

Approximate Bayesian computation for stochastic hybrid systems with ergodic behaviour

Sascha Desmettre*, Agnes Mallinger†, Amira Meddah‡, Irene Tubikanec§

Abstract

Piecewise diffusion Markov processes (PDifMPs) form a versatile class of stochastic hybrid systems that combine continuous diffusion processes with discrete event-driven dynamics, enabling flexible modelling of complex real-world hybrid phenomena. The practical utility of PDifMP models, however, depends critically on accurate estimation of their underlying parameters. In this work, we present a novel framework for parameter inference in PDifMPs based on approximate Bayesian computation (ABC). Our contributions are threefold. First, we provide detailed simulation algorithms for PDifMP sample paths. Second, we extend existing ABC summary statistics for diffusion processes to account for the hybrid nature of PDifMPs, showing particular effectiveness for ergodic systems. Third, we demonstrate our approach on several representative example PDifMPs that empirically exhibit ergodic behaviour. Our results show that the proposed ABC method reliably recovers model parameters across all examples, even in challenging scenarios where only partial information on jumps and diffusion is available or when parameters appear in state-dependent jump rate functions. These findings highlight the potential of ABC as a practical tool for inference in various complex stochastic hybrid systems.

Keywords

Simulation-based inference, Approximate Bayesian computation, Parameter estimation, Stochastic hybrid systems, Piecewise diffusion Markov processes.

MSC Subject classifications

60H35, 62F15, 93C30

Acknowledgements

We thank JKU Linz, in particular the vice rectorates for Finance and Entrepreneurship and for Research and International Affairs, for supporting this work with means of their 8th Invest Call.

1 Introduction

Combining continuous-time and event-driven dynamics, stochastic hybrid systems (SHS) have emerged as a powerful modelling framework over the past decades [11, 12, 15]. They have been

*Institute of Financial Mathematics and Applied Number Theory, Johannes Kepler University Linz (sascha.desmettre@jku.at)

†Institute of Financial Mathematics and Applied Number Theory and Institute of Applied Statistics, Johannes Kepler University Linz (agnes.mallinger@jku.at)

‡Institute of Stochastics, Johannes Kepler University Linz (amira.meddah@jku.at)

§Institute of Applied Statistics, Johannes Kepler University Linz (irene.tubikanec@jku.at)

applied across a wide range of fields, ranging from mathematical biology [3, 8, 9, 13, 28] and neuroscience [6, 29] to biochemistry [34] and mathematical finance [5, 23].

A prominent subclass of SHS is formed by piecewise deterministic Markov processes (PDMPs), which combine deterministic continuous dynamics – typically modelled by ordinary differential equations (ODEs) – with stochastic jump dynamics. PDMPs were first introduced by Davis in 1984 [15]. A few years later, Blom [4] proposed a more general model in which the continuous dynamics are governed not by ODEs but by stochastic differential equations (SDEs), and state-dependent reset maps were introduced, leading to the formulation of piecewise diffusion Markov processes (PDifMPs). The theoretical foundations of these models were subsequently established by Bujorianu and Lygeros in 2003 [10]. Notably, jump-diffusion models [32] form a related class of systems that also combine SDE dynamics with random jumps. However, PDifMPs provide a more flexible modelling framework by allowing for state-dependent regime changes, thereby capturing a broader range of hybrid stochastic behaviours.

From a computational perspective, simulating PDifMPs remains challenging due to the interaction between stochastic diffusion dynamics and discrete jumps. Dedicated numerical methods have recently been proposed and shown to converge to such hybrid systems [9], and exact schemes for first-passage times have been developed to capture threshold-dependent events [17, 18]. Despite these advances, suitable statistical inference methods for PDifMPs, which enable parameter estimation, remain largely uninvestigated in the literature.

A PDifMP may depend on several parameters that govern both its SDE and jump process dynamics. In this work, we focus on parameter estimation using approximate Bayesian computation (ABC), a simulation-based approach for Bayesian parameter inference that is particularly useful when the underlying likelihood function is unknown or intractable. Over the past decades, ABC has become one of the most widely used and well-established tools for inference in complex and realistic mathematical models (see [27, 35] for reviews). Originally introduced in the context of population genetics [2], it has since been successfully applied to a wide range of problems arising in diverse application areas. More recently, ABC has been employed for parameter inference in SDE models [24, 25, 30], particularly for ergodic SDEs [7, 19, 33], and has also received limited attention in the context of jump-diffusion models [14, 22]. To our knowledge, ABC has not yet been applied to parameter inference in PDifMPs.

Adapting the ABC method for inference in PDifMPs involves two major challenges. First, it requires efficient simulation of synthetic datasets (sample paths) of the PDifMP. We address this by proposing detailed algorithms for path simulation: one for constant jump rate functions and another for state-dependent jump rates. Second, the quality of ABC strongly depends on informative summary statistics. To this end, we extend the summary statistics proposed in [7] for SDE models to PDifMPs, taking into account their hybrid nature. These statistics are particularly effective for PDifMPs with ergodic behaviour, as ergodicity allows relevant information to be extracted from individual datasets rather than requiring repeated simulations for each parameter configuration.

We assess the performance of our inference method on four representative PDifMP examples that empirically exhibit ergodic behaviour. Our approach achieves strong inference results across all examples, including challenging scenarios where only the number of jumps (rather than their exact timing) is observed, the SDE component is only partially observed, and/or parameters appear in a state-dependent jump rate function. These results demonstrate the potential of ABC, and in particular our proposed method, for inference in PDifMPs.

This paper is organised as follows. In Section 2, we recall the general PDifMP framework and introduce detailed algorithms for simulating PDifMP sample paths. In Section 3, we present several representative example PDifMPs and demonstrate empirically that they exhibit ergodic behaviour. In Section 4, we review the ABC method and describe its adaptation for inference in PDifMP models with ergodic behaviour. The corresponding inference results are reported in Section 5. Finally, Section 6 concludes with a summary and an outlook on future research directions. Accompanying R code is provided on GitHub (cf. Section 4.2).

2 Piecewise diffusion Markov processes

This section discusses PDifMPs. In Section 2.1, we recall the general framework of PDifMPs. In Section 2.2, we describe in detail how they can be simulated. In Section 3, we introduce a diverse set of representative model problems from the class of PDifMPs with ergodic behaviour, which serve as the basis for the analyses in the subsequent sections.

2.1 General framework of PDifMPs

Let $T > 0$ and $m \in \mathbb{N}$. Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a complete probability space with a complete and right-continuous filtration $(\mathcal{F}_t)_{t \in [0, T]}$. Moreover, let $(W_t)_{t \in [0, T]}$ be an m -dimensional standard Wiener process on $(\Omega, \mathcal{F}, \mathbb{P})$ and adapted to $(\mathcal{F}_t)_{t \in [0, T]}$.

A PDifMP is a stochastic process $U = (U_t)_{t \in [0, T]}$ adapted to $(\mathcal{F}_t)_{t \in [0, T]}$ consisting of two possibly interacting components, i.e. $U_t := (X_t, Z_t)$ for all $t \in [0, T]$. Its state space is given by $E = D_1 \times D_2$ with $D_1 \subseteq \mathbb{R}^{d_1}$ and $D_2 \subseteq \mathbb{R}^{d_2}$, for $d_1, d_2 \in \mathbb{N}$. The Borel σ -algebra $\mathcal{B}(E)$ is generated by the topology of E .

Description of PDifMP components The component $Z = (Z_t)_{t \in [0, T]}$ is a jump process with right-continuous, piecewise-constant paths and takes values in D_2 . Specifically, the (random) jump times $(J_k)_{k \in \{0, \dots, N^j-1\}}$, where $N^j \in \mathbb{N}$ is the number of jumps occurring within $[0, T]$, are governed by a rate function Λ , which may depend on both components of the PDifMP. On each interjump interval $[J_k, J_{k+1})$, the process Z_t remains constant, i.e., $Z_t \equiv z_k$, for all $t \in [J_k, J_{k+1})$, where $z_k \in D_2$.

The component $X = (X_t)_{t \in [0, T]}$ is a diffusion process with continuous paths and takes values in D_1 . More precisely, on each interjump interval $[J_k, J_{k+1})$, it follows the dynamics of the SDE

$$\begin{aligned} dX_t &= F(X_t, z_k)dt + \Sigma(X_t, z_k)dW_t, \quad t \in [J_k, J_{k+1}), \\ X_{J_k} &= x_{J_k}, \quad x_{J_k} \in D_1, \end{aligned} \tag{1}$$

where the drift coefficient $F : E \rightarrow \mathbb{R}^{d_1}$ and the diffusion coefficient $\Sigma : E \rightarrow \mathbb{R}^{d_1 \times m}$ depend on the corresponding value z_k of the jump process Z_t .

Denote by ϕ the solution (flow) of SDE (1), i.e.

$$X_t = \phi(t - J_k, u_k), \quad t \in [J_k, J_{k+1}),$$

where $u_k := (x_{J_k}, z_k)$. The next initial condition $X_{J_{k+1}} = x_{J_{k+1}}$ for the SDE over the subsequent interjump interval $[J_{k+1}, J_{k+2})$ is obtained by $\phi(J_{k+1} - J_k, u_k)$. This means that although ϕ is defined only on the interval $[J_k, J_{k+1})$, its right-limit value at J_{k+1} is used to define the next initial condition, thereby ensuring the continuity of the paths for the process X_t . Moreover, the next value z_{k+1} of the jump process Z_t is drawn according to a transition kernel $Q(x_{J_{k+1}}, z_k, \cdot)$, which may depend on both components X_t and Z_t of the PDifMP.

Note that we define the initial jump time as $J_0 := 0$ and the initial condition $u_0 = (x_0, z_0)$ is a given value from E .

Characteristic triple A PDifMP can be defined by its characteristic triple (ϕ, Λ, Q) :

- $\phi : [0, T] \times E \rightarrow D_1$ is the flow associated with the piecewise-defined SDE (1).
- $\Lambda : E \rightarrow \mathbb{R}_+$ is a non-negative measurable map, that defines the jump rate of the PDifMP.
- $Q : E \times \mathcal{B}(D_2) \rightarrow [0, 1]$ is a transition kernel that assigns to each $u = (x, z) \in E$ a probability measure $Q(u, \cdot)$ on $(D_2, \mathcal{B}(D_2))$, describing the distribution of the post-jump state of Z .

We refer the reader to Section 3 for concrete choices of (ϕ, Λ, Q) and to Remark 4 in that section for possible extensions.

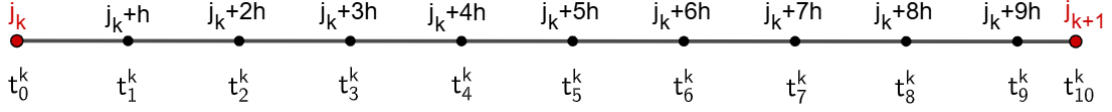


Figure 1: Example time grid for $N_k = 9$.

Algorithm 1 SDE: path simulation

Input: Flow ϕ , jump times j_k and j_{k+1} , step size h , initial value x_{j_k} , constant z_k

Output: Path of SDE between two consecutive jumps

- 1: Set $x[0] = x_{j_k}$
 - 2: Determine $\tau = j_{k+1} - j_k$
 - 3: Determine $N = \lfloor \tau/h \rfloor$
 - 4: Determinate $lastStepSize = \tau - Nh$
 - 5: **for** $n=0:(N-1)$ **do**
 - 6: Set $x[n+1] = \phi(h, x[n], z_k)$
 - 7: **end for**
 - 8: Set $x[N+1] = \phi(lastStepSize, x[N], z_k)$
 - 9: **return** $x[1 : (N+1)]$
-

Assumptions We assume the following regularity conditions.

1. The drift F and the diffusion coefficient Σ are such that there exists a unique strong solution for SDE (1). (Standard conditions are global Lipschitz continuity and linear growth [37].)
2. The jump rate function Λ is such that the probability of infinitely many jumps occurring within the finite time interval $[0, T]$ is zero.
3. The jump rate function Λ is such that the probability that at least one jump occurs within $[0, T]$ is positive.

2.2 Simulation of PDifMPs

In this section, we detail algorithms for the simulation of paths of PDifMPs.

Simulation of SDE paths over interjump intervals A key step for the simulation of PDifMP paths is the simulation of paths of the solution of SDE (1) over a given sample interjump interval $[j_k, j_{k+1}]$, where j_k and j_{k+1} are realisations of J_k and J_{k+1} , respectively. This requires to discretise the considered time interval $[j_k, j_{k+1}]$ including the end point j_{k+1} with step size $h > 0$. Since the length $\tau = j_{k+1} - j_k$ might not be a multiple of h , the size of the last step can be smaller than h . Specifically, we obtain the time grid

$$t_n^k = j_k + nh, \quad n = 0, \dots, N_k, \quad \text{where } N_k = \left\lfloor \frac{j_{k+1} - j_k}{h} \right\rfloor,$$

with final time point $t_{N_k+1}^k := j_{k+1}$, i.e. the size of the last step is $\tau - N_k h$ and $\lfloor \cdot \rfloor$ denotes the integer floor function. Figure 1 reports an example time grid for $N_k = 9$. Algorithm 1 summarises the simulation procedure of an SDE path over a general time grid.

Remark 1 Note that Algorithm 1 is called iteratively within Algorithms 2 and 3 to generate PDifMP paths over $[0, T]$. Accordingly, the algorithm includes the value $x[N+1] = x_{j_{k+1}}$ in the returned path so that it serves as the initial condition for the subsequent interval. In contrast, the initial value $x[0] = x_{j_k}$ is not returned, since it coincides with the last value of the preceding interval and is therefore already available.

