output. Because these summary statistics are often highly non-linear functions of the membrane voltage, it is difficult to reason about the statistical properties of the error on these outputs even when such properties are well-understood for the underlying voltage trace. Without such knowledge, the specification of a likelihood function, and consequently, the application of Bayesian inference, becomes impossible. Even when a system is fully observable, if one chooses to employ a stochastic mathematical model of the underlying system, the randomness in the model can make the deterministic calculation of the likelihood of a set of parameters intractable, and Bayesian inference is similarly impossible. There are currently proposed stochastic models for ion channels [12], which would face this problem if employed.

In these situations where we cannot perform *exact* Bayesian inference, we can instead perform *approximate* Bayesian inference. Approximate Bayesian computation (ABC) methods, appropriately, approximate the true likelihood with another proximity metric – such as a custom distance function between vectors of summary statistics, or the squared distance between two realizations of a stochastic model observation – and seek to maximize this instead. The variance of the posteriors produced by these methods is generally inflated relative to the true value, as uncertainty resulting from measurement noise or the underlying system is compounded with uncertainty resulting from incomplete system observation, randomness in the model, or both. Despite this, the incredible generality of ABC methods has made them an area of current interest for the modeling of cardiac cells as well as other biological systems [11, 1, 40].

Due to their generally differing spheres of use, little theoretical or practical literature exists comparing Bayesian and approximate Bayesian inference on a given class of models (although an example on a toy model may be seen in [37]). This might prove frustrating to biological modelers. Cardiac modelers, for example, may wish to know how much uncertainty is introduced by the application of approximate Bayesian methods on clinical summary statistics relative to true Bayesian posteriors that would be obtained using full time-course data. The evaluation of the information content of summary statistics is currently an area of focus in ABC research. Because individually “sufficient” summary statistics – ones that represent the model

output without loss of information – are nearly nonexistent in biological studies, modelers generally employ ensembles of summary statistics with the intention of capturing as much information as possible [20]. Because the selection of optimal summary statistics is often data-dependent, there is not yet a universally agreed upon means of doing so in an approximate context [2, 32]. Cardiac action potentials, however, may be observed fully or partly depending on the system of study, and thus a comparative analysis of the approximate and exact posteriors could give *a priori* quantification of the information retained by a given set of summary statistics.

In order to enable such comparative analyses in cardiac models, we present two ABC sequential Monte Carlo (ABC-SMC) samplers that we have generalized to perform either Bayesian or approximate Bayesian inference with minimal adjustment. The method for generalizing these samplers was proposed originally by Richard Wilkinson [44] for ABC rejection and ABC Markov chain Monte Carlo (MCMC) samplers, but are extended in this paper to the more powerful class of sequential Monte Carlo (SMC) samplers, which use importance sampling to provide performance gains [5, 40, 37, 38]. We compare the Bayesian and approximate Bayesian inference capabilities of two sampling approaches, based on ABC-SMC algorithms proposed by Toni et al. [41] and Del Moral et al. [14], on a range of model problems, highlighting the relative merits of each sampling scheme and their sensitivities to key algorithmic parameters. We culminate with a novel *a priori* assessment of the information content within a set of commonly reported summary statistics of the cardiac action potential, which we determine by comparing the posteriors produced over the O’Hara Rudy action potential model [30] by the approximate Del Moral sampler to those produced by the exact Del Moral sampler using simulated time course data. We limit ourselves to single action potential recordings, which have been shown to contain a great deal of information about the parameters we will be considering [17], though the analyses presented could be easily extended to include multiple repeats. We hope that this will serve as an example of both exact and approximate SMC implementation and application, as well as a novel study on summary statistic evaluation in the field of cardiac modeling, the methodology of which may be extended to other areas of time series analysis.

In Section 2 we present and compare the two SMC samplers, provide pseudocode implementations for each, and discuss the interpretation of key algorithmic parameters. In Section 3 we outline the three models we chose for study along with our simulation conditions and common algorithmic settings for each posterior inference experiment. In Section 4 we present the results of posterior inference on each of the model problems, and in Section

5, we provide a full computational specification and a brief discussion of computational complexity. In Section 6 we discuss the implications of the results of our posterior inferences for choosing between the two samplers for a given system, and for the choice of summary statistics in a cardiac action potential model. Finally, in Section 7 we conclude and discuss potential future directions for this work.

2 Two ABC-SMC samplers

2.1 The acceptance kernel

In the absence of a calculable likelihood function, most ABC methods employ the squared error distance metric:

$$SE(\theta) = \sum_{i=1}^n (f(\mathbf{x}_i, \theta) - \mathbf{y}_i)^2 \quad (1)$$

to define the proximity of simulated data – in this instance, simulated according to model $f(\cdot, \cdot)$ at n points \mathbf{x} under parameters θ – to (noisy) experimental data \mathbf{y} . While minimization of the squared error function is appealing due to its equivalency to maximizing a Gaussian likelihood, which is commonly employed in biological models to represent observational noise, ABC methods make no assumptions about the distribution of this error. Instead, they typically uniformly accept or reject points in parameter space according to the binary function:

$$\rho_\varepsilon(\theta) = \mathbb{I}(SE(\theta) < \varepsilon), \quad (2)$$

where ε represents a “threshold” used to account for model error and/or observational noise. This deterministic kernel function is sometimes referred to as a “top-hat kernel” due to its shape.

Inference using such a top-hat kernel is exact only when model/system error is uniform over a sphere of radius ε . Because this is generally not the case for models of biological systems, employing a top-hat kernel may lead to an overestimation of posterior variance. Wilkinson suggests [44] that by replacing the indicator function in Eq 2 with a probability density function (PDF) representing the true distribution of error, ABC rejection and ABC Markov Chain Monte Carlo samplers can be adjusted to perform exact inference.

If our observation error is instead independently Gaussian with constant variance σ_e at each time point:

$$y_i \sim \mathcal{N}(f(x_i, \bar{\theta}), \sigma_e^2), \quad (3)$$

we should instead adopt the following smooth acceptance kernel to match that error distribution and allow for us to perform exact inference:

$$\rho_\varepsilon(\theta) = \exp\left(-\frac{SE(\theta)}{2(\varepsilon\sigma_e)^2}\right) / c_\varepsilon \quad (4)$$

where c_ε is a normalizing constant and ε is an adjustable “tolerance” controlling the width of the acceptance kernel. The simplest means of performing exact inference using such a kernel would be rejection sampling (Algorithm 1).

Algorithm 1 Probabilistic rejection sampler

- 1: Initialize an empty set $\Theta = \emptyset$ of accepted parameters
 - 2: **while** Θ contains fewer than N particles **do**
 - 3: Draw a sample θ^* from a prior distribution $\pi(\theta)$
 - 4: Add θ^* to Θ with probability $\rho_\varepsilon(\theta^*)$ (roll a die).
 - 5: **end while**
 - 6: **return** Θ
-

The efficiency of a rejection sampler employing the acceptance kernel in Eq 4 is highly dependent on the choice of c_ε . If the constant is set too high, the algorithm will require too many iterations, and if the constant is set too low, then the likelihood function will be “decapitated,” exceeding a value of 1 over some range of parameter space and resulting in uniform sampling over this range regardless of the true distribution (see Fig 1 for a graphical illustration).

ABC Sequential Monte Carlo samplers employ a sequence of decreasing thresholds $\varepsilon_0 > \varepsilon_1 > \dots > \varepsilon_T$ to sample from a sequence of intermediate distributions, smoothing the difference between the prior and posterior. Adopting the probabilistic acceptance kernel in Eq 4 will allow us to adjust these algorithms to perform exact inference as the iteration proceeds, as $\rho_\varepsilon(\theta) \rightarrow \mathcal{N}(f(\mathbf{x}, \theta), \sigma_e^2)$ when $\varepsilon \rightarrow 1$. In this case, the tolerance parameter ε can be analogized to the “temperature” parameter in simulated annealing. Thus, the adjusted algorithm similarly “heats” the likelihood to an initial value ε_0 , at which point it is highly smooth, then slowly “cools” the system towards the true posterior through a sequence of intermediate distributions

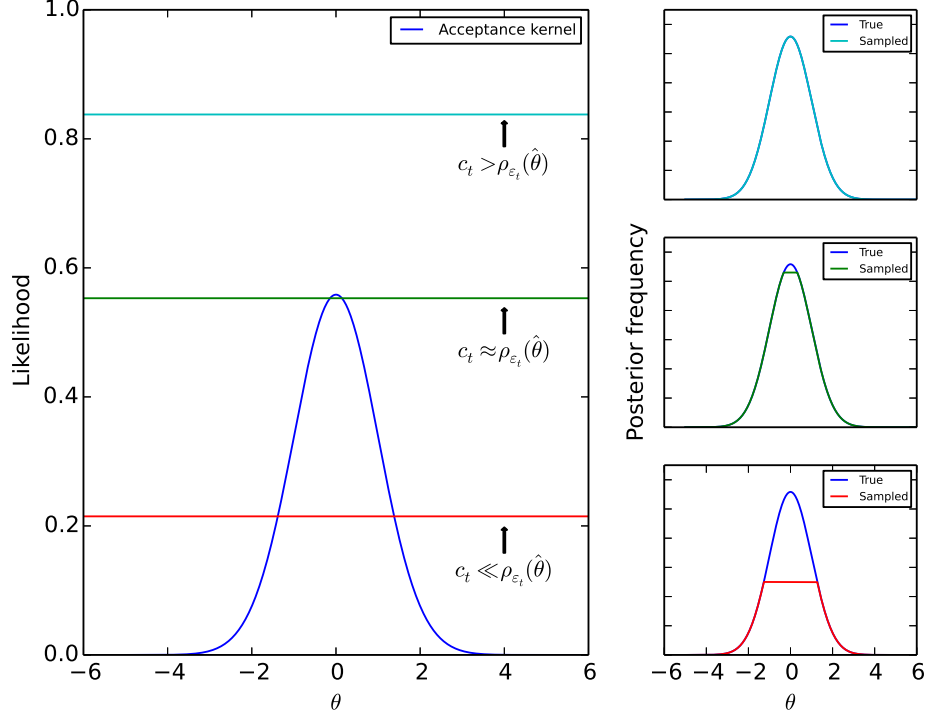


Figure 1: **Impact of the magnitude of the normalizing constant c_{ε_t} of Eq 4 on posteriors produced by a rejection sampler.** The left-hand plot demonstrates a sample (Gaussian) acceptance kernel function, with three potential values for the normalizing constant shown relative to an empirical estimate of the mode $\rho_{\varepsilon_t}(\hat{\theta})$. Right-hand plots show the true posterior (blue) along with empirical posteriors that would result from employing a sampler with an acceptance kernel normalized to a value greater than (top), slightly less than (middle), or much less than (bottom) its true modal value.

defined by ε_t . Such sequential strategies have shown benefits even for so-called “static” systems, where the likelihood function is known and MCMC methods would involve only a single posterior [5].

2.2 The generalized Toni algorithm

The first sampler we consider is based on the one proposed by Toni et al. [41]. This sampler maintains a constant number of particles N that, at each intermediate distribution, are updated by rejection sampling (along with local perturbation according to a fixed proposal distribution) from the previous posterior estimate, but with a lower error threshold. This is represented schematically in Fig 2.

We propose a generalized version of the Toni sampler in Algorithm 2 that employs an arbitrary acceptance kernel $\rho_\varepsilon(\theta)$. Let $\pi(\theta)$ represent the prior distribution on parameters θ . We will draw a series of intermediate distributions composed of n particles $\{\theta_t^{(i)}\}$ estimating the posterior, where the subscript t denotes the index of the intermediate distribution and the superscript i indicates the index of the particle *within said estimate*. We define a (series of) proposal distribution(s) $K_t(\theta'|\theta)$ which describe the probability of a particle moving from θ to θ' in parameter space between iterations. Finally, we define weights for these particles, $\{w_t^{(i)}\}$, that are updated after each iteration according to the following scheme:

$$w_t^{(i)} \propto \frac{\pi(\theta_t^{(i)})}{\sum_{j=1}^N w_{t-1}^{(j)} K_t(\theta_t^{(i)}|\theta_{t-1}^{(j)})}. \quad (5)$$

These particle weights $\{w_t^{(i)}\}$ are calculated according to a scheme proposed by [13] that minimizes their variance with respect to those of the previous estimate $\{w_{t-1}^{(i)}\}$.

The α parameter in Algorithm 2 can be thought of as the parameter defining the “cooling schedule” in simulated annealing, as it controls the percent reduction of ε_t at each iteration. If the posterior estimate update does not succeed within a given number of iterations ($I_{max} = 5 \times 10^6$), we re-attempt with a more conservative reduction of ε_t for the next iteration. This allows adaptive adjustment of the threshold and has given us good results with the top-hat acceptance kernel, where the value of the squared error to be minimized is related to the magnitude of the model output and would otherwise require much problem-specific adjustment.

A major advantage of this scheme is its amenability to parallelization. Because rejection sampling (lines 7-14) is performed independently for each

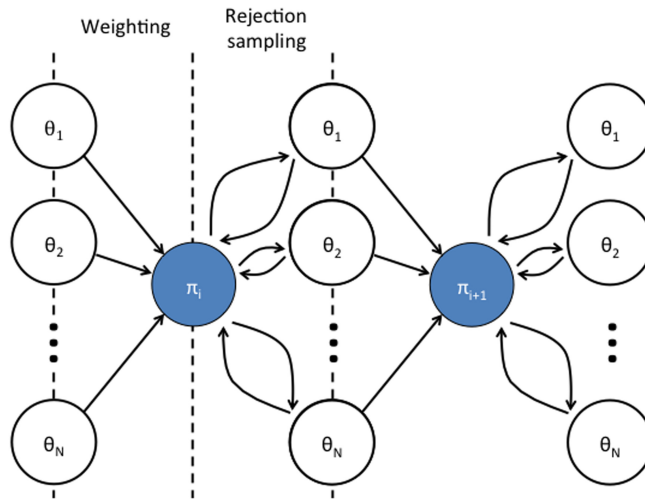


Figure 2: **Schematic representation of the Toni SMC sampler.** “Weighting” encompasses lines 20-22 of algorithm 2, where importance weights are assigned to each particle to allow for sampling the next estimate. “Rejection sampling” encompasses lines 7-14, where samples are drawn from the previous estimate, perturbed according to the proposal distribution, then accepted or rejected (and subsequently re-sampled) according to the kernel function.

Algorithm 2 The generalized Toni SMC sampler

Parameters: n : number of particles, k_{min}, I_{max} : termination conditions, α : cooling schedule, ε_0 : initial error threshold

```
1: Set  $t = 0$ ,  $k_0 = \varepsilon_0$  { $t$  is the iteration number,  $k_t$  controls reduction in
   error threshold for this iteration.}
2: while  $k_t > k_{min}$  do
3:   if  $t = 0$  then
4:     Using Algorithm 1 with tolerance  $\varepsilon_0$ , draw  $n$  particles  $\{\theta_0^{(i)}\}$ 
5:     Set  $w_0^{(i)} = 1/n$  for  $i = 1$  to  $n$ 
6:   else
7:     for  $i = 1$  to  $n$  do
8:       Draw  $\theta^*$  from  $\{\theta_{t-1}^{(i)}\}$  (with weights  $\{w_{t-1}^{(i)}\}$ )
9:       Draw  $\theta^{**}$  from the proposal distribution  $K_t(\theta|\theta^*)$ 
10:      if  $\pi(\theta^{**}) = 0$  then
11:        Repeat from line 8
12:      end if
13:      Set  $\theta_t^i = \theta^{**}$  with probability  $\rho_{\varepsilon_t}(\theta^{**})$  (roll a die). Otherwise,
        repeat from line 8.
14:    end for
15:    if Lines 7-14 did not succeed in populating  $\{\theta_t^{(i)}\}$  within  $I_{max}$  draws
        from the previous posterior estimate then
16:       $k_t \leftarrow \alpha k_t$ 
17:       $\varepsilon_t \leftarrow \varepsilon_{t-1} - k_t$ 
18:      Repeat from line 7
19:    else
20:      for  $i = 1$  to  $n$  do
21:        Set  $w_t^{(i)} = \pi(\theta_t^{(i)}) / \sum_{j=1}^n (w_{t-1}^{(j)} K_t(\theta_t^{(i)}|\theta_{t-1}^{(j)}))$ 
22:      end for
23:    end if
24:  end if
25:  Normalize  $\{w_t^{(i)}\}$ 
26:   $k_{t+1} = \min(k_t, \alpha \varepsilon_t)$ 
27:   $\varepsilon_{t+1} = \varepsilon_t - k_{t+1}$ 
28:   $t \leftarrow t + 1$ 
29: end while
30: return  $(\{\theta_t^{(i)}\}, w_t^{(i)})$ 
```

particle, sampling can be divided between different threads/processes. The calculation of weights (lines 20-22) can also be parallelized once rejection sampling for the associated particles has completed. In our Python implementation, this parallelization was accomplished through the use of the Pathos multiprocessing module [21, 22].

2.3 The generalized Del Moral algorithm

A potential drawback of the Toni algorithm is the complexity of its weighting scheme. We note that this weighting scheme (Eq 5) has an asymptotic cost of $O(N)$ for each particle, leading to an $O(N^2)$ cost for updating weights during distribution update. While the cost of simulation is usually assumed to dominate the runtime of the sampler, this expensive weighting step may add noticeable overhead for large numbers of particles regardless of the model chosen.

This computational complexity in part motivated Del Moral et al. to propose an alternative ABC-SMC implementation based on Metropolis-Hastings updates, rather than rejection sampling, that allowed them to employ an approximation of the weighting scheme in Eq 5:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{\mathbb{I}(SE(\theta_{t-1}^{(i)}) < \varepsilon_t)}{\mathbb{I}(SE(\theta_{t-1}^{(i)}) < \varepsilon_{t-1})} = w_{t-1}^{(i)} \frac{\rho_{\varepsilon_t}(\theta_{t-1}^{(i)})}{\rho_{\varepsilon_{t-1}}(\theta_{t-1}^{(i)})}. \quad (6)$$

This scheme boasts $O(1)$ calculation of particle weights, leading to only $O(N)$ cost for a single distribution update. While this approximation overestimates the variance of the incremental weights relative to the scheme in Eq 5, the authors believe the approximation should be sufficiently close as long as sequential posterior estimates are sufficiently close (i.e., $\pi_t(\theta) \approx \pi_{t-1}(\theta)$).

The Del Moral algorithm effectively maintains a set of n independent Markov chains that are periodically resampled when the effective sample size (ESS):

$$ESS(\{w_t^{(i)}\}) = 1 / \sum_{i=1}^N ((w_t^{(i)})^2) \quad (7)$$

falls below some threshold (generally taken to be $n/2$), indicating a significant drop in entropy. This implementation is represented schematically in Fig 3, and a generalized version employing an arbitrary acceptance kernel $\rho_\varepsilon(\theta)$ is described in Algorithm 3.

The adaptive ε_t schedule (controlled by a cooling parameter α identically to our implementation of the Toni algorithm in Algorithm 2) remains the

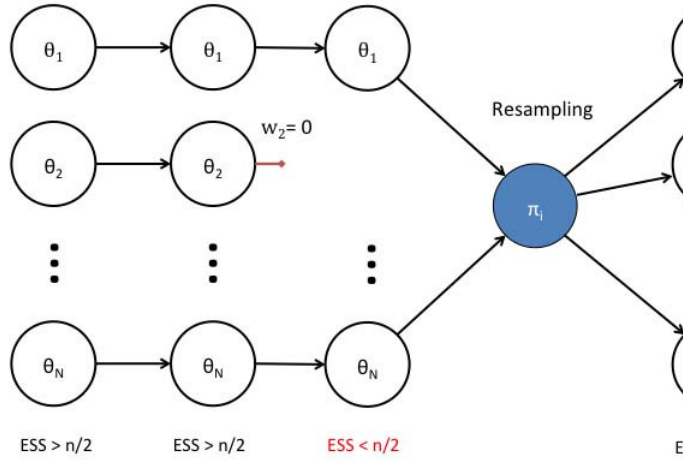


Figure 3: **Schematic representation of the Del Moral SMC sampler.** Between rounds, particles with non-zero weights evolve according to independent Markov Chains (lines 22-25, Algorithm 3) and weight calculation (lines 7-9) can happen concurrently. If effective sample size (ESS) falls below a critical fraction, “Resampling” occurs to repopulate the posterior estimate with uniformly weighted draws from the previous estimate (lines 16-21).

Algorithm 3 The generalized Del Moral SMC sampler

Parameters: n : number of particles, k_{min} : termination condition, α : cooling schedule, ε_0 : initial error threshold

- 1: Set $t = 0$, $k_0 = \varepsilon_0$ { t is the iteration number, k_t controls reduction in error threshold for this iteration.}
 - 2: **while** $k_t > k_{min}$ **do**
 - 3: **if** $t = 0$ **then**
 - 4: Using Algorithm 1 with tolerance ε_0 , draw n particles $\{\theta_0^{(i)}\}$
 - 5: Set $w_0^{(i)} = 1/n$ for $i = 1$ to n
 - 6: **else**
 - 7: **for** $i = 1$ **to** n **do**
 - 8: Set $w_t^{(i)} = w_{t-1}^{(i)} \rho_{\varepsilon_t}(\theta_t^{(i)}) / \rho_{\varepsilon_{t-1}}(\theta_t^{(i)})$.
 - 9: **end for**
 - 10: **if** all weights $\{w_t^{(i)}\}$ are 0 **then**
 - 11: $k_t \leftarrow \alpha k_t$
 - 12: $\varepsilon_t \leftarrow \varepsilon_{t-1} - k_t$
 - 13: Repeat from line 7
 - 14: **end if**
 - 15: Normalize $\{w_t^{(i)}\}$
 - 16: **if** $ESS(\{w_t^{(i)}\}) < n/2$ **then**
 - 17: Resample n particles $\{\theta^{(i)*}\}$ from $\{\theta_{t-1}^{(i)}\}$ using the new weights $\{w_t^{(i)}\}$.
 - 18: **for** $i = 1$ **to** n **do**
 - 19: Set $\theta_{t-1}^{(i)} = \theta^{(i)*}$ and $w_t^{(i)} = 1/n$.
 - 20: **end for**
 - 21: **end if**
 - 22: **for** $i = 1$ **to** n **do**
 - 23: Draw θ^* from the proposal distribution $K_t(\theta | \theta_{t-1}^{(i)})$
 - 24: Set $\theta_t^i = \theta^*$ with probability $\min(\frac{\rho_{\varepsilon_t}(\theta^*) K(\theta_{t-1}^{(i)} | \theta^*) \pi(\theta^*)}{\rho_{\varepsilon_t}(\theta_{t-1}^{(i)}) K(\theta^* | \theta_{t-1}^{(i)}) \pi(\theta_{t-1}^{(i)})}, 1)$ (roll a die). Otherwise, set $\theta_t^{(i)} = \theta_{t-1}^{(i)}$.
 - 25: **end for**
 - 26: **end if**
 - 27: $k_{t+1} = \min(k_t, \alpha \varepsilon_t)$
 - 28: $\varepsilon_{t+1} = \varepsilon_t - k_{t+1}$
 - 29: $t \leftarrow t + 1$
 - 30: **end while**
 - 31: **return** $(\{\theta_t^{(i)}, w_t^{(i)}\})$
-

same as in the Toni implementation, but as this sampling scheme performs a fixed number of Metropolis-Hastings steps for each intermediate distribution update, there is no need to set a maximum number of iterations I_{max} for the posterior update step. Instead, since the weighting scheme is effectively counting the number of particles that would “survive” the ε_t update, the conditional on line 10 determines whether this update would result in all particles being rejected by the kernel ρ_{ε_t} . One might replace this check with a more general condition, where the update fails if the *ESS* drops below some critical threshold, which we expect would increase resistance of the posterior estimates to underdispersion resulting from an overly aggressive choice of cooling schedule.