Algorithm 2 PDifMP with constant jump rate: path and jump time simulation

Input: Flow ϕ , transition kernel Q , jump rate λ , time horizon T , initial values x_0, z_0 and step size h

Output: Path of PDifMP

- 1: Set $x[0] = x_0, z[0] = z_0, j[0] = 0$ and $k = 0$.
 - 2: Sample τ from the exponential distribution with rate λ .
 - 3: Set $j[k+1] = j[k] + \tau$.
 - 4: **while** $j[k+1] < T$ **do**
 - 5: Set x_{last} to the last value of x .
 - 6: Simulate a path of the SDE flow ϕ between $j[k]$ and $j[k+1]$ using Algorithm 1 with step size h , initial value x_{last} and constant $z[k]$, and append it to x .
 - 7: Determine $z[k+1]$ from the transition kernel Q .
 - 8: Sample τ from the exponential distribution with rate λ .
 - 9: Set $k = k + 1$.
 - 10: Set $j[k+1] = j[k] + \tau$.
 - 11: **end while**
 - 12: Set $j[k+1] = T$.
 - 13: Set x_{last} to the last value of x .
 - 14: Simulate a path of the SDE flow between $j[k]$ and $j[k+1]$ with Algorithm 1 with step size h , initial value x_{last} and constant $z[k]$, and append it to x .
 - 15: Draw $z[k+1]$ from the transition kernel Q .
 - 16: **return** x, z and j .
-

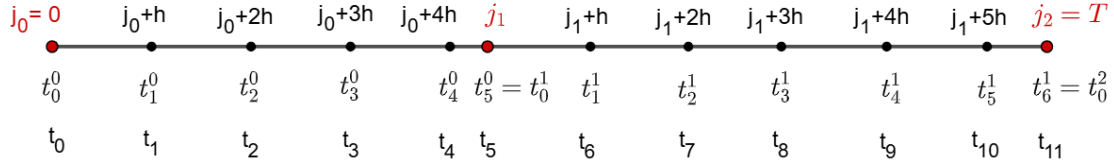


Figure 2: Example time grid for $N^j = 3$ and $N^x = 12$.

Simulation of PDifMP paths with constant jump rate For simplicity, we first consider a constant jump rate function $\Lambda \equiv \lambda > 0$. The construction of a PDifMP path then works according to the following steps.

Fix the initial jump time $j_0 = 0$ and the initial condition $u_0 = (x_0, z_0) \in E$, and set $k = 0$. Simulate the next jump time by generating a waiting time τ from the exponential distribution with rate λ , and set $j_{k+1} = j_k + \tau$. Given $u_k = (x_{j_k}, z_k)$, generate a path of the SDE solution

$$X_t = \phi(t - j_k, u_k), \quad t \in [j_k, j_{k+1}],$$

cf. Algorithm 1. Then, draw the next constant z_{k+1} from the transition kernel Q and set $k = k + 1$. Repeat this procedure until a jump time $j_{k+1} \geq T$ is generated and set $j_{k+1} = T$. These steps are summarised in Algorithm 2.

Simulation of PDifMP paths with process-dependent jump rate function In this case, the diffusion dynamics remain unchanged, while the jump times can no longer be drawn directly from an exponential distribution and should instead be simulated using the *thinning method* [9, 26], which adapts the classical approach for non-homogeneous Poisson processes to state-dependent intensities. We consider non-constant jump rate functions Λ , which may depend on the states of the process $U_t = (X_t, Z_t)$ and are bounded by a constant $\lambda > 0$. In what follows, we restrict attention to the case where Λ depends only on the continuous component, i.e. $\Lambda(U_t) := \Lambda(X_t)$.

Algorithm 3 PDifMP with state-dependent and bounded jump rate function

Input: Flow ϕ , transition kernel Q , jump rate Λ , upper bound λ of Λ , time horizon T , initial values x_0, z_0 and step size h

Output: PDifMP

```
1: Set  $x[0] = x_0, z[0] = z_0, j[0] = 0, k = 0$ .
2: Sample  $\tau$  from the exponential distribution with rate  $\lambda$ .
3: Set  $j_{old} = j[k]$ .
4: Set  $j_{new} = j[k] + \tau$ .
5: while  $j_{new} < T$  do
6:   Set  $x_{last}$  to the last value of  $x$ 
7:   Simulate a path of the SDE flow  $\phi$  between  $j_{old}$  and  $j_{new}$  using Algorithm 1 with step size  $h$ , initial value  $x_{last}$  and constant  $z[k]$ , and append it to  $x$ .
8:   Set  $accept = \text{true}$  with probability  $\Lambda(x_{last}, z[k])/\lambda$ 
9:   if accept then
10:    Draw  $z[k+1]$  from the transition kernel.
11:    Set  $j[k+1] = j_{new}$ .
12:    Set  $k = k + 1$ .
13:   end if
14:   Sample  $\tau$  from the exponential distribution with rate  $\lambda$ .
15:   Set  $j_{old} = j_{new}$ .
16:   Set  $j_{new} = j_{old} + \tau$ .
17: end while
18: Set  $x_{last}$  to the last value of  $x$ 
19: Simulate a path of the SDE flow  $\phi$  between  $j_{old}$  and  $T$  using Algorithm 1 with step size  $h$ , initial value  $x_{last}$  and constant  $z[k]$ , and append it to  $x$ .
20: Set  $j[k+1] = T$ .
21: Draw  $z[k+1]$  from the transition kernel.
22: Return  $x, z$  and  $j$ .
```

The steps are as follow: Draw a candidate j_{new} for the next jump time from an exponential distribution with rate λ and simulate the path x of X_t up to j_{new} (cf. Algorithm 1). The candidate j_{new} is accepted as a jump time with probability $\Lambda(x_{j_{new}})/\lambda$. If accepted, the new discrete state z_{k+1} is drawn from the transition kernel Q and the index k is incremented by one. Regardless of whether the candidate was accepted or not, we set $j_{old} = j_{new}$, sample a new waiting time τ from the exponential distribution with rate λ and add it to the previous candidate j_{old} in order to obtain the new candidate j_{new} . The whole simulation procedure is reported in Algorithm 3.

Note that the time grid for the path x is not specified explicitly in Algorithms 2 and 3. We denote the points on this grid by $t_i, i = 0, \dots, N^x - 1$, where N^x is the total number of time points. An example configuration with $N^x = 12$ and $N^j = 3$ jumps is shown in Figure 2, which also illustrates how this global grid relates to the piecewise grids reported in Figure 1.

3 Representative example PDifMPs

In this section we introduce diverse example PDifMPs that will serve as test problems for the inference via ABC. Each PDifMP test problem is motivated by an underlying SDE, which governs the dynamics of the component X between two consecutive jumps. A suitable transition kernel is then specified for each test problem to determine the values z_k of the component Z , which strongly influence the dynamics of X . In most analyses in Section 5, we focus on a constant jump rate function $\Lambda \equiv \lambda > 0$. In Section 5.2, however, we extend our analysis to non-constant jump rate functions that depend on the state of the component X .

For the reader's convenience, we first recall the motivating SDEs in Section 3.1 and then use them to construct the corresponding PDifMPs in Section 3.2.

3.1 Underlying SDEs

Wiener process with drift (WPWD) This process is described by the SDE

$$dX_t = \gamma dt + \sigma dW_t, \quad t \in [0, T], \quad X_0 = x_0 \in \mathbb{R}, \quad (2)$$

with drift $\gamma \in \mathbb{R}$ and noise parameter $\sigma > 0$. Its solution (flow) is given by

$$X_t = \phi(t, x_0; \gamma) = x_0 + \gamma t + \sigma W_t.$$

Note that, for each $t \in [0, T]$,

$$X_t \sim \mathcal{N}(x_0 + \gamma t, \sigma^2 t).$$

Therefore, since both the mean and variance of this process change linearly with time, its distribution does not converge to a stationary distribution.

Ornstein-Uhlenbeck process (OU) This process is described by the SDE

$$dX_t = \gamma_1(\gamma_2 - X_t)dt + \sigma dW_t, \quad t \in [0, T], \quad X_0 = x_0 \in \mathbb{R}, \quad (3)$$

where $\gamma_2 \in \mathbb{R}$ and $\gamma_1, \sigma > 0$. This SDE can be solved explicitly. Its solution (flow) is given by

$$X_t = \phi(t, x_0; \gamma_1, \gamma_2) = x_0 e^{-\gamma_1 t} + \gamma_2(1 - e^{-\gamma_1 t}) + \sigma \int_0^t e^{-\gamma_1(t-s)} dW_s.$$

For each $t \in [0, T]$,

$$X_t \sim \mathcal{N}\left(x_0 e^{-\gamma_1 t} + \gamma_2(1 - e^{-\gamma_1 t}), \frac{\sigma^2}{2\gamma_1}(1 - e^{-2\gamma_1 t})\right),$$

where the variance can be determined using Itô's isometry. Therefore, for $t \rightarrow \infty$, the distribution of the OU process converges to a stationary distribution given by

$$\mathcal{N}\left(\gamma_2, \frac{\sigma^2}{2\gamma_1}\right). \quad (4)$$

Weakly damped stochastic harmonic oscillator (WDSHO) This process is two-dimensional, i.e.

$X_t = (X_t^{(1)}, X_t^{(2)})^\top$, $t \in [0, T]$, and is described by the SDE

$$d \begin{pmatrix} X_t^{(1)} \\ X_t^{(2)} \end{pmatrix} = \begin{pmatrix} X_t^{(2)} \\ -\gamma_1^2 X_t^{(1)} - 2\gamma_2 X_t^{(2)} \end{pmatrix} dt + \begin{pmatrix} 0 \\ \sigma \end{pmatrix} dW_t, \quad t \in [0, T], \quad X_0 = x_0 = (x_0^{(1)}, x_0^{(2)})^\top, \quad (5)$$

where $\gamma_1, \gamma_2 > 0$ with $\gamma_1^2 - \gamma_2^2 > 0$ and $\sigma > 0$. The equation is a linear SDE with additive noise and can be solved explicitly. Specifically, it can be written as

$$dX_t = A(\gamma_1, \gamma_2)X_t dt + \Sigma dW_t, \quad t \in [0, T], \quad X_0 = x_0 = (x_0^{(1)}, x_0^{(2)})^\top,$$

where

$$A(\gamma_1, \gamma_2) := \begin{pmatrix} 0 & 1 \\ -\gamma_1^2 & -2\gamma_2 \end{pmatrix} \quad \text{and} \quad \Sigma := \begin{pmatrix} 0 \\ \sigma \end{pmatrix}, \quad (6)$$

with solution (flow) given by

$$X_t = \phi(t, x_0; \gamma_1, \gamma_2) = e^{A(\gamma_1, \gamma_2)t} x_0 + \int_0^t e^{A(\gamma_1, \gamma_2)(t-s)} \Sigma dW_s.$$

Therefore, for each $t \in [0, T]$,

$$X_t \sim \mathcal{N}\left(\text{Exp}(t; \gamma_1, \gamma_2)x_0, \text{Cov}(t; \gamma_1, \gamma_2)\right),$$

where

$$\text{Exp}(t; \gamma_1, \gamma_2) = e^{A(\gamma_1, \gamma_2)t} = e^{-\gamma_2 t} \begin{pmatrix} \cos(\kappa t) + \frac{\gamma_2}{\kappa} \sin(\kappa t) & \frac{1}{\kappa} \sin(\kappa t) \\ -\frac{\gamma_1}{\kappa} \sin(\kappa t) & \cos(\kappa t) - \frac{\gamma_2}{\kappa} \sin(\kappa t) \end{pmatrix},$$

with $\kappa := \sqrt{\gamma_1^2 - \gamma_2^2} > 0$, and

$$\text{Cov}(t; \gamma_1, \gamma_2) = \int_0^t e^{A(\gamma_1, \gamma_2)(t-s)} \Sigma \Sigma^\top (e^{A(\gamma_1, \gamma_2)(t-s)})^\top ds, \quad (7)$$

with

$$\begin{aligned} \text{Cov}(t; \gamma_1, \gamma_2)_{11} &= \frac{\sigma^2}{4\gamma_2\gamma_1^2} - \frac{\sigma^2 e^{-2\gamma_2 t}}{4\gamma_2\gamma_1^2\kappa^2} (\gamma_1^2 - \gamma_2^2 \cos(2\kappa t) + \gamma_2\kappa \sin(2\kappa t)), \\ \text{Cov}(t; \gamma_1, \gamma_2)_{12} &= \text{Cov}(t; \gamma_1, \gamma_2)_{21} = \frac{\sigma^2}{2\kappa^2} e^{-2\gamma_2 t} \sin^2(\kappa t), \\ \text{Cov}(t; \gamma_1, \gamma_2)_{22} &= \frac{\sigma^2}{4\gamma_2} - \frac{\sigma^2 e^{-2\gamma_2 t}}{4\gamma_2\kappa^2} (\gamma_1^2 - \gamma_2^2 \cos(2\kappa t) - \gamma_2\kappa \sin(2\kappa t)). \end{aligned}$$

It follows that, for $t \rightarrow \infty$, the distribution of this process converges to a stationary distribution given by

$$\mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \frac{\sigma^2}{4\gamma_2\gamma_1^2} & 0 \\ 0 & \frac{\sigma^2}{4\gamma_2} \end{pmatrix}\right).$$

Thus, the value of the parameter γ_2 influences the variance of the stationary distribution of both components of the process. In contrast, the value of γ_1 only influences the variance of the stationary distribution of the first component. Moreover, γ_1 impacts the oscillation frequencies of both components. In particular, larger values of γ_1 correspond to smaller variance and higher frequency in the first component, and to higher frequency in the second component.

Simple stochastic harmonic oscillator (SimpleSHO) This process corresponds to a stochastic harmonic oscillator with $\gamma_2 \equiv 0$. Specifically, it is described by SDE (5) with γ_2 set to zero, which can be written as

$$dX_t = A(\gamma_1, 0)X_t dt + \Sigma dW_t, \quad t \in [0, T], \quad X_0 = x_0 = (x_0^{(1)}, x_0^{(2)})^\top,$$

where A and Σ are as in (6). Its solution (flow) is given by

$$X_t = \phi(t, x_0; \gamma_1) = e^{A(\gamma_1, 0)t} x_0 + \int_0^t e^{A(\gamma_1, 0)(t-s)} \Sigma dW_s.$$

Thus, for each $t \in [0, T]$,

$$X_t \sim \mathcal{N}\left(\text{Exp}(t; \gamma_1)x_0, \text{Cov}(t; \gamma_1)\right),$$

where

$$\text{Exp}(t; \gamma_1) = e^{A(\gamma_1, 0)t} = \begin{pmatrix} \cos(\gamma_1 t) & \frac{1}{\gamma_1} \sin(\gamma_1 t) \\ -\gamma_1 \sin(\gamma_1 t) & \cos(\gamma_1 t) \end{pmatrix},$$

and

$$\text{Cov}(t; \gamma_1) = \int_0^t e^{A(\gamma_1, 0)(t-s)} \Sigma \Sigma^\top (e^{A(\gamma_1, 0)(t-s)})^\top ds = \frac{\sigma^2}{2} \begin{pmatrix} \frac{2\gamma_1 t - \sin(2\gamma_1 t)}{2\gamma_1^3} & \frac{\sin^2(\gamma_1 t)}{\gamma_1^2} \\ \frac{\sin^2(\gamma_1 t)}{\gamma_1^2} & \frac{2\gamma_1 t + \sin(2\gamma_1 t)}{2\gamma_1} \end{pmatrix}. \quad (8)$$

Thus, unlike the WDSHO, the SimpleSHO does not converge to a stationary distribution.