We have generally been using a Gaussian proposal distribution of fixed width, chosen to be 10% of the variance of the prior for each parameter. However, we have found that adaptively adjusting the width of the kernel (scaling the covariance matrix by a constant) based on the acceptance rate of draws, as is done for many MCMC samplers, has helped our Del Moral implementation. We use an update scheme adopted from the PyMC [31] implementation of tuned MCMC that increases/decreases proposal variance proportionally to deviation of the acceptance rate from the 20%-50% range at fixed intervals (every 100 proposals) – see Supplementary Algorithm A.1 for more details.

3 Models and Methods

3.1 The linear model

The following linear model was chosen as an initial test bed for the two algorithms:

$$f(x, a) = ax, \quad y_i \sim \mathcal{N}(f(x_i), \sigma_e^2). \quad (8)$$

This model has a known Gaussian posterior over the slope parameter a when a conjugate Gaussian prior $\pi(a)$ is assumed over the slope parameter:

$$\pi(a) \sim \mathcal{N}(0, \sigma_a^2) \rightarrow P(a|\mathbf{y}) \sim \mathcal{N}\left(\frac{\mathbf{x}^\top \mathbf{y}}{\mathbf{x}^\top \mathbf{x} + \sigma_e^2 \sigma_a^{-2}}, \frac{\sigma_e^2}{\mathbf{x}^\top \mathbf{x} + \sigma_e^2 \sigma_a^{-2}}\right). \quad (9)$$

For each fitting experiment, data were simulated as a single vector of n noisy observations \mathbf{y} taken at fixed time points \mathbf{x} . $n=301$ time points were chosen, symmetrically about 0: $\mathbf{x} = \{-150, -149, \dots, 0, \dots, 149, 150\}$. The true value of a was chosen as $\bar{a} = 5$, while the width of the prior σ_a was set to 10, and the magnitude of Gaussian noise σ_e was set to 1.

A Gaussian proposal distribution $K(\theta'|\theta) \sim \mathcal{N}(\theta, 1)$ was chosen for both samplers during both exact and approximate inference. The width of this distribution was periodically updated in the Del Moral scheme as described in Supplementary Algorithm A.1 in order to heuristically attain a 20-50% acceptance rate of Metropolis-Hastings steps.

When employing the top-hat kernel (Eq 2), the initial tolerance ε_0 for both samplers was set to $\max_i(SE(\theta_0^i))$, the maximum error among N samples drawn from the prior. Both algorithms were set to terminate when $k_t \leq k_{min} = 1$, or when $\varepsilon_t < 301$, the sample size of the data. This second criterion was introduced because the expected value of $SE(\hat{\theta})$ at the true solution, \bar{a} , is equal to the number of observations n , which thus sets a reasonable estimate for the final error ε_T and would ensure timely termination. In the presence of noisy observations, the acceptance probability of the top-hat kernel (Eq 2) approaches 0 for all θ as $\varepsilon_t \rightarrow 0$ when a finite number of observations are employed, and thus a positive limit is expected on ε_T .

When employing a Gaussian kernel (Eq 4), the initial tolerance ε_0 for both samplers was set to 1000, as it gave a reasonable acceptance rate for the first round of rejection sampling. Both algorithms were set to terminate when $k_t < k_{min} = 0.001$, or when $\varepsilon_t = 1$, as in the probabilistic kernel (Eq 4) the interpretation of ε is a multiplier of the true standard deviation of the error model, and thus decreasing below 1 would cause underdispersion in the final estimate.

3.2 The polynomial model

The following polynomial model was chosen for subsequent comparison to investigate the effect of high-dimensional parameter space on the samplers:

$$f(x) = \sum_{i=1}^P c_{i-1} x^{i-1}, \quad \mathbf{y}_t \sim \mathcal{N}(f(\mathbf{x}_t), \sigma_e^2), \quad t = \{0, \dots, T\}, \quad (10)$$

where \mathbf{x} ranges from -1 to 1 in 301 equally spaced increments, $\sigma_e^2 = 1$, and $c_0 = 6, c_1 = -6, c_2 = 5, c_3 = -5, c_4 = 4, c_5 = -4, c_6 = 3, c_7 = -3, c_8 = 2, c_9 = -2, c_{10} = 1, c_{11} = -1, c_{12} = 2, c_{13} = -2, c_{14} = 3, c_{15} = -3$. Output from this model is depicted graphically in Supplementary Fig. B.1.

All parameters were given uniform priors over the range (-10,10) to remove any effects of the prior PDF from particle weighting in the SMC algorithms. An independent Gaussian proposal distribution $K(\theta'|\theta) \sim \mathcal{N}(\theta, 2)$ was set over each model parameter for both samplers. The width of this distribution was periodically updated in the Del Moral scheme as described in Supplementary Algorithm A.1.

For both samplers, the number of particles was set to 1,000 and the cooling schedule α was set to 0.2. For approximate inference using the top-hat kernel (Eq 2), the initial and terminal conditions were the same as those used for the linear model in the previous section. For exact inference using the Gaussian kernel (Eq 4), the initial tolerance was set to $\varepsilon_0 = 100$, and the algorithms were set to terminate when $\varepsilon_t = 1$.

3.3 The O’Hara-Rudy action potential model

The final model chosen for examination was the O’Hara-Rudy cardiac action potential model. Like most action potential models, the O’Hara-Rudy formulation models the change in membrane potential as a linear function of the total transmembrane current $I_{\text{tot}}(t)$:

$$\frac{dV}{dt} \propto I_{\text{tot}}(t) = \sum_{x=1}^m I_x(t), \quad (11)$$

which in turn is treated as the sum of m currents. Each such current is associated with a particular ion channel – a protein spanning the membrane that specializes in the transport or exchange of specific types of ions. These channels generally go through changes of shape in response to changes in cellular conditions (generally transmembrane potential), which affect the rate at which ions may pass through them. This allows for the restriction of ion channel activity to certain portions of the action potential, which is essential for restoring the cell to an excitable state between firings.

The O’Hara-Rudy model employs the so-called Hodgkin-Huxley formalism to model 13 ionic currents. This formalism, pioneered by the eponymous authors in 1952 [16], models transmembrane ionic conductance as a polynomial function of one or more “gating variables,” which are in turn modeled by time- and voltage-dependent ordinary differential equations (ODEs), as exemplified below:

$$I_x(t) = (V - V_x)G_x \prod_{i=1}^{M_x} (s_{x,i}(t))^{k_{x,i}} \quad (12)$$

$$\frac{ds_{x,i}}{dt} = f(s_{x,i}, t, V), \quad s_{x,i}(0) = s_{x,i,0}, \quad 0 \leq s_{x,i}(t) \leq 1 \quad \forall s_{x,i}, t, \quad (13)$$

where all $s_{x,i}$ are gating variables associated with current I_x , V_x is the “reversal potential” of the channel (the potential at which no net ionic flow is observed through the channel) and G_x is the maximum conductance attainable through the channel. This formalism treats each ion channel as a

collection of M_x types of subunits, each with multiplicity $k_{x,i}$, that must all be active in order to allow for the transport/exchange of ions. The activation probability of each of these subunits is modeled by a gating variable $s_{x,i}$, which can also be thought of as the active fraction of each subunit across the whole membrane. While the more general Markov model formulations can be (and have been) employed for ion channel models in an effort to more accurately capture their kinetics, the added computational complexity limits their use in large-scale tissue simulations [15]. Thus, Hodgkin-Huxley style models retain their popularity due to their simplicity and scalability.

In this study, inference was performed over the 13 maximum ionic conductance parameters (G_x in Eq 12). These are linked directly to expression levels of the associated genes, which are in turn a major source of variation between cell types and even individuals. While inference could additionally be performed on the parameters controlling gating (Eq 12-13), experimentalists generally perform these types of fitting using current data gathered from patch clamp experiments [24]. As these studies are typically performed prior to whole-cell action potential recordings, we treat these gating parameters as known in order to be congruous with modern practices.

For each current, a uniform prior of between 50% and 200% of the values reported in the original paper was assumed. An independent Gaussian proposal distribution was set over each conductance parameter, with standard deviation equal to 10% of its reported value (Table 1).

Action potentials were simulated using the functional curation extension to Chaste [7, 23], using a CellML [18] model file for the O’Hara-Rudy model annotated with metadata tags to allow the adjustment of maximum conductance parameters. The transmembrane potential was recorded every 0.25ms for 500ms.

For exact inference (SMC), the full time course data were considered by the sampler. “Experimental” data were generated by simulating the action potential using the parameter values reported in the original publication and applying independent Gaussian noise according to $\mathcal{N}(0, 0.25)$ at each time point [30]. Assuming knowledge of this noise model, the Gaussian acceptance kernel (Eq 4) with $\sigma_e = 0.25$ was employed.

For approximate inference (ABC-SMC), five commonly reported clinical summary statistics of the action potential were considered:

1. ADP90 - the time required for the membrane to re-polarize to 90% of its resting membrane potential.
2. ADP50 - the time required for the membrane to re-polarize to 50% of its resting membrane potential.

Table 1: **Reported values and inference priors for the 13 maximum conductance parameters in the O’Hara-Rudy action potential model.**

Parameter	Reported value ($\mu A/\mu F$)	Uniform prior range ($\mu A/\mu F$)
G_{Na}	75	[32, 150]
G_{NaL}	7.5×10^{-3}	$[3.2 \times 10^{-3}, 0.015]$
G_{to}	0.02	[0.01, 0.04]
G_{CaL}	1×10^{-4}	$[5 \times 10^{-5}, 2 \times 10^{-4}]$
G_{Kr}	0.046	[0.023, 0.092]
G_{Ks}	3.4×10^{-3}	$[1.7 \times 10^{-3}, 6.8 \times 10^{-3}]$
G_{K1}	0.1908	[0.09, 0.4]
G_{NaCa}	4×10^{-4}	$[2 \times 10^{-4}, 8 \times 10^{-4}]$
G_{NaK}	30	[15, 60]
G_{Nab}	3.75×10^{-10}	$[1.8 \times 10^{-10}, 7.5 \times 10^{-10}]$
G_{Kb}	3×10^{-3}	$[1.5 \times 10^{-3}, 6 \times 10^{-3}]$
G_{pCa}	5×10^{-4}	$[2.5 \times 10^{-4}, 1 \times 10^{-3}]$
G_{Cab}	2.5×10^{-8}	$[1.25 \times 10^{-8}, 5 \times 10^{-8}]$

3. Peak potential - the largest positive membrane potential observed over the course of the action potential.
4. Resting potential - the steady-state membrane potential (approximated by the most negative membrane potential observed during the action potential).
5. Maximum upstroke velocity - the largest value of $\frac{dV}{dt}$ observed during the action potential.

These quantities, represented visually in Fig 4, were calculated using the standard cardiac library available in functional curation Chaste [7]. “Experimental” data were generated by simulating the action potential using the parameter values reported in the original publication, adding noise as was done for exact inference data, and then calculating the five summary statistics [30]. The five summary statistics were treated as a single vector, with each being normalized by its “experimental” value to control for differences in scale. Assuming no knowledge of the noise model, the top-hat acceptance kernel (Eq 2) was employed over the vector of five statistics.

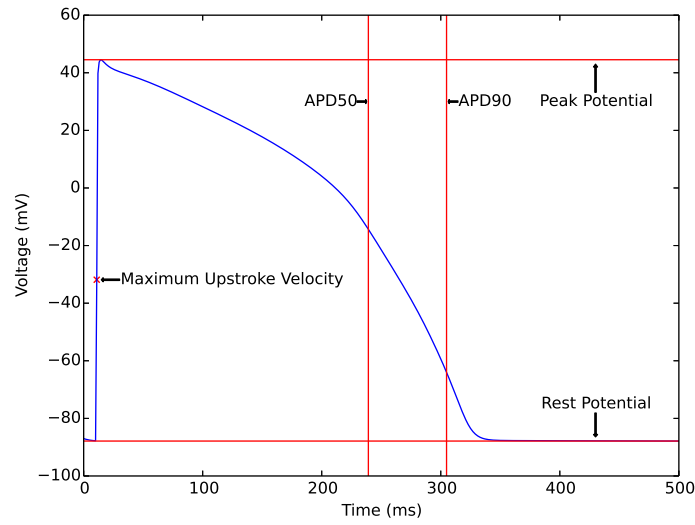


Figure 4: **Visual representation of the summary statistics used by the approximate samplers for the O’Hara-Rudy action potential model.** Vertical lines represent the times (ms) used for the values of APD50 and APD90 (respectively labeled), while horizontal lines represent the voltages (mV) used for the values of peak and rest potential (respectively labeled). The red “x” indicates the point at which $\frac{dV}{dt}$ (mV/ms) is greatest, i.e. the point of maximum upstroke velocity.

Due to the computational complexity added by calculating summary statistics, the Del Moral sampler was limited to 1,000 particles for both probabilistic and approximate inference. To ensure gradual evolution of the posterior, and consequently make Eq 6 a good approximation of the true importance sampling weights, the cooling schedule α was set to 0.2, and resampling was triggered at 60% ESS (as opposed to 50% on line 16 of Algorithm 3). An additional minimum of 30% ESS was set for all posterior updates (as opposed to the simpler nonzero requirement on line 10).

The approximate Del Moral sampler was set to terminate when the threshold ε fell below the value of the objective function evaluated for data simulated under the generating parameters (Table 1). The probabilistic solver was set to terminate at $\varepsilon = 1$, the point at which exact posterior sampling was being performed.

4 Results

4.1 The linear model

The first model chosen for a comparative study of the two samplers was the linear growth model with Gaussian observation error (Eq 8). Using the known theoretical posterior on the model parameter (Eq 9), we can verify if the empirical posteriors produced by our exact SMC algorithms are correct. We note that the theoretical posterior mean, as shown in Figs 5, 7, and 8 differs from the generating value $a = 5$ by less than 0.01%.

We began with a comparison of the ABC inference capabilities of both samplers on the model, performing posterior inference with the Toni and Del Moral samplers using the standard top hat kernel (Eq 2), which does not match the true (Gaussian) distribution of observation error (Eq 8). We first compared the two samplers with a common number of particles (1000) and $\alpha = 0.5$, aiming to halve the maximum error among particles in the population after each update. Both samplers terminated at $\varepsilon_T = 301$ and the resulting posteriors are shown in Fig 5. We see that while both posteriors capture the theoretical mean and span the same range of values, the posterior produced by the Del Moral sampler shows significantly higher variance than the one produced by the Toni sampler, which shows the uniformity expected when employing a top-hat kernel.

We sought to investigate the effects of increased population size and relaxed cooling schedule on the uniformity of the Del Moral posterior estimates for the linear model. Both of these changes would increase the similarity between subsequent posterior estimates, and thus would be expected to im-

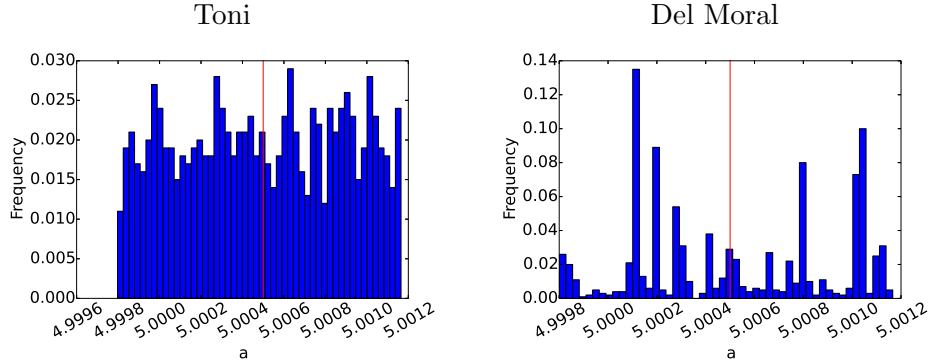


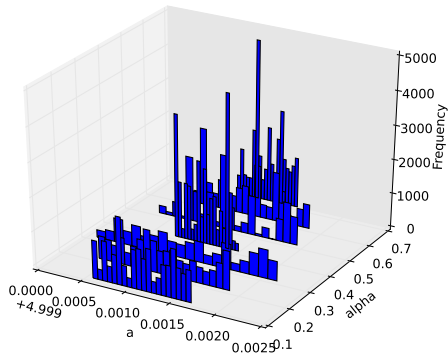
Figure 5: **Posteriors generated by the Toni and Del Moral samplers over the linear model when using a top hat acceptance kernel.** Vertical bars represent frequencies of binned values of a among particles in the posterior estimate, while the vertical red line represents the theoretical posterior mean calculated with Eq 9.

prove the approximation of the true importance sampling weights employed by the Del Moral algorithm [14]. In Fig 6 we see that either by increasing the number of particles from 1,000 to 10,000 or by decreasing the cooling schedule α below 0.3, the variance in the resulting Del Moral posterior estimate decreases dramatically and it approaches the uniformity of the Toni posterior. While the variance in the final posterior decreases proportionally to α for both 1,000 and 10,000 particle samplers, the effect is much more pronounced at the lower population size.

We next ran both ABC-SMC variants on the same linear model, but employing the probabilistic acceptance kernel that matches the observational error distribution (Eq 4). Informed by the results of the approximate study of the linear model, we decreased the cooling schedule parameter α to 0.25 for both samplers and increased the number of particles employed by the Del Moral sampler to 10,000 to improve the quality of the weighting approximation. We continued to use 1,000 particles for the Toni sampler due to its reasonable performance under these settings when performing approximate inference.

Applying the probabilistic Del Moral sampler, we see in Fig 7 that the resulting posterior estimate shows a high degree of homology with the theoretical posterior. The histogram of values of the slope parameter a appears normally distributed, and captures the theoretical mean almost exactly. Ad-

1,000 Particles



10,000 Particles

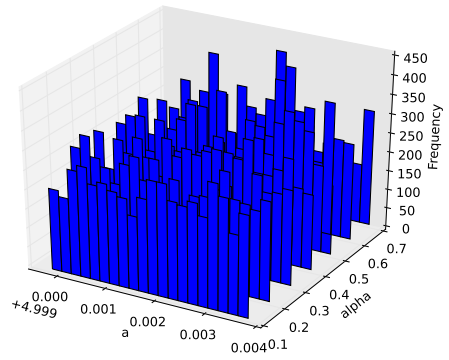


Figure 6: **Effect of population size and α in the Del Moral sampler on posteriors produced over the linear model.** Posteriors produced using 1000 (left) and 10,000 (right) particles at varying values of the algorithmic parameter α . Histograms parallel to the “a” axis represent the frequencies of particles in the final posterior estimate generated under the corresponding value of α on the “alpha” axis. Note that the a -axis has been shifted in both graphs, as noted in the bottom-left marking on each.

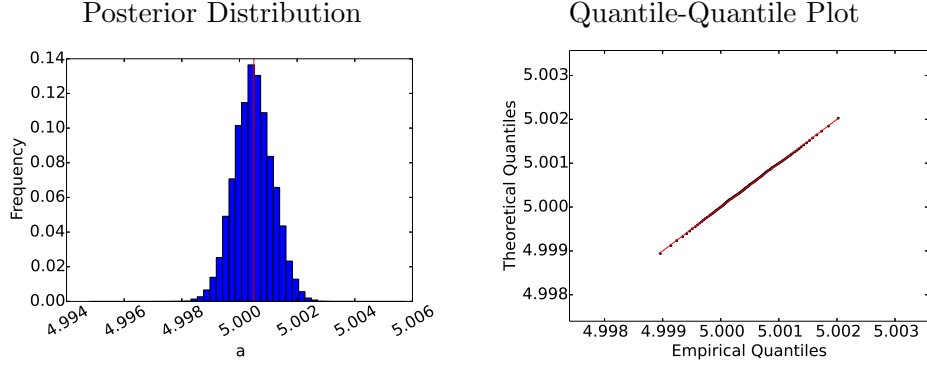


Figure 7: **Posterior generated by the Del Moral sampler over the linear model when using a Gaussian acceptance kernel.** (Left) Histogram of particle frequencies, with red vertical line indicating theoretical mean. (Right) Quantile-quantile plot comparing the PPF (percent point function) of the estimated posterior (x-axis) to that of the theoretical posterior (y-axis) by paired estimation at 100 equally spaced quantiles.

ditionally, the quantile-quantile plot, which plots the percent point function (PPF) of the theoretical posterior distribution (Eq 9) against that of the empirical posterior, shows a linear relationship with slope 1, indicating that the two distributions are Gaussian with equal variance.

In order to similarly employ the probabilistic Toni sampler, however, we must additionally consider the value of the normalizing constant c_{ε_t} in Eq 4. In the Del Moral sampler, the constant is cancelled by the Metropolis-Hastings update, but in rejection-based samplers like Algorithm 2, it controls the tradeoff between a reasonable acceptance rate (line 13) and correctness of the sampler. Wilkinson suggests choosing for c_{ε_t} the modal value of the acceptance kernel, where $SE(\theta) = 0$ [44], which would improve the overall acceptance rate but still bound the function between 0 and 1. In the presence of noise, however, no single parameter set will generate such error-free predictions, so we instead employed the empirical mode $\rho_{\varepsilon_t}(\hat{\theta})$, where $\hat{\theta}$ is the maximum likelihood estimate of the parameters given the noisy data.

In Fig 8 we see the sensitivity of the Toni sampler to three values for c_{ε_t} chosen relative to this empirical mode. When $c_{\varepsilon_t} = 0.1\rho_{\varepsilon_t}(\hat{\theta})$, deliberately below the modal value, we see the posterior take on a nearly uniform character around the mean, with a high nonlinearity in the quantile-quantile relationship confirming disagreement with the theoretical Gaussian poste-

rior. Conversely, when $c_{\varepsilon_t} = 10\rho_{\varepsilon_t}(\hat{\theta})$, the algorithm terminates early with final $\varepsilon_T > 3$ due to repeatedly exceeding the I_{max} criterion (line 15 of Algorithm 2), and the marginal and quantile-quantile plots reveal the final posterior estimate to be overdispersed relative to the theoretical posterior in Eq 9. Finally, when $c_{\varepsilon_t} = \rho_{\varepsilon_t}(\hat{\theta})$, we see overall agreement between the empirical posterior and the theoretical one, despite some leveling off near the mode. In both cases of $c_{\varepsilon_t} = 0.1\rho_{\varepsilon_t}(\hat{\theta})$ and $c_{\varepsilon_t} = \rho_{\varepsilon_t}(\hat{\theta})$ the value of the acceptance kernel was observed to take on values greater than 1, although with far less frequency in the latter, and only at later iterations.

4.2 The polynomial model

We next sought to explore the effect of the dimension of parameter space on the convergence of both algorithms by performing inference on the polynomial model detailed in Eq 10 with an increasing number of free parameters, P . For a number of free parameters increasing from 3 up to 16, and while keeping all other parameters constant between algorithms, we measured the convergence of both algorithms by recording the standard deviation of the posterior weights as the iteration proceeds, which should approach a constant value. For the Toni sampler, we performed only approximate inference using the top-hat kernel (Eq 2) over the full trace to avoid the confounding effects of the choice of normalizing constant (as illustrated in the previous section). For the Del Moral sampler, we performed inference using both the exact and approximate variants.