3.2 PDifMP test problems

We now introduce four representative PDifMP test problems (TPs), each constructed by extending one of the SDEs from Section 3.1 into a hybrid stochastic framework. In all cases, a deterministic model parameter of the underlying SDE, typically γ , γ_1 , or γ_2 , is replaced by a piecewise-constant jump process Z , which evolves at random jump times. This construction couples the underlying continuous dynamics of X with the discrete switching behaviour of Z .

Remark 2 Recall that, in general, $Q : E \times \mathcal{B}(D_2) \rightarrow [0, 1]$ denotes a transition kernel assigning to each $u = (x, z) \in E$ a probability measure $Q(u, \cdot)$ on $(D_2, \mathcal{B}(D_2))$. In all test problems below, Q is deterministic and thus reduces to a measurable mapping $Q : E \rightarrow D_2$, assigning a single post-jump value z_{k+1} rather than a distribution. We therefore write $z_{k+1} = Q(x_{J_{k+1}})$ or $z_{k+1} = Q(z_k)$, depending on the case considered.

TP 1 (OU-PDifMP) The idea of the example PDifMP, which is based on the Ornstein-Uhlenbeck process (3), is to let the mean γ_2 of the stationary distribution (cf. (4)) change at random jump times, i.e. γ_2 is replaced by a jump process Z . Specifically, let $b > 0$, then a value of z_k is either equal to b (positive) or equal to $-b$ (negative), depending on the position x_{J_k} of the process X at the jump time J_k . Therefore, the state space of the PDifMP is $E = D_1 \times D_2 = \mathbb{R} \times \{-b, b\}$. In the following, we set $x_0 = 0$ and $z_0 = b$. Moreover, to simplify the notation (in particular, to remove the now unnecessary subscript), we replace γ_1 with η .

Structurally, the OU-PDifMP comprises the following components:

- Underlying SDE:

$$\begin{aligned} dX_t &= \eta(z_k - X_t)dt + \sigma dW_t, \quad t \in [J_k, J_{k+1}), \\ X_{J_k} &= x_{J_k} \in D_1 = \mathbb{R}, \end{aligned}$$

with stochastic flow map given by

$$X_t = \phi(t - J_k, x_{J_k}, z_k) = x_{J_k} e^{-\eta(t-J_k)} + z_k(1 - e^{-\eta(t-J_k)}) + \sigma \int_{J_k}^t e^{-\eta(t-s)} dW_s.$$

- Transition function:

$$z_{k+1} = Q(x_{J_{k+1}}) = \begin{cases} b, & x_{J_{k+1}} \leq 0 \\ -b, & x_{J_{k+1}} > 0 \end{cases},$$

where $x_{J_{k+1}} = \phi(J_{k+1} - J_k, x_{J_k}, z_k)$.

TP 2 (WDSHO-PDifMP) The idea of the example PDifMP, which is based on the weakly damped stochastic harmonic oscillator (5), is to let the frequency of the described oscillations change at the random jump times. Specifically, γ_2 is now denoted by η and γ_1 is replaced by a jump process Z with values z_k that alternate between a fixed number 2 and a parameter $b > \eta$. The state space of this PDifMP is given by $E = D_1 \times D_2 = \mathbb{R}^2 \times \{2, b\}$, the transition mapping for the z_k -values in D_2 is independent from X , and the initial values for the diffusion and jump component are set to $x_0 = (1, 1)^\top$ and $z_0 = b$, respectively.

In particular, the WDSHO-PDifMP comprises the following parts:

- Underlying SDE:

$$\begin{aligned} d \begin{pmatrix} X_t^{(1)} \\ X_t^{(2)} \end{pmatrix} &= \begin{pmatrix} X_t^{(2)} \\ -z_k^2 X_t^{(1)} - 2\eta X_t^{(2)} \end{pmatrix} dt + \begin{pmatrix} 0 \\ \sigma \end{pmatrix} dW_t, \quad t \in [J_k, J_{k+1}), \\ X_{J_k} &= x_{J_k} \in D_1 = \mathbb{R}^2, \end{aligned}$$

with solution

$$X_t = \phi(t - J_k, x_{J_k}, z_k) = e^{A(z_k, \eta)(t - J_k)} x_{J_k} + \int_{J_k}^t e^{A(z_k, \eta)(t - s)} \Sigma \, dW_s,$$

where A and Σ are as in (6).

- Transition function:

$$z_{k+1} = Q(z_k) = \begin{cases} 2, & z_k = b \\ b, & \text{else} \end{cases}.$$

TP 3 (WPWD-PDifMP) The idea of the example PDifMP, which is based on the Wiener process with drift (2), is to let the drift γ change at random jump times, by replacing it with a jump process Z . Specifically, let $b > 0$, then a drift value z_k is either equal to b (positive) or equal to $-b$ (negative), depending on the position x_{J_k} of the process X at the jump time J_k (as for the OU-PDifMP). The state space of this PDifMP is $E = D_1 \times D_2 = \mathbb{R} \times \{-b, b\}$, and the initial values are set to $x_0 = 0$ and $z_0 = b$.

The WPWD-PDifMP can be described by:

- Underlying SDE:

$$\begin{aligned} dX_t &= z_k dt + \sigma dW_t, \quad t \in [J_k, J_{k+1}), \\ X_{J_k} &= x_{J_k} \in D_1 = \mathbb{R}, \end{aligned}$$

with solution

$$X_t = \phi(t - J_k, x_{J_k}, z_k) = x_{J_k} + z_k(t - J_k) + \sigma(W_t - W_{J_k}).$$

- Transition function:

$$z_{k+1} = Q(x_{J_{k+1}}) = \begin{cases} b, & x_{J_{k+1}} \leq 0 \\ -b, & x_{J_{k+1}} > 0 \end{cases},$$

where $x_{J_{k+1}} = \phi(J_{k+1} - J_k, x_{J_k}, z_k)$.

TP 4 (SwitchedSHO-PDifMP) This example PDifMP is based on a combination of both the weakly damped and the simple stochastic harmonic oscillator (cf. Eq. (5)). Specifically, γ_1 is now denoted by η and γ_2 is replaced by a jump process Z with values z_k that alternate between 0 (SimpleSHO case) and a parameter $b \in (0, \eta)$ (WDSHO case). Similar as in TP 2, the transition function is independent of X , the state space is $E = D_1 \times D_2 = \mathbb{R}^2 \times \{0, b\}$, and the initial values are set to $x_0 = (1, 1)^\top$ and $z_0 = b$.

The PDifMP can be described by:

- Underlying SDE:

$$\begin{aligned} d \begin{pmatrix} X_t^{(1)} \\ X_t^{(2)} \end{pmatrix} &= \begin{pmatrix} X_t^{(2)} \\ -\eta^2 X_t^{(1)} - 2z_k X_t^{(2)} \end{pmatrix} dt + \begin{pmatrix} 0 \\ \sigma \end{pmatrix} dW_t, \quad t \in [J_k, J_{k+1}), \\ X_{J_k} &= x_{J_k} \in D_1 = \mathbb{R}^2, \end{aligned}$$

with solution

$$X_t = \phi(t - J_k, x_{J_k}, z_k) = e^{A(\eta, z_k)(t - J_k)} x_{J_k} + \int_{J_k}^t e^{A(\eta, z_k)(t - s)} \Sigma \, dW_s,$$

where A and Σ are as in (6).

- Transition function:

$$z_{k+1} = Q(z_k) = \begin{cases} 0, & z_k = b \\ b, & \text{else} \end{cases}.$$

Remark 3 In TP 1 and TP 3 we allow so-called no-move jumps, i.e. the transition function is defined such that the next value z_{k+1} may equal the previous value z_k . These no-move jumps are not visible in the path of the PDifMP, since no regime change occurs at the corresponding jump times.

Remark 4 Several extensions of the PDifMP framework considered here are possible, although they are beyond the scope of this manuscript. First, note that in all TPs discussed, the underlying SDEs are explicitly solvable, which allows for exact simulation of the corresponding flow ϕ (cf. Algorithm 1). In many practical applications, however, the drift coefficient may be nonlinear, making the SDE unsolvable in closed form. In such cases, appropriate numerical methods can be employed to simulate approximate realisations of the flow ϕ , cf. [9].

Second, in all examples considered, the state space D_2 of the component Z is binary. This setting can be extended to a continuous state space, for instance $D_2 = \mathbb{R}$. In this case, the next value z_{k+1} can be determined through a suitable stochastic transition mechanism, for instance by sampling from a continuous distribution that may depend on the current state of the process, e.g. Gaussian transition kernel centred at z_k , with a variance that may depend on the current position of the process.

Remark 5 Note that, the adaptive time grid of PDifMPs (cf. Figures 1 and 2) may result in very small step sizes, which can introduce numerical inaccuracies in the computation of the covariance matrices (7) and (8). To prevent such issues, we round the jump times in TP 2 and TP 4 to guarantee a minimal step size.

Path of test problems Figure 3 shows a sample path of X over the time interval $[0, 100]$ for each of the four considered PDifMP test problems. The vertical dashed lines indicate the corresponding jump times. For TP 2 (WDSHO-PDifMP) and TP 4 (SwitchedSHO-PDifMP) only the path of the first component $X^{(1)}$ is reported.

The path of TP 1 (OU-PDifMP) illustrates how the component X alternates between the two regimes, with “stationary” means $z_k = b = 2$ and $z_k = -b = -2$, respectively.

The path of TP 2 (WDSHO-PDifMP) also switches between two regimes. In the first regime, it exhibits a high frequency and small variance ($z_k = b = 20$), while in the second regime, it shows a lower frequency and larger variance ($z_k = 2$). In both regimes, the process fluctuates around the underlying stationary mean 0.

The path of TP 3 (WPWD-PDifMP) illustrates how the component X evolves as the drift alternates between $z_k = b = 2$ and $z_k = -b = -2$.

The path of TP 4 (SwitchedSHO-PDifMP) demonstrates how the component X alternates between two types of stochastic harmonic oscillators: a weakly damped oscillator ($z_k = b = 1$) with underlying stationary distribution $\mathcal{N}(0, 1/16)$ and a non-ergodic simple oscillator ($z_k = 0$) whose amplitude, on average, increases over time.

Ergodic behaviour of test problems All considered example PDifMPs exhibit empirically observed ergodic behaviour. For the X component of these processes, this means that time averages computed along a single sufficiently long trajectory coincide with ensemble (phase-space) averages taken over multiple realisations. Formally,

$$\lim_{T \rightarrow +\infty} \frac{1}{T} \int_0^T f(X_s) ds = \int f(x) \mu(dx),$$

i.e., the long-time average of a measurable function f converges to its expectation with respect to the underlying invariant measure μ .

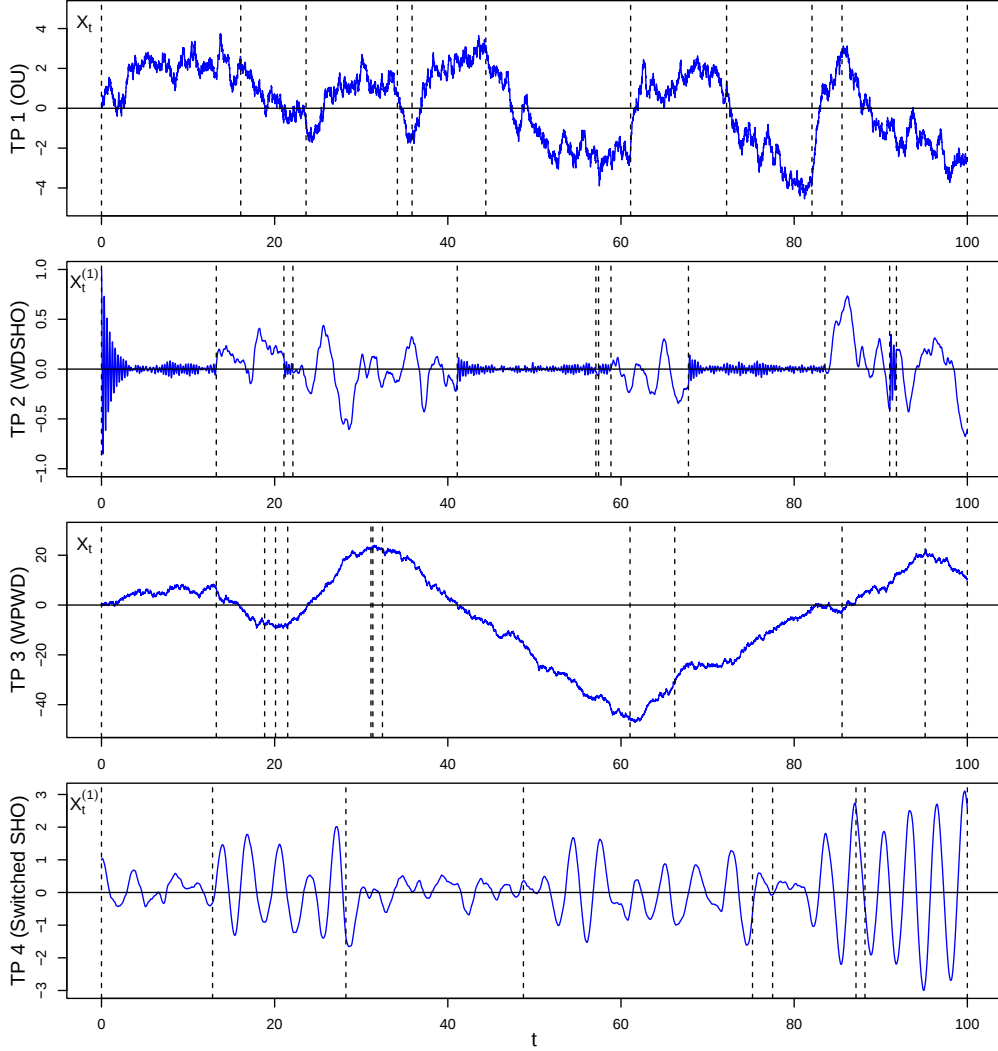


Figure 3: Paths of the four test problems.