As we see in Fig 9, all samplers show an initial increase in posterior weight variance from the uniform prior across all values of P . In both the approximate and exact Del Moral samplers, we see that regardless of the value of P , the sampler maintains a ceiling on weight variance as the importance sampling proceeds, thanks to resampling steps which redistribute weight across the posterior (indicated by sudden drops in weight variance). It is difficult to make a quantitative comparison between the exact and approximate versions of the sampler at a given iteration, due to the differing interpretations of the ε tolerance parameter, but for a fixed set of particles we would expect the weights of the approximate Del Moral sampler to have a higher variance due to the approximation of the Gaussian likelihood function with the top-hat kernel.

In the approximate Toni sampler, we see that at $P \leq 6$, the variance of posterior weights largely decreases as the iterations proceed, yet at large values of P the variance swings wildly between iterations, with effects becoming more pronounced at higher values of P . Upon termination, the posterior re-

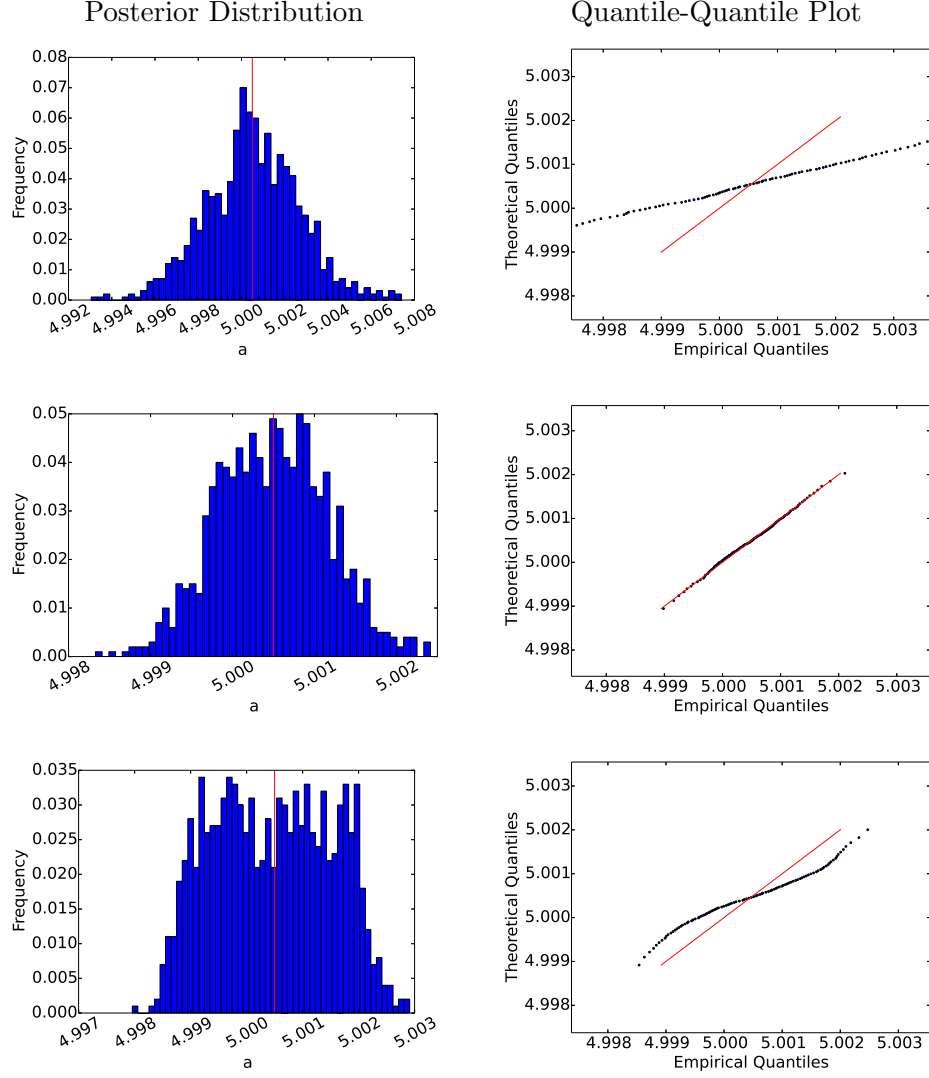


Figure 8: **Effect of c_{ε_t} on posteriors generated by the Toni algorithm over the linear model with a Gaussian acceptance kernel** (Top) $c_{\varepsilon_t} = 10\rho_{\varepsilon_t}(\hat{\theta})$, (Middle) $c_{\varepsilon_t} = \rho_{\varepsilon_t}(\hat{\theta})$, (Bottom) $c_{\varepsilon_t} = 0.1\rho_{\varepsilon_t}(\hat{\theta})$, where $\hat{\theta}$ is the maximum likelihood estimate of the slope parameter. (Left) Histogram of particle frequencies, with red vertical line indicating theoretical mean. (Right) quantile-quantile plot comparing the PPF of the estimated posterior (x-axis) to that of the theoretical posterior (y-axis) at 100 equally spaced quantiles.

turned by the Toni sampler when $P = 16$ contained a single particle whose weight accounted for over 34% of the distribution.

4.3 The O’Hara-Rudy action potential model

As with the high-dimensional polynomial model described in the previous section, we found that the Toni sampler struggled with highly variable particle weights when performing inference on the 13-parameter O’Hara-Rudy model (described below), resulting in posterior estimates collapsing to single-point estimates (not shown). Consequently, we have chosen to employ only the Del Moral sampler in order to compare probabilistic inference on action potential traces with approximate inference on summary statistics of the action potential. In this analysis, we employ simulated data for all solvers, as we aim to investigate the identifiability of the model *a priori*, before any experimental recordings are carried out.

We began our analysis of the O’Hara-Rudy action potential model by performing posterior inference over the full 13-parameter model, using simulated action potential data with added noise. From the selection of marginal distributions presented in Fig 10 (a full visual summary of the marginal posteriors can be found in Supplementary Fig. B.3), we see that the marginal posteriors are well-constrained for both major currents such as G_{Na} and minor currents such as $G_{Na,b}$, but may show noticeable mean deviation from the values used to generate the data. We expect deviations of the posterior means from generating values due to the effects of random noise (as in the linear model), as well as redundancies in the system, in which parameters may vary in a compensatory manner so that model output remains largely unchanged. The presence of the latter effect can be confirmed by Supplementary Fig. B.2, in which we observe great consistency between action potentials simulated under the mean posterior values and the data used to fit the model despite their differing values. It is also worth noting that the width of these posterior distributions is a function of the number of sample points employed, and thus the location of the generating values outside of some “credible interval” of the posterior mean can be most accurately attributed to low variance in the estimator.

To ensure that the solver was not becoming stuck in a local minimum, we repeated the fitting 10 times and reported the results in Table 2 (top). For nearly all model parameters, particularly the major currents G_{Na} , G_{to} , G_{Kr} , and G_{K1} , the posterior mean averaged over these ten repeated inferences showed strong agreement with the generating values, as well as low standard deviation across the repeats. As such, the mean posterior devia-

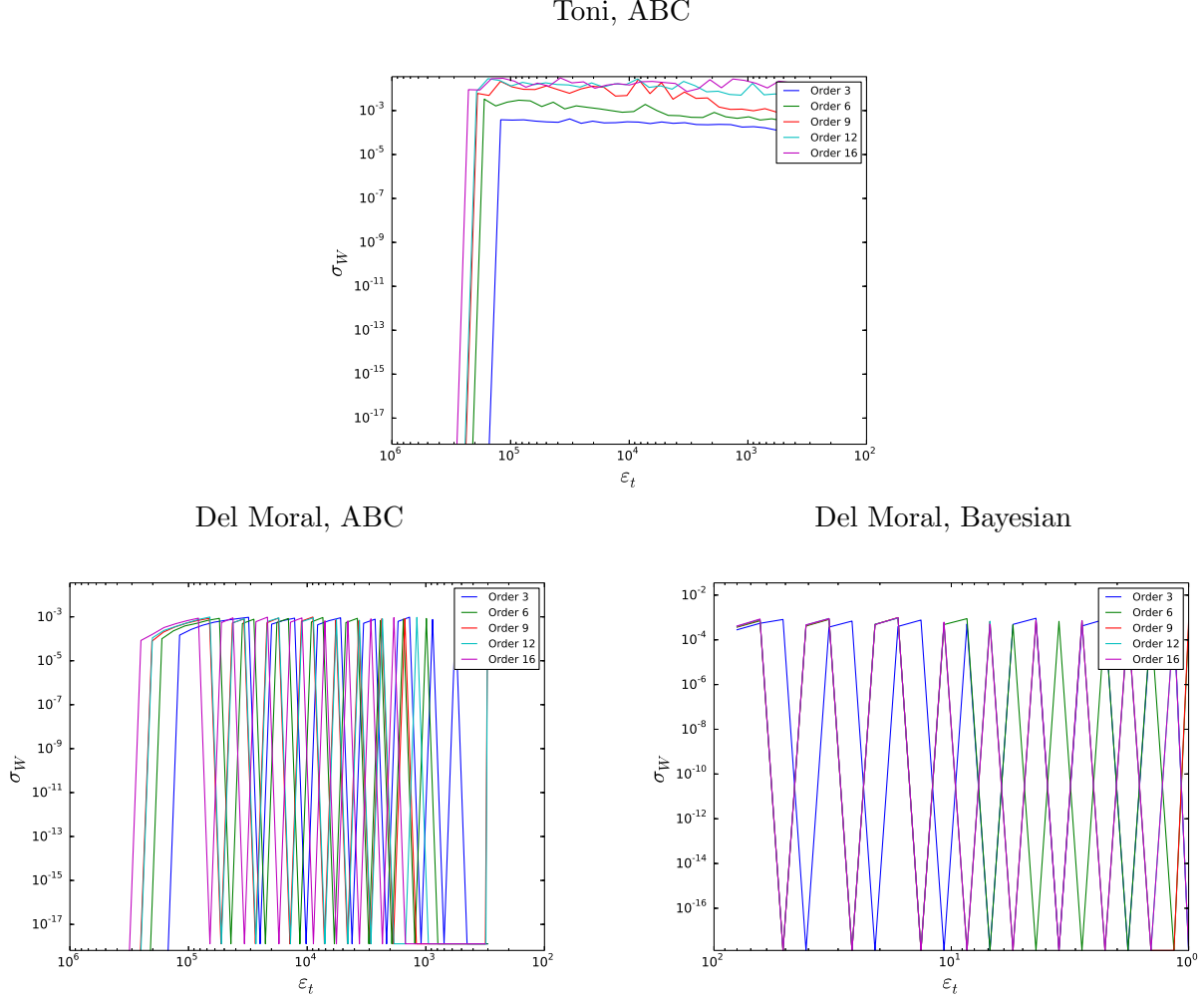


Figure 9: **Effect of model dimension on stability of Toni and Del Moral samplers.** Variance of particles weights (σ_w , plotted in log-scale) in the approximate Toni sampler (top) and both the approximate (bottom left) and exact (bottom right) Del Moral samplers, each with 1,000 particles, as they iterate over posterior estimates of the polynomial model with increasing numbers of parameters (indicated by their “order”). Iteration of the algorithm is marked on the x -axis (log-scale) by the value of the algorithmic threshold parameter (ε_t).

tions observed in single instances of inference are likely due to the difficulty of covering the 13-dimensional space with only 1,000 sampler particles. While greater consistency between the posterior mean and the generating values may be obtained by increasing the population size of the sampler, computational restrictions imposed by the calculation of summary statistics in our subsequent analyses limited us to 1,000 particles in the approximate sampler. As such, we maintained this number across both exact and approximate samplers to allow for maximal homology between the two during our comparisons.

We next fit the 13-parameter model to five summary statistics of the same noisy action potential – APD50, APD90, peak potential, resting potential, and maximum upstroke velocity (see Section 3.3) – using the approximate Del Moral sampler. In Fig 11 (and Supplementary Fig. B.4), we see a general widening of the approximate marginal posteriors relative to those produced using full time course data, as well as multimodality approaching uniformity in the posteriors over parameters such as G_{Nab} and G_{NaCa} . In Table 2 we quantify the increase in uncertainty resulting from the use of summary statistics by presenting the ratio of posterior to prior standard deviation, both when full data are employed (σ_P/σ_0) and when only summary statistics are employed (σ_A/σ_0). These ratios normalize against the scale and initial spread of the parameters, allowing for the assessment of relative uncertainty about a parameter after inference.

In Table 2 (top), we see that that G_{Na} and G_{Kr} remain the two best constrained parameters, as indicated by the value of σ_A/σ_0 , with nothing else even reaching 50% reduction in uncertainty. If we compare the uncertainty reduction for these two descriptors to that when full time course data were used (σ_P/σ_0), however, we see that G_{Na} only shows less than two-fold relative increase in uncertainty when summary statistics were employed, while G_{Kr} shows a nearly 10-fold increase in relative uncertainty when these summary statistics are employed. Thus, while we can tell simply by examining posterior-prior variance ratios that the five summary statistics contain the most information about G_{Na} , and to a lesser extent G_{Kr} , we can tell by comparing to the corresponding ratios from exact inference over time-course data that a disproportionate amount of information is being lost by some major (G_{Kr} , G_{to} , G_{K1}) and minor (G_{Nab} , G_{NaK}) currents when these summary statistics are employed.

In order to control for under-determination in the system (the 13 free parameters exceeding the five data points) when summary statistics are employed, we repeated both fittings – using full time trace data for the exact Del Moral sampler and using summary statistic data for the approximate

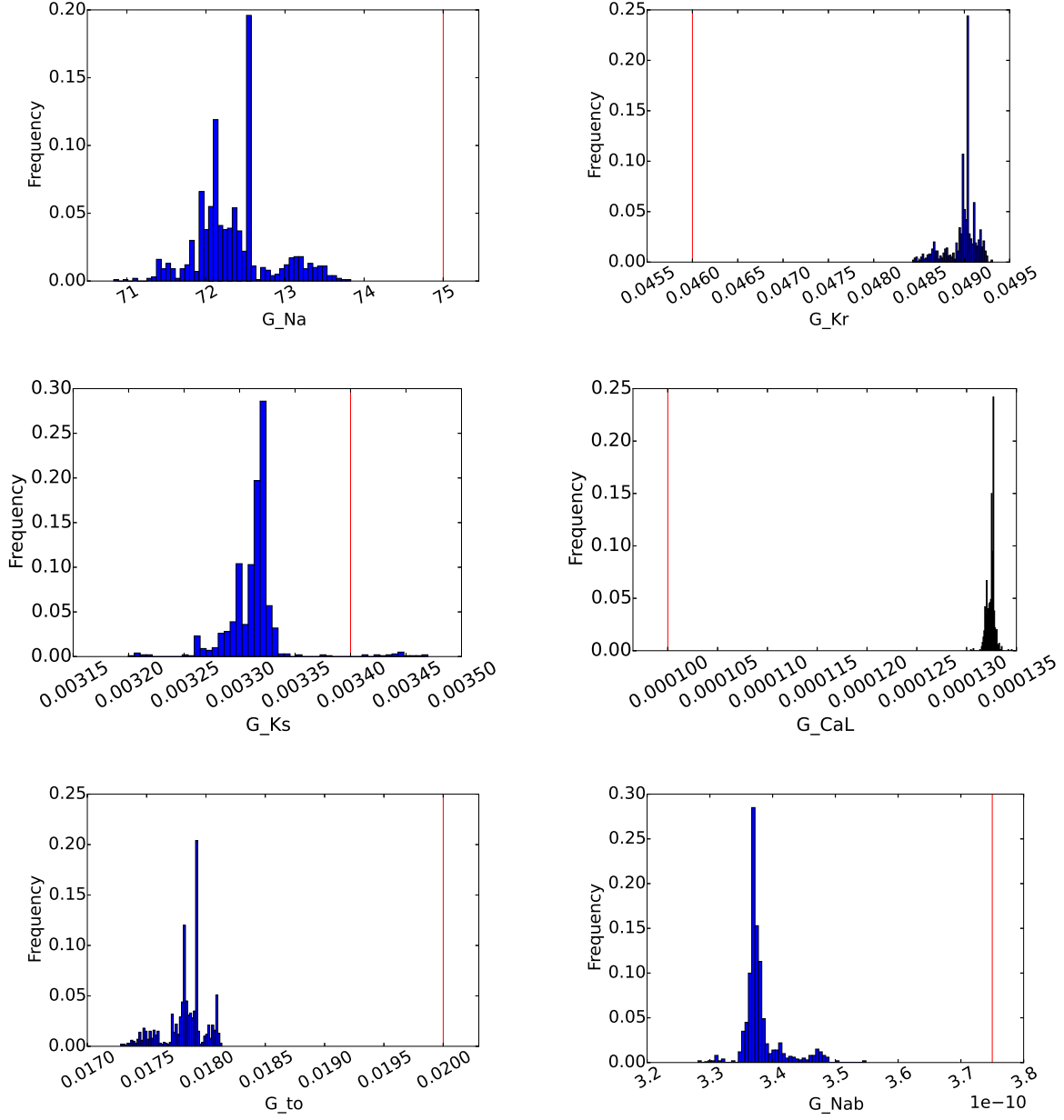


Figure 10: **Selected marginal distributions from the posterior produced by the exact Del Moral algorithm on the 13-parameter O'Hara-Rudy model.** Inference was performed using full time trace data. Generating values for the simulated data are indicated by vertical red lines.

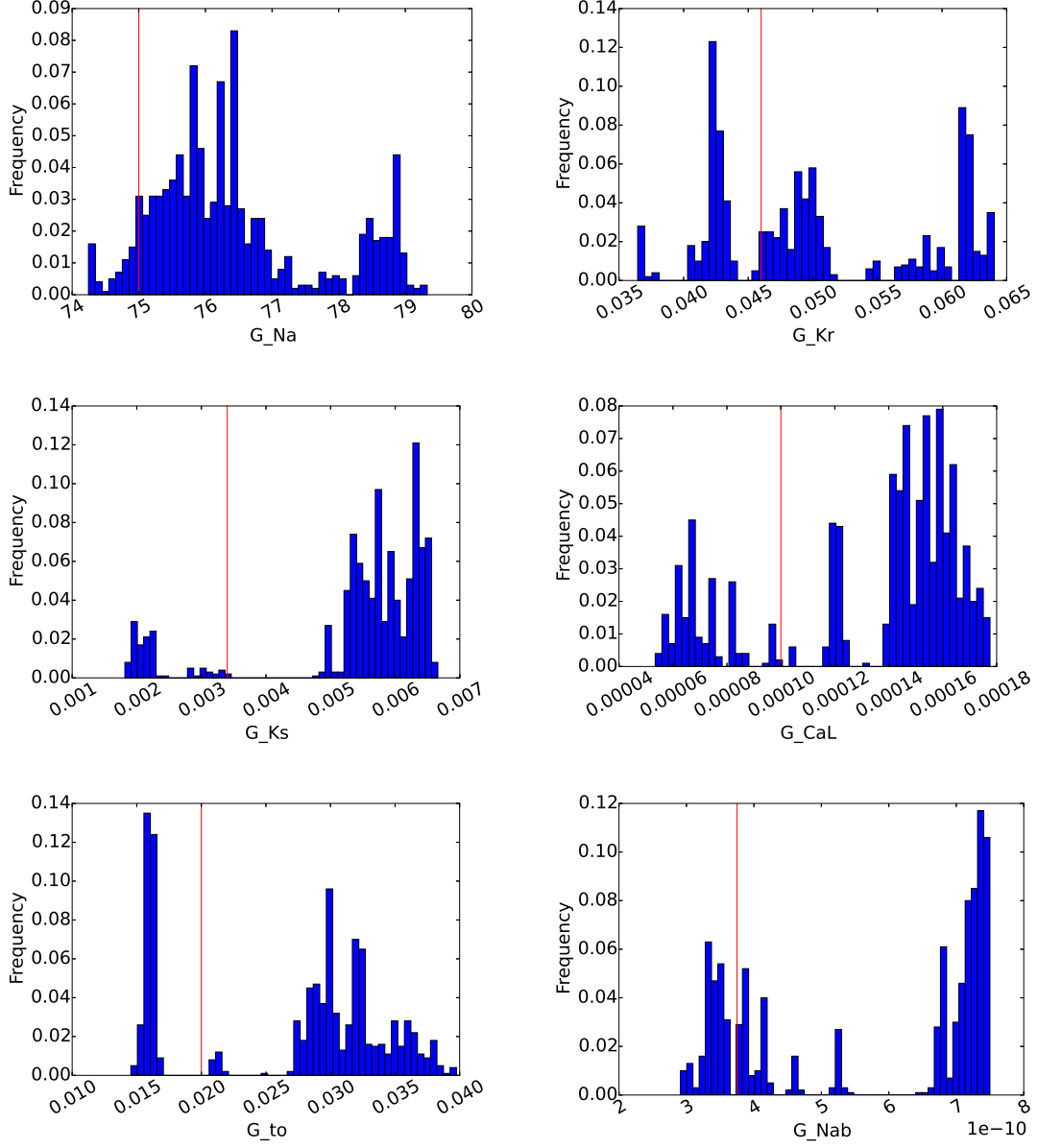


Figure 11: **Selected marginal distributions from the posterior produced by the approximate Del Moral algorithm on the 13-parameter O'Hara-Rudy model.** Inference was performed using summary statistic data. Generating values for the simulated data are indicated by vertical red lines.

Table 2: Summaries of the marginal distributions over free parameters of the O’Hara-Rudy model produced by the Del Moral algorithm.

13 Free Parameters

Name	True Value	μ_P (std.)	σ_P/σ_0 (std.)	μ_A (std.)	σ_A/σ_0 (std.)
G_{Na}	75	75.8 (3.90)	0.0173 (0.00855)	76.4 (0.752)	0.0292 (0.0115)
G_{NaL}	7.5×10^{-3}	8.06×10^{-3} (2.80×10^{-3})	0.131 (0.252)	1.04×10^{-2} (8.50×10^{-4})	0.729 (0.261)
G_{to}	0.02	0.0215 (2.72×10^{-3})	0.0453 (0.0725)	0.0252 (3.89×10^{-3})	0.658 (0.209)
G_{CaL}	1×10^{-4}	1.11×10^{-4} (1.43×10^{-5})	0.101 (0.181)	1.22×10^{-4} (2.39×10^{-5})	0.684 (0.238)
G_{Kr}	0.046	0.0467 (5.01×10^{-3})	0.0315 (0.0615)	0.0497 (4.11×10^{-3})	0.297 (0.0839)
G_{Ks}	3.4×10^{-3}	4.45×10^{-3} (1.57×10^{-3})	0.0756 (0.110)	4.63×10^{-3} (9.38×10^{-4})	0.729 (0.249)
G_{K1}	0.1908	0.192 (0.0171)	0.0400 (0.0501)	0.184 (0.0461)	0.765 (0.301)
G_{NaCa}	8×10^{-4}	1.04×10^{-3} (2.98×10^{-4})	0.155 (0.251)	1.01×10^{-3} (1.95×10^{-4})	0.779 (0.217)
G_{NaK}	30	39.8 (12.4)	0.0984 (0.167)	42.9 (4.95)	0.691 (0.117)
G_{Nab}	3.75×10^{-10}	5.04×10^{-10} (1.38×10^{-10})	0.0564 (0.0804)	4.58×10^{-10} (1.07×10^{-10})	0.804 (0.235)
G_{Kb}	3×10^{-3}	3.17×10^{-3} (9.73×10^{-4})	0.189 (0.317)	3.72×10^{-3} (8.18×10^{-4})	0.696 (0.158)
G_{pCa}	5×10^{-4}	6.664×10^{-4} (2.01×10^{-4})	0.104 (0.122)	5.73×10^{-4} (7.44×10^{-5})	0.642 (0.157)
G_{Cab}	2.5×10^{-8}	3.63×10^{-8} (9.51×10^{-9})	0.111 (0.238)	3.31×10^{-8} (4.22×10^{-9})	0.676 (0.212)

5 Free Parameters

Name	True Value	μ_P (std.)	σ_P/σ_0 (std.)	μ_A (std.)	σ_A/σ_0 (std.)
G_{Na}	75	75.0 (0.389)	0.0221 (4.05×10^{-3})	75.7 (0.0736)	0.0123 (7.32×10^{-4})
G_{Kr}	0.046	0.0461 (5.94×10^{-5})	3.14×10^{-3} (1.31×10^{-3})	0.0458 (2.70×10^{-5})	0.0103 (2.37×10^{-4})
G_{pCa}	5×10^{-4}	6.07×10^{-4} (1.46×10^{-4})	0.521 (0.216)	6.20×10^{-4} (8.47×10^{-5})	1.04 (0.167)
G_{to}	0.02	0.0201 (2.45×10^{-4})	0.0449 (0.0210)	0.0204 (4.54×10^{-4})	0.193 (0.0184)
G_{Cab}	2.5×10^{-8}	2.92×10^{-8} (5.17×10^{-9})	0.474 (0.267)	2.86×10^{-8} (1.75×10^{-9})	0.978 (0.0887)

Columns marked “ μ_P ” and “ σ_P/σ_0 ” denote the posterior mean and posterior-prior standard deviation ratio for each parameter when exact inference is performed, while columns marked “ μ_A ” and “ σ_A/σ_0 ” denote corresponding quantities from the approximate sampler employing the five summary statistics described in Section 3.3. Each of these quantities is reported as the average over 10 repeated inferences, with corresponding standard deviations in parentheses.