This property is crucial for parameter inference in PDifMPs via ABC (cf. Sections 4 and 5), as reliable statistical information can be extracted from a single path, without the need for repeated path simulations under a fixed parameter configuration. Consequently, ergodicity greatly facilitates both the construction and computational efficiency of the proposed ABC inference method.

Figure 4 illustrates this ergodic behaviour for the considered PDifMP test problems, showing that densities derived from a single path (green dashed lines) closely match densities computed from multiple realisations of X_{t^*} , $t^* \gg 0$ (black solid lines). Note that the SDEs underlying the first two test problems are ergodic, so the observed ergodic behaviour of the resulting PDifMPs might be expected. In contrast, the SDEs for the third and fourth test problems are non-ergodic, making the emergence of ergodic behaviour in these cases particularly remarkable.

The density of TP 1 (OU-PDifMP) is bimodal, exhibiting two different means. Similarly, TP 2 (WDSHO-PDifMP) shows two different variances: a tall, narrow central peak (resulting from the smaller variance) and wide, heavy tails (resulting from the larger variance). Moreover, the peaks of the densities for TP 3 (WPWD-PDifMP) are not perfectly aligned, raising some doubts regarding the theoretical ergodicity of that test problem.

The possibility of formally proving ergodicity for the considered test problems is currently under investigation, but lies beyond the scope of this manuscript.

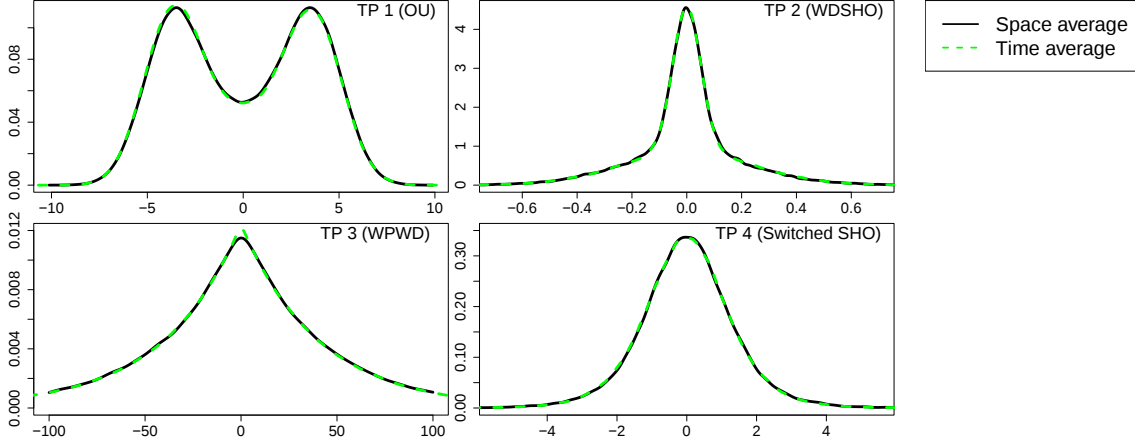


Figure 4: Space and time averages of the four test problems.

4 Approximate Bayesian computation

We now address the problem of parameter inference within PDifMPs. For complex stochastic models such as PDifMPs, the underlying likelihood function is difficult to handle. Even if the underlying SDE of a PDifMP is exactly solvable and its transition density known, constructing a likelihood for the full PDifMP remains unclear. In principle, one could attempt a piecewise formulation if the exact jump times were known, but this is rarely the case in practice. When the underlying SDE is not analytically solvable or the PDifMP is only partially observed, even such piecewise constructions become infeasible. Approximate Bayesian computation (ABC) addresses this challenge by replacing the unavailable likelihood with extensive simulations of synthetic datasets generated from the model. These simulated datasets are then used to construct an approximate posterior distribution for the unknown parameter vector of interest.

Specifically, let y denote the observed dataset, θ the parameter vector to be inferred from y , and $\pi(\theta)$ a prior distribution for θ . The basic ABC algorithm proceeds as follows:

1. Draw a parameter value θ' from the prior distribution $\pi(\theta)$.
2. Conditional on θ' , simulate a synthetic dataset $y_{\theta'}$ from the model.
3. Accept θ' as a sample from the ABC posterior distribution if the distance $d(\cdot, \cdot)$ between a set of summary statistics $s(\cdot)$ of the observed data y and the synthetic dataset $y_{\theta'}$ is below a chosen threshold $\delta \geq 0$, i.e. if $d(s(y), s(y_{\theta'})) < \delta$.

These steps are repeated until a desired number of accepted samples from the ABC posterior distribution has been obtained.

4.1 ABC for PDifMPs

In this section, the ABC methodology is adapted for parameter inference in PDifMPs.

Observed dataset For PDifMPs, the observed dataset y , on which inference will be performed, is defined as

$$y := \{x, N^j\}, \quad (9)$$

where $x = (x_{t_i})_{i=0}^{N^x-1}$ is a time-series dataset consisting of discrete-time measurements of the component X at the times t_i , $i = 0, \dots, N^x - 1$, with $t_0 = 0$ and $t_{N^x-1} = T$, and N^j is the number of jumps that occurred within the observation interval $[0, T]$.

When X has multiple components, as in TP 2 (WDSHO-PDifMP) and TP 4 (SwitchedSHO-PDifMP), we assume that only the first coordinate $X^{(1)}$ is observed. That is, the observed time

series is of the form $x = x^{(1)} = (x_{t_i}^{(1)})_{i=0}^{N^x-1}$. This allows the proposed ABC method to handle challenging partially observed cases, as may be relevant in real-world settings. We note, however, that observing the second component $X^{(2)}$ or both components of X would be possible and would lead to comparable inferential results.

Note that our inference framework does not rely on observing the jump times $j = (j_k)_{k=0}^{N^j-1}$, which are often not available in practice. In some applications, however, the jump times may indeed be observed. To illustrate a case where the exact locations of jump times provide additional information for inference, an example is presented in Appendix A.

Furthermore, it would be possible to include the second jump-related component $z = (z_k)_{k=1}^{N^j}$ in the inference. However, since z might not be observable in practice, we do not use it in our analysis.

Parameter vector The example PDifMPs introduced in the previous section are defined by an underlying SDE, a transition function, and a rate function governing the jump dynamics. Our goal is to simultaneously infer one parameter for each component of the model. Specifically, the SDE is influenced by a diffusion parameter σ , the transition function depends on a parameter b (which in turn affects the SDE dynamics), and the jump rate function is determined by a parameter λ . We aim to infer these three parameters jointly, i.e., to estimate

$$\theta = (\sigma, b, \lambda). \quad (10)$$

Note that, in most of our analyses, we fix η (additional drift parameter of the underlying SDE) to 1/2, 1, and 2 for TP 1 (OU-PDifMP), TP 2 (WDSHO-PDifMP), and TP 4 (SwitchedSHO-PDifMP), respectively. However, in Section 5.3, we include η in the parameter vector θ and demonstrate that this additional parameter can be also inferred without difficulty.

Prior distribution Following [7, 19, 33], we employ uniform prior distributions. In particular, we consider

$$\sigma \sim U(0, 10), \quad b \sim U(0, 10), \quad \lambda \sim U(0, 1), \quad (11)$$

with exceptions for specific test problems. For TP 2 (WDSHO-PDifMP), we choose $b \sim U(2, 100)$, since the model is only well-defined for $b > \eta$, which is set to 1. For TP 4 (SwitchedSHO-PDifMP), we consider $b \sim U(0, 1)$, as the problem is only well-defined for $b \in (0, \eta)$, and η is set to 2. Note that our experiments show that the choice of the prior has limited impact on the inferential results, consistent with the findings of [33] for ABC inference in SDE models.

Model simulation A key step in the ABC methodology is the simulation of synthetic datasets y_θ from the model. For PDifMPs, we employ the two algorithms introduced in Section 2: Algorithm 2 for constant jump rates, and Algorithm 3 for state-dependent bounded jump rate functions. Throughout, we set the step size $h = 10^{-2}$.

Summary statistics Due to the intrinsically stochastic and erratic nature of PDifMPs, simulated paths can vary substantially even when generated under identical parameter configurations. Nevertheless, the empirically observed ergodic behaviour of the PDifMP example problems enables the extraction of informative summary statistics from individual datasets derived from single sample paths, i.e. datasets $y_\theta = \{x_\theta, N_\theta^j\}$.

To capture information about θ from the time-series component x_θ (which is itself influenced by the jump dynamics), we adopt the summary statistics originally proposed in [7] and subsequently used in [19, 33] for ABC inference in SDE models. Specifically, we use x_θ to estimate both the underlying invariant density f_{x_θ} of the X -component of the process and its invariant spectral density S_{x_θ} , via standard kernel density and periodogram estimators, respectively (cf. Section 4.2).

Our analyses further indicate that including also the mean quadratic variation statistic

$$V_{x_\theta} = \frac{1}{N^x} \sum_{i=1}^{N^x} (x_{t_{i-1}} - x_{t_i})^2,$$

improves the estimation of the noise parameter σ .

To capture the hybrid nature of PDifMPs, which combine SDE and jump dynamics, we also include N_θ^j (the number of jumps generated under a given θ) in the proposed set of summaries, without relying on the jump times.

In summary, we define the PDifMP summary statistics of a dataset $y_\theta = \{x_\theta, N_\theta^j\}$ as

$$s(y_\theta) := \{f_{x_\theta}, S_{x_\theta}, V_{x_\theta}, N_\theta^j\}. \quad (12)$$

Distance measures The proposed PDifMP summary statistics (12) comprise both function-valued components (f_{x_θ} and S_{x_θ}) and scalar components (V_{x_θ} and N_θ^j). To quantify the discrepancy between two functions, we define $d_{\text{fun}}(\cdot, \cdot)$ as the sum of absolute pointwise differences. For scalar quantities, we use the absolute difference as the distance measure.

Combining these, we define the overall distance between two sets of summary statistics as

$$d(s(y), s(y_\theta)) = w_1 d_{\text{fun}}(f_y, f_{y_\theta}) + w_2 d_{\text{fun}}(S_y, S_{y_\theta}) + w_3 |V_x - V_{x_\theta}| + w_4 |N^j - N_\theta^j|,$$

where $s(\cdot)$ is as in (12), and w_1, w_2, w_3, w_4 are weights chosen to ensure that the four terms are of comparable magnitude.

To compute these weights, one can, for instance, fix $w_1 = 1$ and generate multiple synthetic datasets y_θ under different parameter values θ drawn from the prior $\pi(\theta)$. Their corresponding distances to the observed dataset y are then computed, and the respective weight is set to the reciprocal of the median of the resulting distances.

4.2 Algorithm and implementation details

The basic ABC algorithm outlined at the beginning of Section 4 is computationally inefficient. This inefficiency arises because candidate parameter values θ' are sampled from the prior distribution, which typically covers much larger parameter regions than the desired posterior distribution. As a result, a substantial amount of computational effort (such as model simulations, summary computations, and distance evaluations) is spent on candidate values θ' , which are ultimately rejected.

To overcome this limitation, sequential ABC algorithms have been developed, with the Sequential Monte Carlo ABC (SMC-ABC) algorithm being the current gold-standard method. In this sequential approach, one initially runs the basic ABC procedure (cf. the beginning of Section 4) until a certain number of parameter values have been accepted as samples from the approximate posterior. In subsequent iterations, new parameter candidates are obtained by resampling from the previously accepted values (according to their associated weights) and perturbing them with an appropriate perturbation kernel. Here, we employ the standard Gaussian perturbation kernel proposed in [21].

This iterative procedure is repeated until a predefined stopping criterion is met, such as reaching a maximum number of synthetic data simulations (maximal computational budget) or a minimal percentage of accepted parameter values (minimal acceptance rate). The optimal stopping criterion may vary depending on the model under consideration. Further details on the SMC-ABC algorithm can be found in [1, 16, 27, 36]. A precise description of the specific algorithm used in this work is provided, e.g., in Algorithm 1 of [33].

The SMC-ABC algorithm is implemented in R [31], with Algorithms 1-3 coded in C++, using the package Rcpp [20]. Parallelisation is achieved using the `foreach` and `doParallel` packages, and the multivariate Gaussian perturbation kernel is implemented using the `mvnfast` package. Following [7], estimates of the invariant density and spectral density are obtained using the R functions `density` and `spectrum`, respectively.

We provide well-documented and user-friendly sample code, including complete algorithmic details and specifications, on GitHub. In particular, code for path simulation of the PDifMP test problems (i.e., implementations of Algorithms 1–3) is available at: https://github.com/AgnesMall/Simulation_of_PDifMPs. Furthermore, code for PDifMP inference via SMC-ABC can be found at: https://github.com/AgnesMall/PDifMP_inference_SMC-ABC.

Setting	σ	b	λ
1	1	2	0.1
2	1	2	0.2
3	2	4	0.2

Table 1: Parameter settings used to simulate the respective observed dataset for TP 1 (OU-PDifMP).

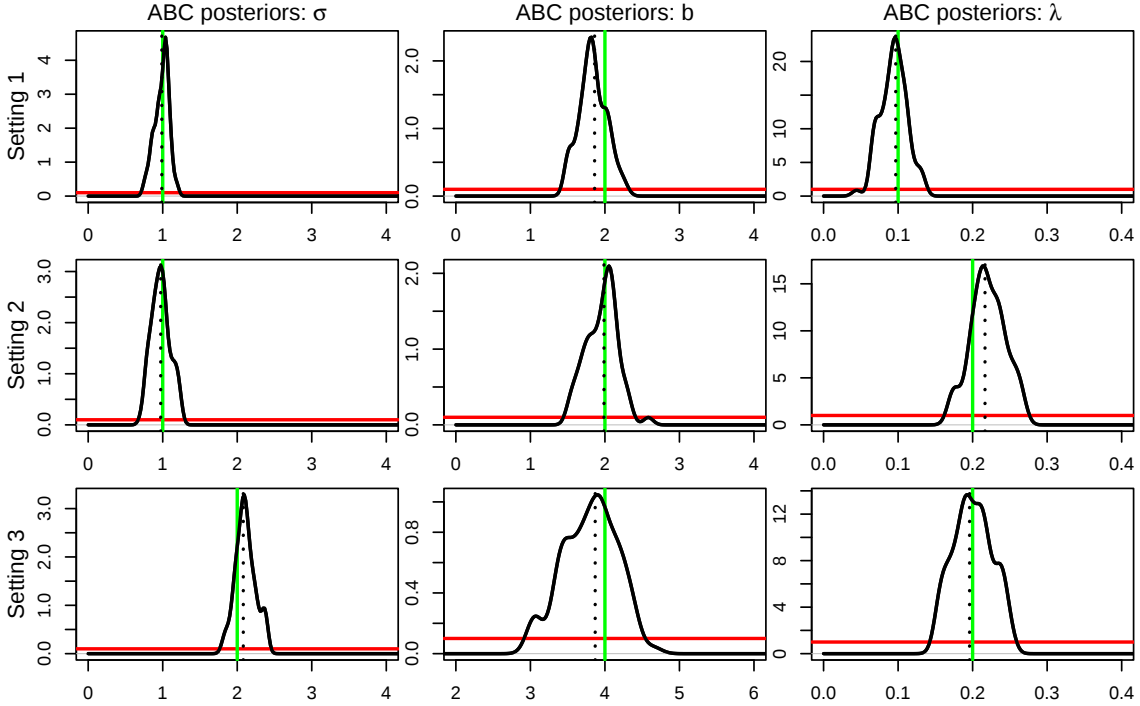


Figure 5: Marginal ABC posterior densities of the parameter vector θ (10) for TP 1 (OU-PDifMP), derived under three different observation settings.