Del Moral sampler – allowing only the five currents whose conductances were best constrained by the summary statistics in our previous inference (as indicated by σ_A/σ_0 in Table 2 (top)) to vary, and holding all other parameters at their reported values. While G_{Na} and G_{Kr} were clearly the two best constrained, the next best three – G_{pCa} , G_{to} , and G_{Cab} – belonged to a class of parameters with roughly identical values of σ_A/σ_0 , and thus their selection was somewhat arbitrary.

We see in Table 2 (bottom), as well as Supplementary Fig. B.5 and Sup-

plementary Fig. B.6, that under these conditions there is better agreement between posterior mean and reported value for both exact and approximate inference, demonstrating the solver’s increased ability to cover the parameter space in lower dimensions for a fixed number of particles.

In Table 2 (bottom), we see that relative to the posteriors on the 13-parameter model, the three major currents in this model – G_{Na} , G_{Kr} , G_{to} – show similar or decreased posterior uncertainty when time series data are employed, and dramatically reduced uncertainty when summary statistics are employed. This suggests that when compensatory interactions between model currents are removed, the approximate sampler in particular is better able to constrain these individual major currents. We notice, however, by comparing σ_A/σ_0 to σ_P/σ_0 in the five-parameter model, that G_{Kr} and G_{to} still exhibit a disproportionate increase in posterior uncertainty relative to G_{Na} when employing summary statistics, just as they did in the 13-parameter model. G_{Na} in fact exhibits a slight decrease in σ_A/σ_0 relative to σ_P/σ_0 , though this is likely an artifact of the solver as the information contained in the summary statistics is necessarily a subset of the information contained in the full time trace.

Even when time trace data are available, however, it is relatively more difficult to estimate background currents such as G_{pCa} and G_{Cab} in the five-parameter model. In Table 2 (bottom), we see that the value of σ_A/σ_0 for these minor currents is much higher than that of the major currents, and actually shows an average-case increase in uncertainty relative to the 13-parameter model. This may be due to the effects of noise added to the action potential, which causes the true maximally likely model parameters to differ slightly from the parameters used to generate the data when only a finite number of observations are used during fitting. Thus, when we fixed 8 of the model parameters to the values from Table 1 (as opposed to the best values found when fitting to the full trace), we may have influenced the solver to explore a region of “flatter” likelihood for the parameters G_{pCa} and G_{Cab} , resulting in an increase in uncertainty. Because summary statistics were not shown to have significantly more information about these parameters than others in the 13-parameter model, it is unsurprising that the posteriors produced by approximate inference in this case are completely uninformative relative to the prior, as indicated by a σ_A/σ_0 of 1 in Table 2 (bottom). The degree of apparent constraint of these minor current parameters in the approximate posterior over the 13-parameter model was likely an artifact of under-determination of the system and an insufficient number of particles in the solver.

5 Computation

All code used in this study was written in Python.

We found that, on a 6-core (hyperthreaded) Intel Xeon machine @ 3.5Ghz with 16GB RAM, performing inference on the linear model using the approximate Toni algorithm took 3090 seconds, while inference using the Del Moral algorithm took 400 seconds. The calculation of weights for our implementation of the Toni algorithm contributed significantly to this runtime, taking on the order of 60 seconds per iteration. We recognize inefficiencies in this particular implementation brought on by an abstraction layer that we have enforced, and as such future implementations may significantly improve this step in particular, which amounts to evaluating and summing N^2 PDFs. While the exact Del Moral algorithm was almost identical in speed to the approximate version, we found that the performance of the exact Toni algorithm varied greatly depending on the normalizing constant, which controls the acceptance rate and, consequently, the accuracy of the algorithm, as we have shown.

On the same machine, a single forward simulation of the O’Hara-Rudy model took 0.064 seconds. Pairing this simulation with summary statistic calculation brought the overall forward simulation time up to 0.45 seconds, as calculation of APD with post-processing functions in functional curation is particularly expensive. As a result, execution of the approximate Del Moral algorithm on the full 13-parameter model took 72,716 seconds, while the exact Del Moral algorithm, which did not perform summary statistic calculation, took only 7,891 seconds. On the five-parameter model, the approximate Del Moral algorithm took 63,509 seconds, while the exact variant took only 7,512 seconds.

6 Discussion

When we performed approximate inference on the linear model, we saw that the Del Moral sampler performed noticeably worse than the Toni sampler under identical settings, returning highly-varying posteriors inconsistent with the uniformity expected under such an inference strategy. We presumed that this increase in variance was due to the approximation employed by Eq 6, which only holds when the evolution of the posterior estimates is very gradual. We confirmed this by showing that increasing the number of particles in the sampler (to mitigate the effects of randomness) and/or relaxing the cooling schedule (controlled by the α parameter), both of which reduce

the difference between subsequent estimates, lead to a drastic increase in posterior uniformity.

The performance of the Del Moral sampler on this problem was likely additionally hindered by the fact that we limited ourselves to a single noisy data set. When using the top-hat kernel and a single data realization, the weights of the Del Moral sampler (Eq 5) become binary indicators of whether or not the particle meets the final threshold. This scheme has inherently high variation relative to the Toni weighting scheme (Eq 5), in which all particles are guaranteed to have nonzero weight at all times due to rejection sampling, and weights are smoothed by a denominator informed by the previous posterior estimate and proposal distribution. When repeated experimental data are available, the variance of the Del Moral weighting scheme can be reduced, as the weights may instead indicate the *fraction* of experimental data that is within a threshold of the simulated data rather than a binary quantity. Similarly, when employing a stochastic model, one may generate multiple simulations (at added computational cost) for each proposed parameter set and weight particles based on the fraction of simulated data that fall within a threshold of the experimental data. In the case of deterministic models and singular data, however, our results indicate that the approximate Del Moral sampler may require a larger number of particles and/or reduced cooling schedule to match the performance of the approximate Toni sampler.

Despite this advantage, our investigation into the linear and polynomial models revealed that the Toni sampler is highly sensitive to the choice of a kernel normalizing constant when performing exact inference, and relatively sensitive to the dimension of the model when performing either exact or approximate inference. When performing Gaussian inference on the linear model, we observed that setting the kernel normalizing constant too high resulted in a large number of rejected samples in later iterations that eventually caused the algorithm to terminate before converging to the true posterior (achieved when $\varepsilon = 1$), while setting the constant too low resulted in a uniform posterior due to decapitation of the kernel. While in theory setting the normalizing constant to the modal value of the distribution will cap the value of the acceptance kernel at 1 and lead to exact estimation, we have found that in practice small errors in estimation resulting from experimental noise can result in underestimation and decapitation of the posterior even for the linear model. For more complex models, locating the true modal value will be harder and more expensive, and without extensive system-specific knowledge it would be difficult to choose a coefficient that avoids truncation while ensuring a reasonable acceptance rate.

As the model increases in complexity, as in the case of the polynomial model as we increased the dimensionality of the parameter space, we demonstrated the tendency of particle weights in the Toni sampler to fluctuate highly between iterations. When the parameter space is large relative to the number of particles in the posterior, we expect the denominator of the Toni weighting scheme (Eq 5) to have a high variance due to poor coverage of the proposal distribution $K(\theta|\theta^*)$ by the posterior particles. As such, even with a uniform prior, certain particles may become severely upweighted between iterations and the posterior estimate may collapse to one or several points. The Del Moral weighting scheme (Eq 6) is immune to this effect, and the resampling step in the algorithm itself (Algorithm 3, lines 16-21) effectively caps the variance that the particle weights may attain. Beyond simply increasing the number of particles, which comes at computational cost in the weighting scheme, the performance of the Toni sampler might be improved by employing an adaptive proposal distribution that can redistribute weight on distributions that become too peaked, or by explicit resampling as in the Del Moral scheme, but under current formulation the Del Moral sampler is preferable for high dimensional parameter spaces.

The effects of model dimensionality were also seen during our investigation into the O’Hara-Rudy cardiac action potential model. For a fixed number of 1,000 particles, the Toni sampler struggled with convergence due to the aforementioned weighting difficulties, but our results from the probabilistic Del Moral sampler suggest that it may be inherently hard to cover the 13-dimensional parameter space with this number of particles due to the presence of local minima. Indeed, we saw that fixing a number of the model currents allowed for the more reliable recovery of the globally optimal values for the remaining free parameters in both the exact and approximate Del Moral samplers. We believe this result highlights the importance of *a priori* analyses of high-dimensional biological models, where fitting to simulated data with a known solution can highlight the ability of the solver to recover a known set of parameters in the face of system redundancy.

By examining the posteriors produced by approximate Bayesian inference over the summary statistics APD50, APD90, peak potential, resting potential, and maximum upstroke velocity, we found that these measurements contained the greatest amount of information about the conductances of the fast sodium (G_{Na}) and rapid delayed rectifier (G_{Kr}) currents, as measured by the relative width of their marginal posterior distributions after approximate inference. This makes intuitive sense, as fast sodium current activity is extremely localized to the rapid depolarization phase of the action potential, which is well characterized by maximum upstroke velocity and peak

potential, while the rapid delayed rectifier current is involved in repolarizing the membrane in the later stages of the action potential, a phase which may be well characterized by ADP50, APD90, and resting potential. Other model currents whose phases of activity were not characterized well by the summary statistics, such as the slow delayed rectifier current G_{Ks} which describes a small repolarization of the membrane occurring after reaching peak potential but long before APD50 or APD90, showed very little decrease in posterior variability relative to the prior. This suggests that a certain amount of domain specific knowledge can be employed by experimentalists when choosing the summary statistics to use in fitting currents within a model. However, as we have shown, once these summary statistics have been chosen, ABC methods should then be used to assess the extent to which these statistics can capture the information necessary to fit to all of the parameters of interest. In our case it is clear that the chosen set of five summary statistics contain *sufficient* information about only one of the parameters of interest, G_{Na} , as this was the only parameter that could be recovered to a precision similar to that achievable with the full data set. Care should therefore be taken not to over-interpret changes in the observed values of such summary statistics in experimental data as being in any way indicative of changes in the underlying values of the conductance parameters other than G_{Na} , and to a more limited extent G_{Kr} and G_{to} .