5 Inference results

In this section, we apply the ABC method for inference in PDifMPs, as proposed in the previous section, to the set of test problems with ergodic behaviour introduced in Section 3.2. In Section 5.1, we examine the performance of the ABC algorithm for inferring θ as in (10) in test problems 1–4 under varying observation settings, time horizons, and constant jump rate functions $\Lambda \equiv \lambda$. In Section 5.2, we extend this analysis to non-constant jump rate functions Λ . Finally, in Section 5.3, we evaluate the algorithm’s performance when additional parameters (specifically those appearing in the drift coefficient of the underlying SDE; previously denoted by η) are included in the parameter vector θ (10). Each inference experiment was repeated five times with consistent results. However, for clarity, we report only one representative trial.

5.1 Inference under different observation settings

Inference in TP 1 (OU-PDifMP) for different parameter choices For the first test problem, we estimate θ (10) based on three distinct simulated datasets. Each observed dataset is generated with Algorithm 2, using a time horizon of $T = 500$ and parameter values for σ , b , and λ as

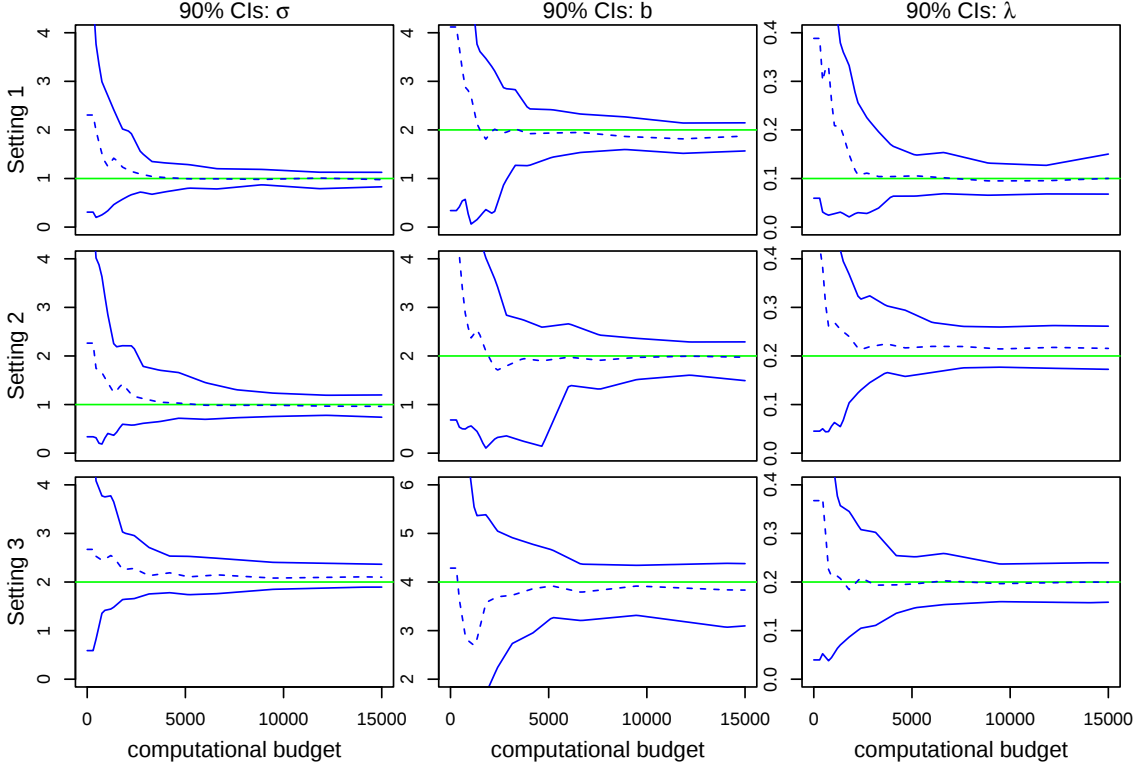


Figure 6: 90% CIs of the marginal posterior densities of the parameter vector θ (10) for TP 1 (OU-PDifMP) as functions of the computational budget, derived under three different observation settings.

specified in Table 1. Setting 1 serves as the standard parameter configuration; in Setting 2, the jump rate of the process is increased, and in Setting 3, all parameters are increased relative to the standard setting.

Figure 5 reports the marginal ABC posterior densities (black solid lines) of θ (10), obtained under the three different observed datasets. Each row corresponds to a specific observation setting, while each column represents a particular parameter. The posterior densities are compared with the corresponding prior densities (red horizontal lines, cf. (11)), for which only a subset of the domain is shown to improve interpretability. Across all settings and parameters, the posteriors are unimodal and cover the underlying true parameter values (green vertical lines). Moreover, the marginal ABC posterior medians (black dotted vertical lines) are consistently close to the true values. Note that the ABC algorithm was stopped after the computational budget (defined as the number of synthetic datasets simulated from the PDifMP) exceeded 10^4 .

Figure 6 shows the 90% credible intervals (CIs) of the posteriors of θ (10) as functions of the computational budget, obtained under the three observation settings. As before, each row corresponds to a specific observation setting, and each column represents a particular parameter. Initially, the CIs (blue solid lines) span broad parameter regions influenced by the respective priors, but they become more narrow and approach the desired posterior regions as the computational budget increases. Correspondingly, the posterior medians (blue dashed lines) approach the underlying true parameter values (green horizontal lines) and approximate them well once the computational budget is sufficiently large.

Inference in TP 1 (OU-PDifMP) for different time horizons We now fix the standard parameter setting (Setting 1) and simulate observed datasets for three different time horizons T . Figure 7 shows the corresponding 90% CIs of the resulting posteriors for θ (10) as functions of the

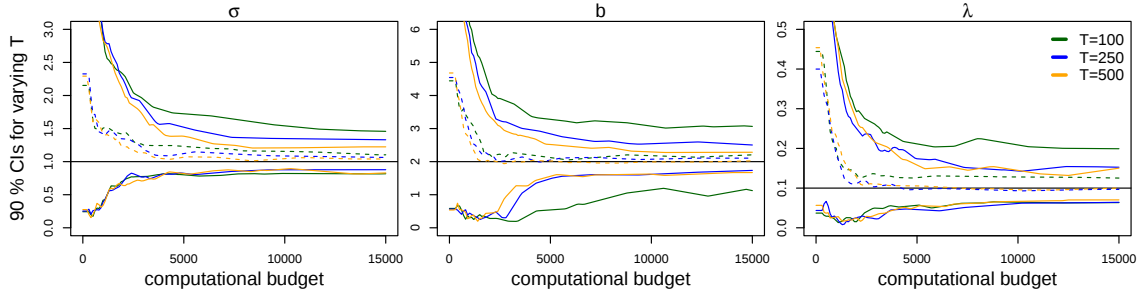


Figure 7: 90% CIs of the marginal posterior densities of the parameter vector θ (10) for TP 1 (OU-PDifMP), derived under three different time horizons T for the standard parameter setting (Setting 1).

Test problem	σ	b	λ
TP 2	1	10	0.1
TP 3	1	2	0.1
TP 4	1	0.1	0.1

Table 2: Parameter settings used to simulate the respective observed dataset for TP 2 (WDSHO-PDifMP), TP 3 (WPWD-PDifMP), and TP 4 (SwitchedSHO-PDifMP).

computational budget. Even for relatively small T , the posteriors cover the true parameter values. As the observation time horizon T increases, the posteriors become narrower and the posterior medians (dashed lines) more closely approximate the true parameter values (horizontal lines). However, observing data over very large time horizons requires substantially greater computational effort, while yielding only marginal improvements in inference accuracy.

Inference in TP 2–4 After validating the ABC method on the first test problem, we proceed to analyze the remaining ones. For each test problem, we consider a single observation setting, with the corresponding true parameter values listed in Table 2. These settings are similar to the standard parameter setting (Setting 1) used for TP 1 (OU-PDifMP), except that the parameter b is adjusted to ensure that TP 2 (WDSHO-PDifMP) and TP 4 (SwitchedSHO-PDifMP) are well defined. The time horizon T is set to 1000 for TP 2 (WDSHO-PDifMP), 1000 for TP 3 (WPWD-PDifMP) and 5000 for TP 4 (SwitchedSHO-PDifMP). We also verified that the proposed ABC method yields similar results under alternative parameter settings.

Figure 8 shows the marginal posterior densities (black solid lines) of θ (10) for test problems 2–4, along with the corresponding prior densities (red horizontal lines). As before, only a subset of the prior domain is displayed (cf. (11)). For all test problems, the posteriors are unimodal and cover the true parameter values (green horizontal lines), which are well approximated by the posterior medians (black dotted vertical lines). It can be observed that the posterior dispersions differ across parameters within the same test problem. For example, in TP 4 (SwitchedSHO-PDifMP), the posteriors of σ and λ have a smaller coefficient of variation than the posterior of b . This indicates that, in this test problem, the parameter b can vary more than the others while still producing paths with comparable summaries. The algorithm was stopped after the computational budget exceeded 1.3×10^4 for TP 2 (WDSHO-PDifMP), 5×10^4 for TP 3 (WPWD-PDifMP), and 1.3×10^4 for TP 4 (SwitchedSHO-PDifMP).

Figure 9 shows the 90% CIs of the posteriors of θ (10) for test problems 2–4 as functions of the computational budget. The x-axis for each parameter differs across the test problems, as for some test problems the posteriors converge to the desired region faster than for others. As observed for TP 1 (OU-PDifMP), the CIs (blue solid lines) initially span broad parameter

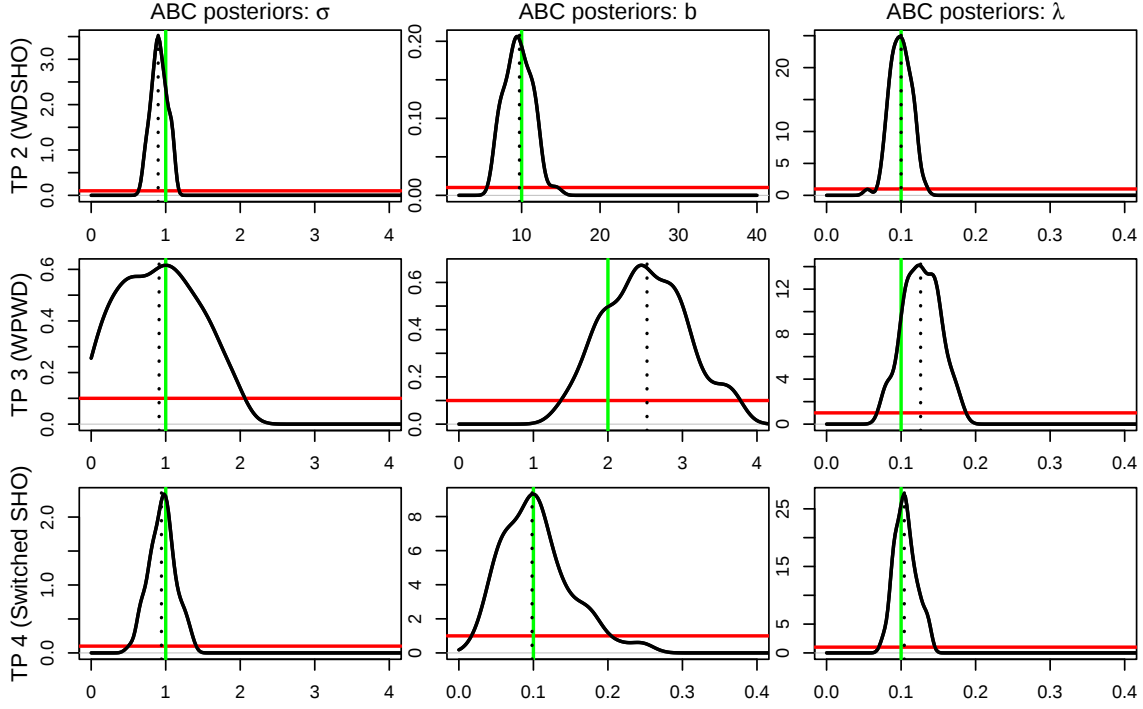


Figure 8: Marginal ABC posterior densities of the parameter vector θ (10) for TP 2 (WDSHO-PDifMP), TP 3 (WPWD-PDifMP), and TP 4 (SwitchedSHO-PDifMP).

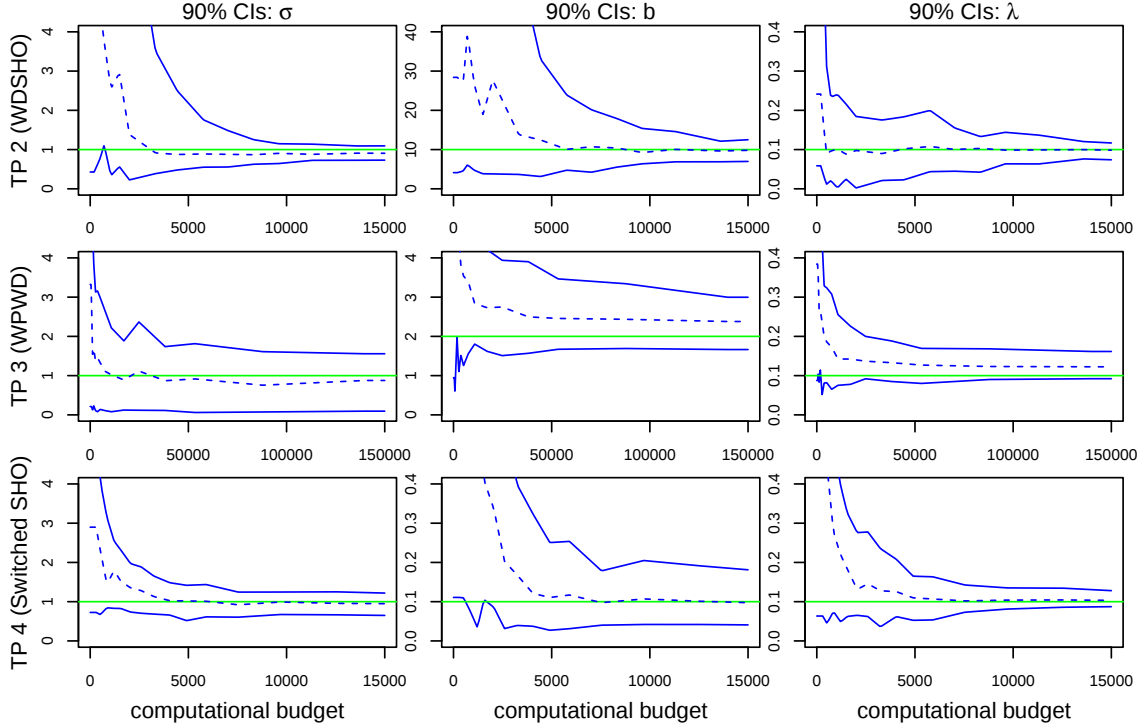


Figure 9: 90% CIs of the marginal posterior densities of the parameter vector θ (10) for TP 2 (WDSHO-PDifMP), TP 3 (WPWD-PDifMP), and TP 4 (SwitchedSHO-PDifMP), as functions of the computational budget.

regions influenced by the respective priors, but they become narrower and converge toward the posterior regions as the computational budget increases. Correspondingly, the posterior medians (blue dashed lines) closely approximate the underlying true parameter values (green horizontal lines) once the computational budget is sufficiently large.

The set of summaries (12) was chosen to enable successful inference in a variety of PDifMPs. However, for some PDifMP models, fewer summaries may suffice to achieve comparably results, while reducing the runtime of the ABC algorithm. For instance, the spectral density S_{x_θ} is particularly useful for capturing the frequency of oscillations. Moreover, for PDifMPs in which the noise parameter σ has a relatively small impact on the density, the mean quadratic variation statistic V_{x_θ} provides valuable information. The invariant density f_{x_θ} provided valuable information about various parameters in all test problems, and the number of jumps N_θ^j is essential for inferring λ .

Finally, we note that for TP 3 (WPWD-PDifMP), the posteriors appear slightly less precise than for the other test problems. In particular, the lower bound of the 90% CI for the parameter σ remains close to the lower bound of the corresponding prior distribution throughout. Moreover, the posteriors for b and λ are not very well centred on the respective true values. We therefore provide a more detailed discussion of this test problem in Appendix A.

5.2 Inference under process-dependent jump rate functions

In the previous experiments, the jump rate function was held constant, i.e. $\Lambda \equiv \lambda > 0$. In this section, we extend our analysis to jump rate functions $\Lambda(X_t; \lambda)$, which depend on the state of the X -component of the PDifMP and a parameter $\lambda > 0$. In particular, we will illustrate that successful inference of θ (10) via the proposed ABC method is also possible when $\lambda \in \theta$ appears as a parameter in Λ .

Process-dependent jump rate functions The following jump rate functions, defined for $x \in D_1$, are investigated:

$$\text{sigmoid: } \Lambda(x; \lambda) = \frac{\lambda}{1 + e^{-x}}, \quad (13)$$

$$\text{reducedCenter: } \Lambda(x; \lambda) = \begin{cases} \lambda/2, & |x| \leq 2 \\ \lambda, & \text{else} \end{cases}, \quad (14)$$

$$\text{cos: } \Lambda(x; \lambda) = \lambda \cos(x) + \lambda. \quad (15)$$

These jump rate functions represent different types of jumping behaviours and are bounded by a constant for all $x \in D_1$ so that the thinning method can be used (cf. Algorithm 3). The sigmoid Λ is monotonically increasing but bounded by λ . The reducedCenter Λ is equal to λ except for the region around 0, more precisely for $|x| \leq 2$, where it is reduced to $\lambda/2$ and overall again bounded by λ . The cos Λ periodically increases and decreases, but is bounded by 2λ .

Figure 10 illustrates how the different jump rate functions affect the path behaviour of the OU-PDifMP. The sigmoid Λ leads to fewer jumps for lower values of X_t , causing the path to move around the negative mean for a longer time than around the positive one. The reducedCenter Λ produces fewer jumps when X_t is close to the origin, making the PDifMP less likely to experience jumps during transitions of X_t from one mean to the other. In contrast, the cos Λ yields a lower jump intensity around $|X_t| = \pi$, which is close the mean in the chosen setting, and therefore the PDifMP is less likely to jump when the path is near the corresponding means.

Similar to Figure 4, Figure 11 illustrates that densities derived from single paths (green dashed lines) of the OU-PDifMP with process-dependent jump rate functions closely resemble the densities computed from multiple realisations of X_{t^*} for sufficiently large t^* (black solid lines). The top left panel corresponds to the top left panel of Figure 4, where the jump rate function is constant. The density of the PDifMP with the sigmoid jump rate function exhibits a bimodal distribution with an asymmetric profile, characterised by a dominant left mode and a smaller secondary mode on the right. For the reducedCenter jump rate function, some probability mass appears near the

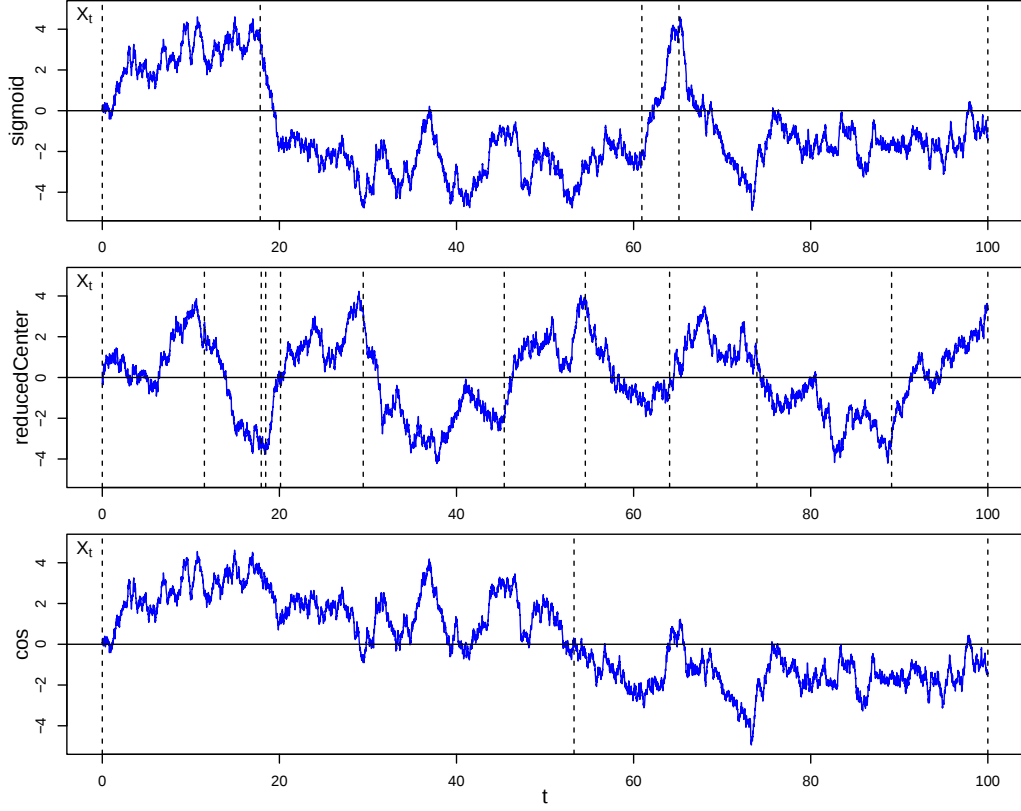


Figure 10: Paths of TP 1 (OU-PDifMP) with sigmoid, reducedCenter, and cos jump rate functions.

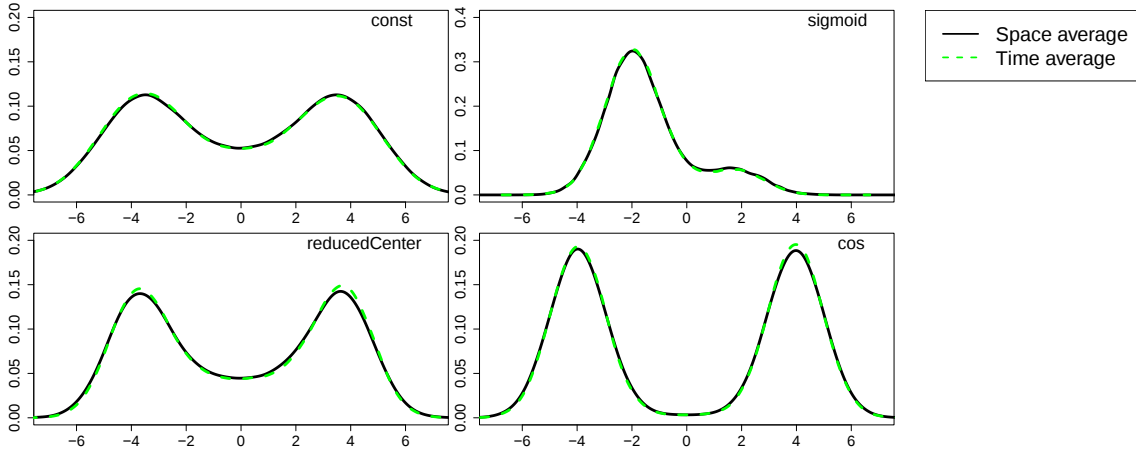


Figure 11: Space and time averages of TP 1 (OU-PDifMP) with constant, sigmoid, reducedCenter, and cos jump rate functions.

centre, although the two peaks remain clearly distinct. In contrast, the density obtained under the cos jump rate function shows two sharply separated peaks with minimal probability mass in the central region. Note that analogous figures can be obtained for the other test problems.

Inferential results We now repeat the inference of θ (10) reported in Section 5.1, this time replacing the constant jump rate function $\Lambda \equiv \lambda$ with the process-dependent jump rate functions (13)–(15). We start with TP 1 (OU-PDifMP), focusing on Setting 1 from before with the same

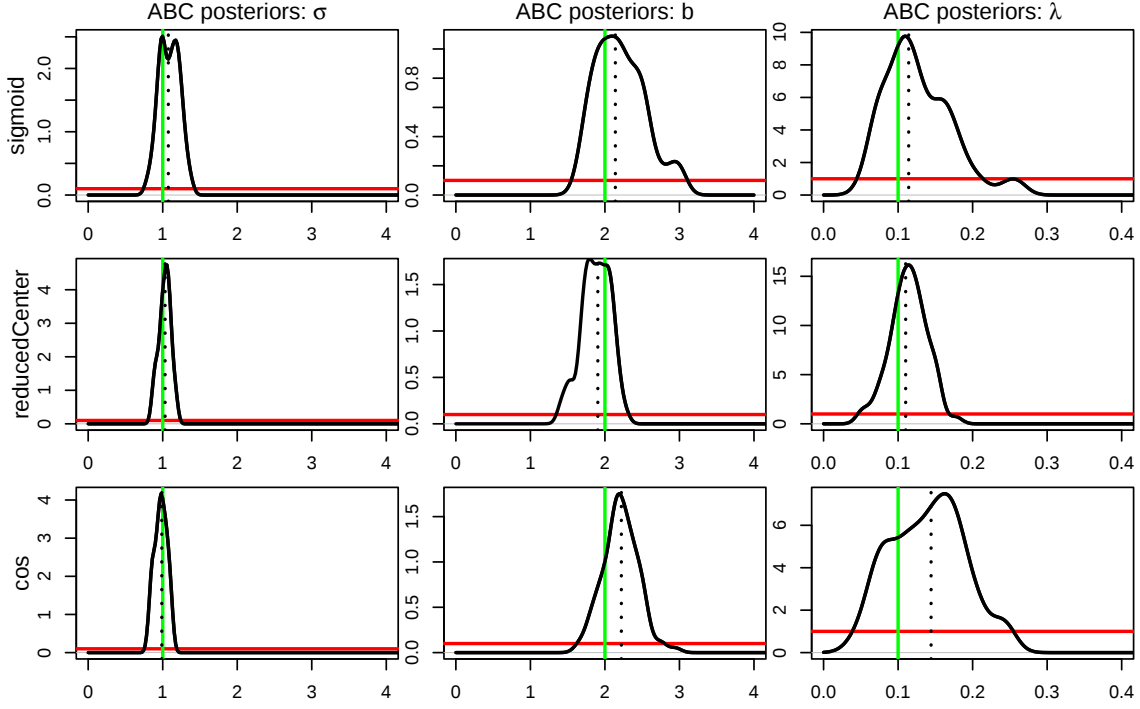


Figure 12: Marginal ABC posterior densities of θ (10) for TP 1 (OU-PDifMP) with sigmoid, reducedCenter, and cos jump rate functions.

time horizon.

Similar to before, Figure 12 shows the marginal posterior densities (black solid lines) obtained under the three jump rate functions, compared to the prior densities (red horizontal lines, cf. (11)). The algorithm was stopped after the computational budget exceeded 4×10^4 for the sigmoid Λ , 10^4 for the reducedCenter Λ , and 1.5×10^4 for the cos Λ . The corresponding 90% CIs are reported in Figure 13. The green solid lines indicate the true parameter values, while the dotted (respectively dashed) lines represent the posterior medians. Comparing these results with those from Setting 1 in Figure 5 and Figure 6, the following observations can be made:

The reducedCenter Λ yields results that are very similar to those obtained with the constant jump rate function. Moreover, for all considered non-constant jump rate functions, the inference of σ closely matches that under the constant jump-rate case, indicating that the estimation of this parameter is highly robust to variations in the jump-rate dynamics. For the cos Λ , the inference of b is likewise consistent with that under a constant Λ . In the case of the sigmoid Λ , the 90% CIs for b suggest a slower convergence to the target posterior region, although the final posterior estimate remains largely unaffected. Regarding the parameter λ , for both the cosine and sigmoid Λ we also observe a slower convergence to the target posterior regime as well as broader CIs and posterior distributions with larger variances, compared to the constant Λ case. Overall, it is noteworthy that the inference performance remains strong across all these different jump rate regimes.