Future work on these samplers will move beyond the realm of simply assessing the amount of information contained in a given experiment, such as the fixed set of summary statistics presented here, and into the automated design of maximally informative experiments (often referred to as “optimal experimental design”). This may consist of optimization over a parameterized input to the system, such as a voltage signal to be applied to the membrane before observing its response over time, or the selection of the subset of summary statistics that contain the most information about the system parameters from a list of possible measurements. Summary statistical data might also be employed for reliable model selection in an ABC context, if multiple formulations are being considered to describe the same phenomenon [33]. Such automated methods could aid experimentalists in complex systems where the most informative measurements to take are not immediately obvious, or may be expensive to record/calculate, and a tradeoff between information content and simplicity must be reached for experimental measurements. Algorithms for automated experiment design would involve the repeated execution of posterior analyses as presented in this paper, and thus would be aided by improvements to sampler efficiency, either through additional parallelization or speedups in model simulation

and summary statistic calculation.

7 Conclusion

We have presented two sequential Monte Carlo samplers, originally proposed for approximate Bayesian inference, that have been adapted to perform exact inference by replacing the kernel function. While both samplers demonstrated sensitivity to the number of parameters in the model to be fit, we demonstrated the additional sensitivity of the Toni sampling scheme to the choice of the kernel normalizing constant during probabilistic inference, which limits the appeal of this algorithm for the purposes of performing exact inference. The Del Moral sampler, which does not share this sensitivity due to its use of Metropolis-Hastings update steps, was able to perform Bayesian inference on the O’Hara-Rudy cardiac action potential model, as well as approximate Bayesian inference on the same model using a set of five common summary statistics. Comparison of the posteriors produced under these inference settings allowed us to conclude that the summary statistics chosen contained a large amount of information about the conductances of the fast sodium current, rapid delayed rectifier current, and inward rectifier current, as represented in the model, but were insufficient to constrain parameters controlling many of the background currents or currents primarily active between the the time of peak membrane voltage and substantial membrane repolarization. Future developments on these methods may be focused on including larger bodies of data, as well as improvements to the efficiencies of the samplers in order to facilitate the efficient exploration of large parameter spaces.

We hope that this study serves as an example of an identifiability analysis of cardiac models, as well as other biologically relevant models, that may be performed under both Bayesian and approximate Bayesian schemes prior to gathering experimental data. Performing this type of analysis would allow modelers to assess the sufficiency of experimental recordings to capture information about model parameters and sub-components, and could therefore lead to the design of experiments that constrain the model most effectively before much time and money are expended in labs or clinics. Thus, we believe that the understanding and informed use of sampling techniques as described in this paper can both facilitate and accelerate the production of cardiac models when *in vitro* or *in vivo* conditions place limitations on one’s ability to observe the system.

8 Author Contributions

A.C.D. conceived of the study, participated in the design of the study, carried out all analyses, and drafted the manuscript. C.H. participated in the design of the study, validation of results, and editing of the manuscript. D.J.G. and J.C. participated in the design of the study and editing of the manuscript.

9 Acknowledgments

A.C.D. would like to thank the Rhodes Trust for their funding and support.

10 Data Accessibility

All code and data used in the study can be found (along with instructions for installation and execution of analyses) in the following “paper tutorial” repository: <https://chaste.cs.ox.ac.uk/trac/wiki/PaperTutorials/DalySMC>.

11 Funding

A.C.D. was funded by the Rhodes Trust, UK. J.C. gratefully acknowledges research support from the 2020 Science programme funded through the EP-SRC Cross-Disciplinary Interface Programme (EP/I017909/1).

References

- [1] M.A. Beaumont, W. Zhang, and D.J. Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.
- [2] M.G.B. Blum, M.A. Nunes, D. Prangle, and S.A. Sisson. A comparative review of dimension reduction methods in approximate Bayesian computation. *Stat. Sci.*, 28(2):189–208, 2013.
- [3] M. Burger. Inverse problems in ion channel modeling. *Inverse Probl.*, 27:1–34, 2011.
- [4] R.J. Carroll, D. Ruppert, L.A. Stefanski, and C.M. Crainiceanu. *Measurement error in nonlinear models: a modern perspective*. Monographs on Statistics and Applied Probability. Chapman and Hall, CRC, Boca Raton, FL, 2 edition, 2006.

- [5] N. Chopin. A sequential particle filter method for static models. *Biometrika*, 89(3):539–551, 2002.
- [6] C. Cobelli and J.J. Di Stefano. Parameter and structural identifiability concepts and ambiguities: a critical review. *Am. J. Physiol.*, 1980.
- [7] J. Cooper, G.R. Mirams, and S.A. Niederer. High-throughput functional curation of cellular electrophysiology models. *Prog. Biophys. Mol. Bio.*, 107(1):11–20, 2011.
- [8] C. Corrado, J. Whitaker, H. Chubb, S. Williams, M. Wright, , J. Gill, M. O’Neill, and S.A. Niederer. Personalized models of human atrial electrophysiology derived from endocardial electrograms. *IEEE T. Bio-Med. Eng.*, 2016.
- [9] C. Corrado, S. Williams, H. Chubb, M. O’Neill, and S.A. Niederer. *Personalization of Atrial Electrophysiology Models from Decapolar Catheter Measurements*, pages 21–28. Springer International Publishing, Cham, 2015.
- [10] D. Csercsik, K.M. Hangos, and G. Szederkenyi. Identifiability analysis and parameter estimation of a single Hodgkin-Huxley type voltage dependent ion channel under voltage step measurement conditions. *Neurocomputing*, 77(1):178–188, 2012.
- [11] A.C. Daly, D.J. Gavaghan, C. Holmes, and J. Cooper. Hodgkin-huxley revisited: reparameterization and identifiability analysis of the classic action potential model with approximate Bayesian computation. *J. R. Soc. Open Sci.*, 2(12), 2015.
- [12] C.E. Dangerfield, D. Kay, and K. Burrage. Modeling ion channel dynamics through reflected stochastic differential equations. *Phys. Rev. E*, 85(5), 2012.
- [13] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *J. R. Statist. Soc. B*, 68(3):411–436, 2006.
- [14] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. An adaptive sequential monte carlo method for approximate bayesian computation. *Stat. Comput.*, 22(5):1009–1020, 2012.
- [15] M. Fink and D. Noble. Markov models for ion channels: versatility versus identifiability and speed. *Phil. Trans. R. Soc. A*, 367:2161–2179, 2009.

- [16] A.L. Hodgkin and A.F. Huxley. A quantitative description of membrane current and its application to conductance and excitation in nerve. *J. Physiol.*, 117(4):500–544, 1952.
- [17] R.H. Johnstone, E.T.Y. Chang, R. Bardenet, T.P. de Boer, D.J. Gavaghan, P. Pathmanathan, R.H. Clayton, and G.R. Mirams. Uncertainty and variability in models of the cardiac action potential: can we build trustworthy models? *J. Mol. Cell. Cardiol.*, 96:49–62, 2016.
- [18] C.M. Lloyd, J.R. Lawson, P.J. Hunter, and P.F. Nielsen. The CellML model repository. *Bioinformatics*, 24(18):2122–3, 2008.
- [19] W.E. Louch, S. Land, S. Niederer, and S. Alexander. Strange bedfellows: biologists and mathematical modelers tie the knot on cardiomyocyte calcium homeostasis. *Drug Discov. Today*, 14:11–16, 2014.
- [20] J.M. Marin, P. Pudlo, C.P. Robert, and R.J. Ryder. Approximate Bayesian computational methods. *Stat. Comput.*, 22(6):1167–1180, 2011.
- [21] M.M. McKerns and M. Aivazis. pathos: a framework for heterogeneous computing, 2010-. <http://trac.mystic.cacr.caltech.edu/project/pathos>.
- [22] M.M. McKerns, L. Strand, T. Sullivan, A. Fang, and M.A.G. Aivazis. Building a framework for predictive science. In *Proceedings of the 10th Python in Science Conference*, 2011.
- [23] G.R. Mirams, C.J. Arthurs, M.O. Bernabeau, R.F. Bordas, J. Cooper, A. Corrias, Y. Davit, S.J. Dunn, A.G. Fletcher, D.G. Harvey, M.E. Marsh, J.M. Osborne, P. Pathmanathan, and D.J. Gavaghan. Chaste: an open-source C++ library for computational physiology and biology. *PLoS Comput. Biol.*, 9(3), 2013.
- [24] E. Neher and B. Sakmann. The patch clamp technique. *Sci. Am.*, 266(3):44–51, 1992.
- [25] S. Niederer, S. Land, S.W. Omholt, and N.P. Smith. Interpreting genetic effects through models of cardiac electromechanics. *Am. J. Physiol.*, 303(11):H1294–H1303, 2012.
- [26] S. Niederer, L. Mitchell, N. Smith, and G. Plank. Simulating human cardiac electrophysiology on clinical time scales. *Front. Physiol.*, 2(14), 2011.

- [27] S.A. Niederer, M. Fink, D. Noble, and N.P. Smith. A meta-analysis of cardiac electrophysiology models. *Exp. Physiol.*, 64(5):486–495, 2009.
- [28] S.A. Niederer and N.P. Smith. At the heart of computational modeling. *J. Physiol.*, 590(6):1331–1338, 2012.
- [29] D. Noble and Y. Rudy. Models of cardiac ventricular action potentials: iterative interactions between experiment and simulation. *Phil. Trans. R. Soc. Lond. A*, 359(1793):1127–1142, 2001.
- [30] T. O’Hara, L. Virag, A. Varro, and Y. Rudy. Simulation of the undiseased human cardiac ventricular action potential: model formulation and experimental validation. *PLoS Comput. Biol.*, 7(5), 2011.
- [31] A. Patil, D. Huard, and C.J. Fonnesbeck. PyMC: Bayesian stochastic modeling in Python. *J. Stat. Soft.*, 35(4):1–81, 2010.
- [32] D. Prangle. Summary statistics in approximate Bayesian computation. 2015.
- [33] P. Pudlo, J.M. Marin, A. Estoup, J.M. Cornuet, M. Gauthier, and C.P. Robert. Reliable ABC model choice via random forests. 2015.
- [34] A. Raue, C. Kreutz, T. Maiwald, U. Klingmuller, and J. Timmer. Addressing parameter identifiability by model-based experimentation. *IET Syst. Biol.*, 4(2):120–130, 2011.
- [35] Y. Rudy and J.R. Silva. Computational biology in the study of cardiac ion channels and cell electrophysiology. *Q. Rev. Biophys.*, 39(1):57–116, 2006.
- [36] I. Siekmann, J. Sneyd, and E. Crampin. MCMC can detect nonidentifiable models. *Biophys. J.*, 103(11):2275–2286, 2012.
- [37] S.A. Sisson, Y. Fan, and M. Tanaka. Sequential Monte Carlo without likelihoods. *Proc. Natl. Acad. Sci.*, 2007.
- [38] S.A. Sisson, Y. Fan, and M. Tanaka. Sequential Monte Carlo without likelihoods: Errata. *Proc. Natl. Acad. Sci.*, 2009.
- [39] N. Smith and S. Niederer. Is computational modeling adding value for understanding the heart? *J. Mol. Cell. Cardiol.*, 96:1, 2016.

- [40] M. Sunnaker, A.G. Busetto, E. Numminen, J.C. Corander, M. Foll, and C. Dessimoz. Approximate Bayesian computation. *PLoS Comput. Biol.*, 9(1), 2013.
- [41] T. Toni, D. Welch, N. Strelkowa, A. Ipsen, and M. Stumpf. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface*, 6(31):187–202, 2009.
- [42] D.J. Wilkinson. Bayesian methods in bioinformatics and computational systems biology. *Brief Bioinform.*, 8(2):109–116, 2007.
- [43] D.J. Wilkinson. *Stochastic modelling for systems biology*. Math. Comput. Biol. Chapman and Hall, CRC, London, UK, 2 edition, 2011.
- [44] R.D. Wilkinson. Approximate Bayesian computation (ABC) gives exact results under the assumption of model error. *Stat. App. Gen. Mol. Bio.*, 12:129–141, 2013.