Similar results are obtained for TP 2 (WDSHO-PDifMP) and TP 4 (SwitchedSHO-PDifMP). For instance, the inference of the parameter σ remains relatively robust across different choices of Λ . For the parameters b and λ , we observe slower convergence to the target posterior region in certain cases, with final posterior estimates that are either largely unaffected or show slightly greater dispersion, compared to the constant Λ case. For TP 3 (WPWD-PDifMP), some deviations are observed, as discussed in Appendix A.3.

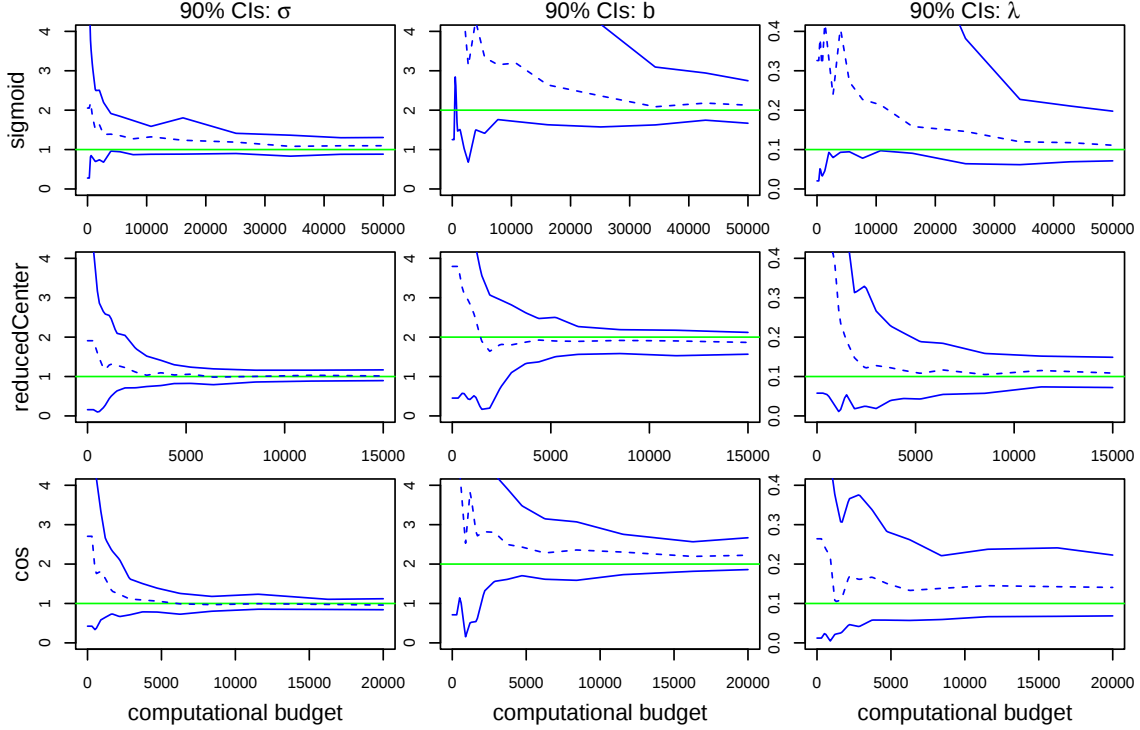


Figure 13: 90% CIs as functions of the computational budget of the marginal posterior densities of θ (10) for TP 1 (OU-PDifMP) with sigmoid, reducedCenter, and cos jump rate functions.

Test problem	η	$\pi(\eta)$
TP 1	1	$U(0, 10)$
TP 4	20	$U(2, 100)$

Table 3: True values and priors for η in TP 1 (OU-PDifMP) and TP 4 (SwitchedSHO-PDifMP).

5.3 Inferring additional parameters

In the previous experiments, the drift parameter η of the underlying SDE was fixed in all relevant PDifMP test problems. Here, we demonstrate that the extended parameter vector

$$\theta = (\sigma, b, \lambda, \eta), \quad (16)$$

can also be successfully inferred with the proposed ABC method. In particular, we use again $\Lambda \equiv \lambda > 0$ and focus on TP 1 (OU-PDifMP) and TP 4 (SwitchedSHO-PDifMP), considering the true values and prior distributions for η reported in Table 3. In TP 1, the parameter η represents the mean-reversion rate, while in TP 4 it influences the oscillation frequencies. The other parameters follow the standard setting, and the time horizon remains the same for each test problem.

Figures 14 and 15 show the corresponding marginal posterior densities and 90% CIs, respectively. Comparing these results with those from Setting 1 in Figures 5 and 6 (for TP 1), and with those from the bottom panels of Figures 8 and 9 (for TP 4), it becomes evident that the inference of σ , b , and λ is not degraded by adding the additional parameter η . Moreover, in both test problems, η is estimated accurately. In particular, for TP 4 (SwitchedSHO-PDifMP), the marginal posterior of η is well centred on the true value and exhibits relatively low variance (see the bottom right panels of 14 and 15). The algorithm was stopped after the computational budget

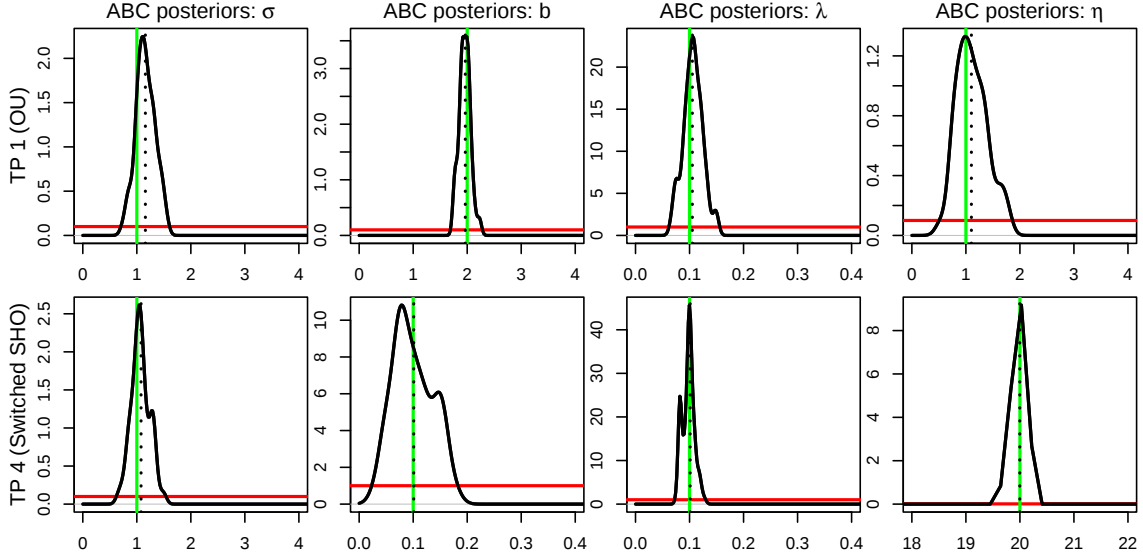


Figure 14: Marginal ABC posterior densities of the parameter vector θ (16) for TP 1 (OU-PDifMP) and TP 4 (SwitchedSHO-PDifMP).

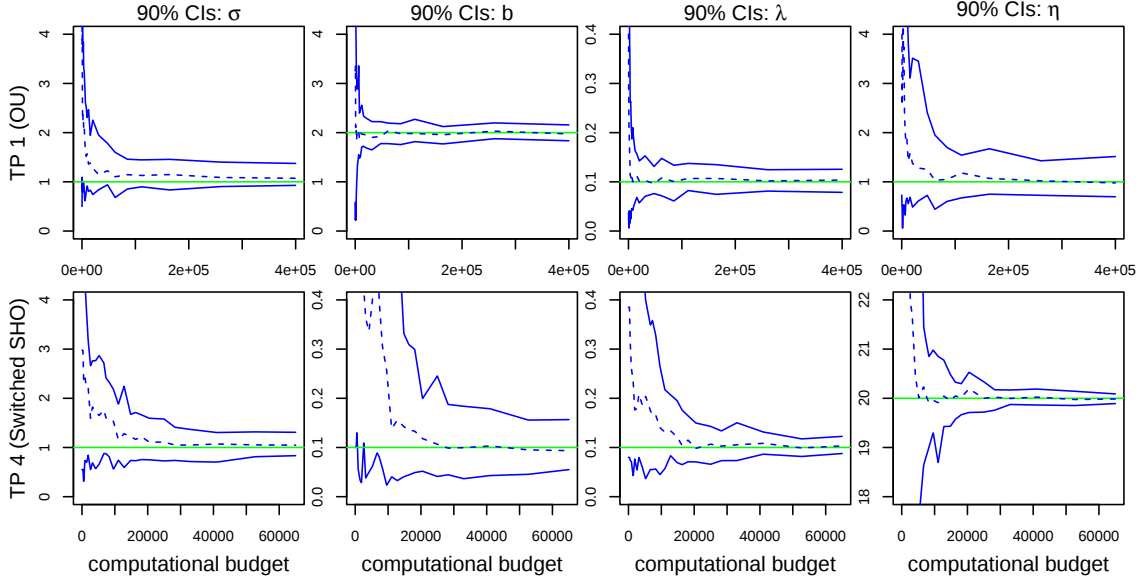


Figure 15: 90% CIs of the marginal posterior densities of the parameter vector θ (16) for TP 1 (OU-PDifMP) and TP 4 (SwitchedSHO-PDifMP), as functions of the computational budget.

exceeded 1.5×10^5 for TP 1 (OU) and 5×10^4 for TP 4 (Switched SHO). Since an additional parameter η is inferred, the convergence is slower, and therefore a larger computational budget was used compared to the settings with three parameters.

6 Conclusion and future work

In this manuscript, we have adapted the approximate Bayesian computation (ABC) methodology for parameter inference in piecewise diffusion Markov process (PDifMP) models that exhibit ergodic behaviour. Specifically, we provided detailed algorithms for simulating PDifMP sample

paths and embedded them within the ABC framework. Furthermore, we extended the summary statistics proposed in [7] for stochastic differential equation (SDE) models to account for the hybrid characteristics of PDifMPs, achieving particular effectiveness in ergodic systems.

The proposed ABC approach successfully inferred model parameters across a range of representative PDifMP test problems that empirically exhibited ergodic behaviour. It performed robustly well under various observation settings, including cases with constant and state-dependent jump rate functions, as well as in challenging scenarios where the jump and/or diffusion components were only partially observed. Our results also indicate that inference accuracy can be further improved by incorporating problem-specific summary statistics. Although this study primarily focused on estimating one parameter per characteristic model component, the method demonstrated strong performance even when additional parameters were included in the parameter vector to be inferred. Moreover, since the example PDifMPs considered here are generic and broadly applicable, our findings suggest that the proposed ABC framework holds promise for a wide range of stochastic hybrid models. Accompanying R code is provided to facilitate implementation and practical use of the method.

This work also opens several avenues for future research. First, while our empirical analyses provide strong indications of ergodicity in the considered test problems, rigorous theoretical proofs remain an open problem. In particular, for TP 3 (WPWD-PDifMP), our results raise some doubts about whether this model is ergodic in general. Second, an important next step will be to extend the ABC methodology to non-ergodic PDifMPs, such as the model recently proposed in [5]. Third, we aim to extend the ABC framework to more complex PDifMP models. The PDifMP test problems studied here involved exactly solvable SDEs, whereas many real-world applications will require efficient and reliable numerical methods to approximate the continuous component. Additionally, we considered state-dependent rate functions with natural upper bounds. However, in applications where such bounds are unavailable, constructing tight artificial bounds will be crucial to maintaining the efficiency of the underlying thinning algorithm. Finally, we plan to apply the developed inference framework to real-world data to further demonstrate its practical potential.

References

- [1] M. A. Beaumont, J.-M. Cornuet, J.-M. Marin, and C. P. Robert. Adaptive approximate Bayesian computation. *Biometrika*, 96(4):983–990, 2009.
- [2] M. A. Beaumont, W. Zhang, and D. J. Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.
- [3] H. C. Berg and D. A. Brown. Chemotaxis in Escherichia Coli analysed by three-dimensional tracking. *Nature*, 239(5374):500–504, 1972.
- [4] H. A. P. Blom. From piecewise deterministic to piecewise diffusion markov processes. In *Proceedings of the 27th IEEE Conference on Decision and Control*, pages 1978–1983. IEEE, 1988.
- [5] E. Buckwar, S. Desmettre, A. Mallinger, and A. Meddah. American option pricing using generalised stochastic hybrid systems. *Journal of Stochastic Analysis*, 6(1):Article 5, 2025.
- [6] E. Buckwar and M. G. Riedler. An exact stochastic hybrid model of excitable membranes including spatio-temporal evolution. *Journal of Mathematical Biology*, 63(6):1051–1093, 2011.
- [7] E. Buckwar, M. Tamborrino, and I. Tubikanec. Spectral density-based and measure-preserving ABC for partially observed diffusion processes. An illustration on Hamiltonian SDEs. *Statistics and Computing*, 30(3):627–648, 2020.
- [8] Evelyn Buckwar, Martina Conte, and Amira Meddah. A stochastic hierarchical model for low grade glioma evolution. *Journal of Mathematical Biology*, 86(6):89, 2023.

- [9] Evelyn Buckwar and Amira Meddah. Numerical approximations and convergence analysis of piecewise diffusion Markov processes, with application to glioma cell migration. *Applied Mathematics and Computation*, 491:129233, 2025.
- [10] M. L. Bujorianu and J. Lygeros. Reachability questions in piecewise deterministic Markov processes. In *International Workshop on Hybrid Systems: Computation and Control*, pages 126–140. Springer, 2003.
- [11] Manuela L. Bujorianu and John Lygeros. *Toward a General Theory of Stochastic Hybrid Systems*, pages 3–30. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [12] C.G. Cassandras and J. Lygeros. *Stochastic Hybrid Systems*. Automation and Control Engineering. CRC Press, 2018.
- [13] B. Cloez, R. Dessalles, A. Genadot, F. Malrieu, A. Marguet, and R. Yvinec. Probabilistic and piecewise deterministic models in biology. *ESAIM: Proceedings and Surveys*, 60:225–245, 2017.
- [14] Michael Creel and Dennis Kristensen. ABC of SV: Limited information likelihood inference in stochastic volatility jump-diffusion models. *Journal of Empirical Finance*, 31:85–108, 2015.
- [15] M. H. A. Davis. Piecewise-deterministic Markov processes: A general class of non-diffusion stochastic models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 46(3):353–376, 1984.
- [16] P. Del Moral, A. Doucet, and A. Jasra. An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing*, 22:1009–1020, 2012.
- [17] S. Desmettre, D. Khurana, and A. Meddah. First-passage time for PDifMPs: an exact simulation approach for time-varying thresholds. *arXiv preprint:2507.07822*, 2025.
- [18] Sascha Desmettre, Devika Khurana, and Amira Meddah. The hybrid exact scheme for the simulation of first-passage times of jump-diffusions with time-dependent thresholds. *arXiv preprint arXiv:2511.00155*, 2025.
- [19] S. Ditlevsen, M. Tamborrino, and I. Tubikanec. Network inference via approximate Bayesian computation. Illustration on a stochastic multi-population neural mass model. *Annals of Applied Statistics*, 2025. to appear.
- [20] D. Eddelbuettel and R. François. Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, 40(8):1–18, 2011.
- [21] S. Filippi, C. P. Barnes, J. Cornebise, and M. P. H. Stumpf. On optimality of kernels for approximate Bayesian computation using sequential Monte Carlo. *Statistical Applications in Genetics and Molecular Biology*, 12(1):87–107, 2013.
- [22] David T. Frazier, Worapree Maneesoonthorn, Gael M. Martin, and Brendan P.M. McCabe. Approximate Bayesian forecasting. *International Journal of Forecasting*, 35(2):521–539, 2019.
- [23] H. Ishijima and M. Uchida. The regime switching portfolios. *Asia-Pacific Financial Markets*, 18:167–189, 2011.
- [24] P. Jovanovski, A. Golightly, and U. Picchini. Towards data-conditional simulation for ABC inference in stochastic differential equations. *Bayesian Analysis*, pages 1–31, 2024.
- [25] T. Kypraios, P. Neal, and D. Prangle. A tutorial introduction to Bayesian inference for stochastic epidemic models using approximate Bayesian computation. *Mathematical Biosciences*, 287:42–53, 2017.

- [26] PA W Lewis and Gerald S Shedler. Simulation of nonhomogeneous Poisson processes by thinning. *Naval Research Logistics Quarterly*, 26(3):403–413, 1979.
- [27] J.-M. Marin, P. Pudlo, C. P. Robert, and R. Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012.
- [28] Amira Meddah. Stochastic hybrid dynamical systems for simulating low-grade glioma evolution. *PhD thesis*, 2024.
- [29] K. Pakdaman, M. Thieullen, and G. Wainrib. Fluid limit theorems for stochastic hybrid systems with application to neuron models. *Advances in Applied Probability*, 42(3):761–794, 2010.
- [30] U. Picchini and M. Tamborrino. Guided sequential ABC schemes for intractable Bayesian models. *Bayesian Analysis*, Advance Publication:1–32, 2024.
- [31] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2011.
- [32] Wolfgang J. Runggaldier. Chapter 5 - jump-diffusion models. In Svetlozar T. Rachev, editor, *Handbook of Heavy Tailed Distributions in Finance*, volume 1 of *Handbooks in Finance*, pages 169–209. North-Holland, Amsterdam, 2003.
- [33] A. Samson, M. Tamborrino, and I. Tubikanec. Inference for the stochastic FitzHugh-Nagumo model from real action potential data via approximate Bayesian computation. *Computational Statistics & Data Analysis*, 204:108095, 2025.
- [34] A. Singh and J. P. Hespanha. Stochastic hybrid systems for studying biochemical processes. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 368(1930):4995–5011, 2010.
- [35] S. A. Sisson, Y. Fan, and M. Beaumont. *Handbook of Approximate Bayesian Computation*. Chapman & Hall/CRC Handbooks of Modern Statistical Methods. CRC Press, Taylor & Francis Group, 2018.
- [36] S. A. Sisson, Y. Fan, and M. M. Tanaka. Sequential monte carlo without likelihoods. *Proceedings of the National Academy of Sciences of the USA*, 104(6):1760–1765, 2007.
- [37] B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer, Berlin, Heidelberg, 6 edition, 2003.

Appendix

A Deeper investigation of TP 3 (WPWD-PDifMP)

In this section, we present additional analyses for TP 3 (WPWD-PDifMP). Section A.1 introduces an auxiliary summary statistic, specifically designed for this test problem when the jump times are observed. In Section A.2, we investigate the influence of the parameter σ on the corresponding estimation results. Finally, Section A.3, provides an extension to non-constant jump rate functions Λ .

A.1 Observation of jump times and extension of summary statistics

In all previous experiments, we assumed that the observed data were of the form (9). In addition to the number of jumps N^j , we now assume to observe the exact times of these N^j jumps, i.e. the observed dataset (9) extends to

$$y = \{x, j\}, \quad \text{where } j = (j_k)_{k=0}^{N^j-1}. \quad (17)$$

This extension enables the construction of additional summary statistics, which may improve the inference of θ (10) using the investigated ABC method.

For instance, given a dataset $y_\theta = \{x_\theta, j_\theta\}$, we can define the summary statistic

$$\text{slope}_{y_\theta} = \text{median} \left\{ \left| \frac{x_{j_{k+1}} - x_{j_k}}{j_{k+1} - j_k} \right|, \quad k = 0, \dots, N_\theta^j - 1 \right\},$$

which incorporates all components of y_θ . This statistic corresponds to the median of the absolute slopes of the X -component between consecutive jumps j_k and j_{k+1} , and is particularly informative for TP 3 (WPWD-PDifMP), where the parameter b is related to the rate at which the time-series increases (or decreases) between jumps. Note that for this statistic we only consider jumps with moves (i.e., a change of the Z -component from b to $-b$ or vice versa indeed happened). It is included into the proposed set of summary statistics (12), yielding

$$s(y_\theta) := \{f_{x_\theta}, S_{x_\theta}, V_{x_\theta}, N_\theta^j, \text{slope}_{y_\theta}\}. \quad (18)$$

We now repeat the inference of θ (10) for TP 3 (WPWD-PDifMP), as reported in Section 5.1, replacing (9) and (12) with (17) and (18), respectively. In Figure 16, we present the corresponding estimation results (orange lines), in comparison with the previous results from Section 5.1 (black and blue lines). The top panels show the marginal posterior densities obtained when stopping the algorithm after the computational budget exceeded 5×10^4 , and the bottom panels display the respective 90% CIs, as functions of the computational budget. Incorporating the slope-summary yields a clear improvement for the parameters b and λ , with posteriors better centred around the true values and exhibiting notably reduced dispersion. In the next section, we investigate the parameter σ more thoroughly.

A.2 Impact of the noise parameter

In this section, we investigate how variations in the parameter σ influence the inferential results. Two settings are considered. Setting 1 corresponds to that in Appendix A.1, with $\sigma = 1$, while in Setting 2 σ is substantially increased to 8.

Figure 17 shows the corresponding 90% CIs (those for Setting 1 coincide with those of Figure 16). Increasing σ accelerates convergence to the target posterior region and yields narrower credible intervals for that parameter. However, the credible intervals for the parameters b and λ become wider.

This behaviour can be explained by looking at Figure 18. For the smaller value of σ in Setting 1, the drift and jump times are more clearly identifiable in the path. In contrast, with the larger σ

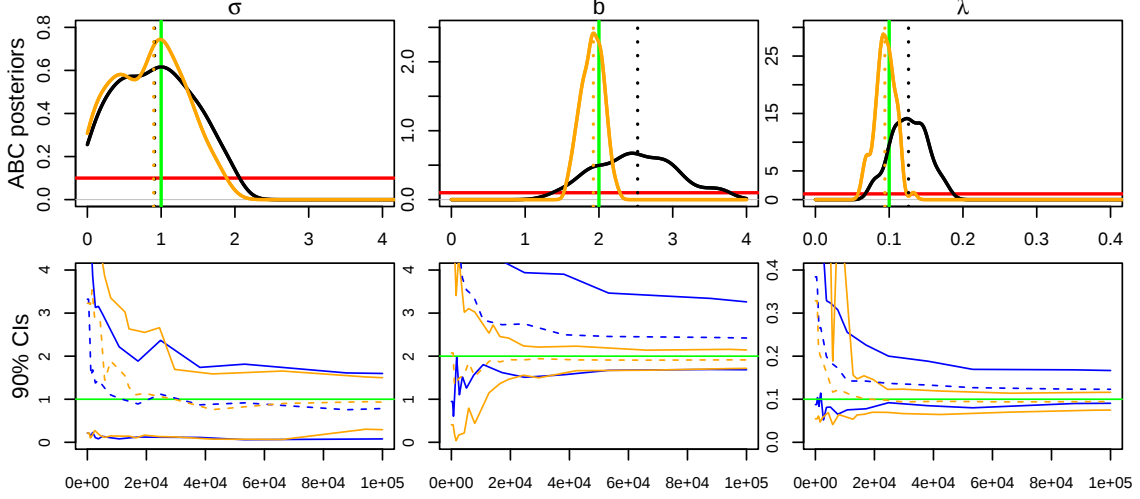


Figure 16: Marginal ABC posteriors densities of θ (10) for TP 3 (WPWD-PDifMP) and 90% CIs as functions of the computational budget. The black (respectively blue) and the orange lines correspond to inferential results derived using the summary statistics (12) and (18), respectively.

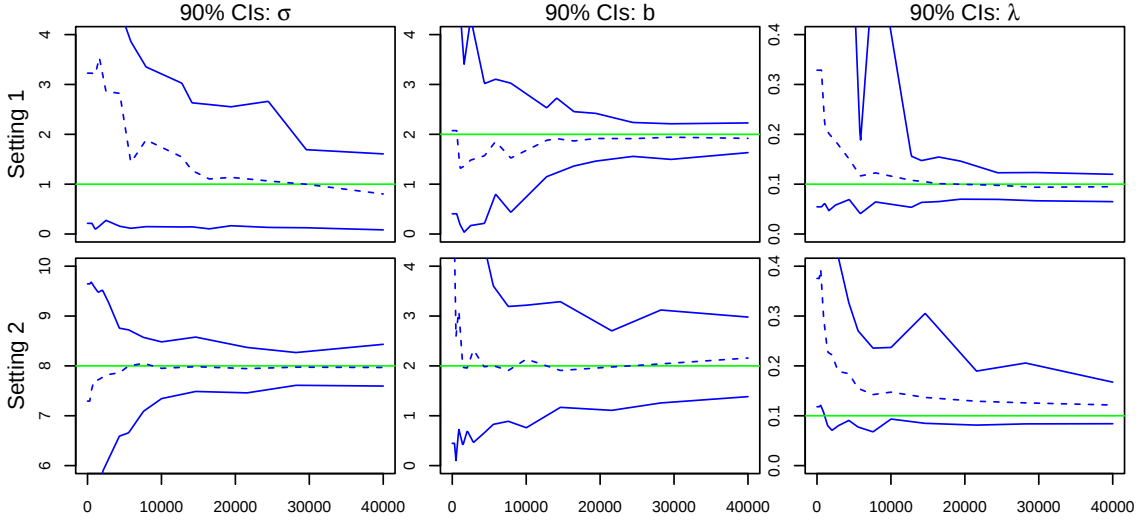


Figure 17: 90% CIs of the marginal posterior densities of θ (10) for TP 3 (WPWD-PDifMP), derived under different values for σ .

in Setting 2, the path exhibits stronger fluctuations, making σ itself easier to estimate, while the drift and jump times become less distinct. This suggests that the beneficial effect of the additional slope-based summary statistic is stronger for smaller values of σ , although satisfactory inference is still achieved under large noise intensities.

A.3 Process-dependent jump rate functions

In this section, we analyse how state-dependent jump rate functions Λ affect the inferential results for TP 3 (WPWD-PDifMP). Specifically, we consider the jump rate functions (13), (14), and (15), respectively, and estimate θ (10), where the parameter λ now appears within these functions. The time horizon is again set to 1000, as in the previous experiments for TP 3, and the summaries (18) are used. Figure 19 shows the corresponding 90% CIs for each of the three inference experiments.

The reducedCenter and cos rate functions have only a minor effect on the inference results

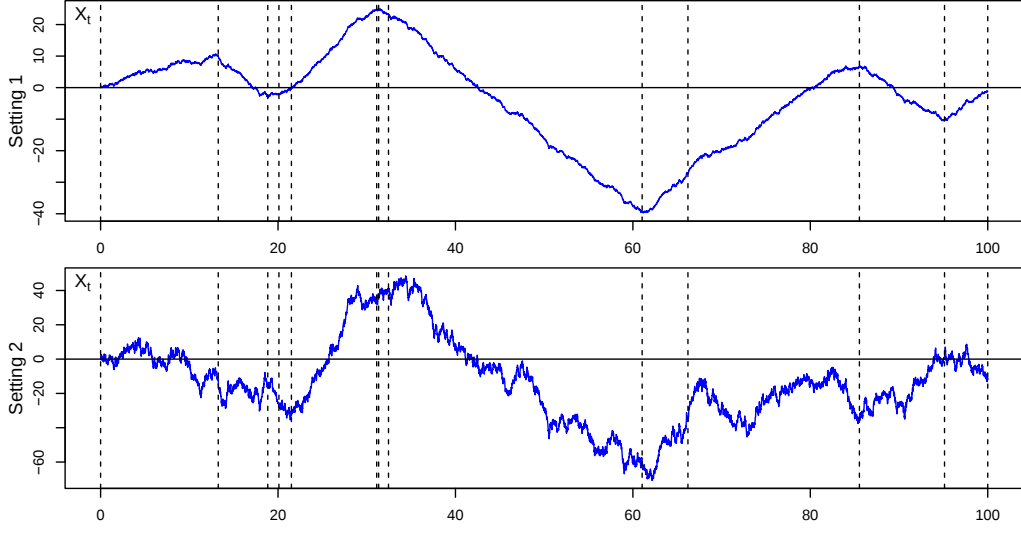


Figure 18: Paths of TP 3 (WPWD-PDifMP) for different values of σ .

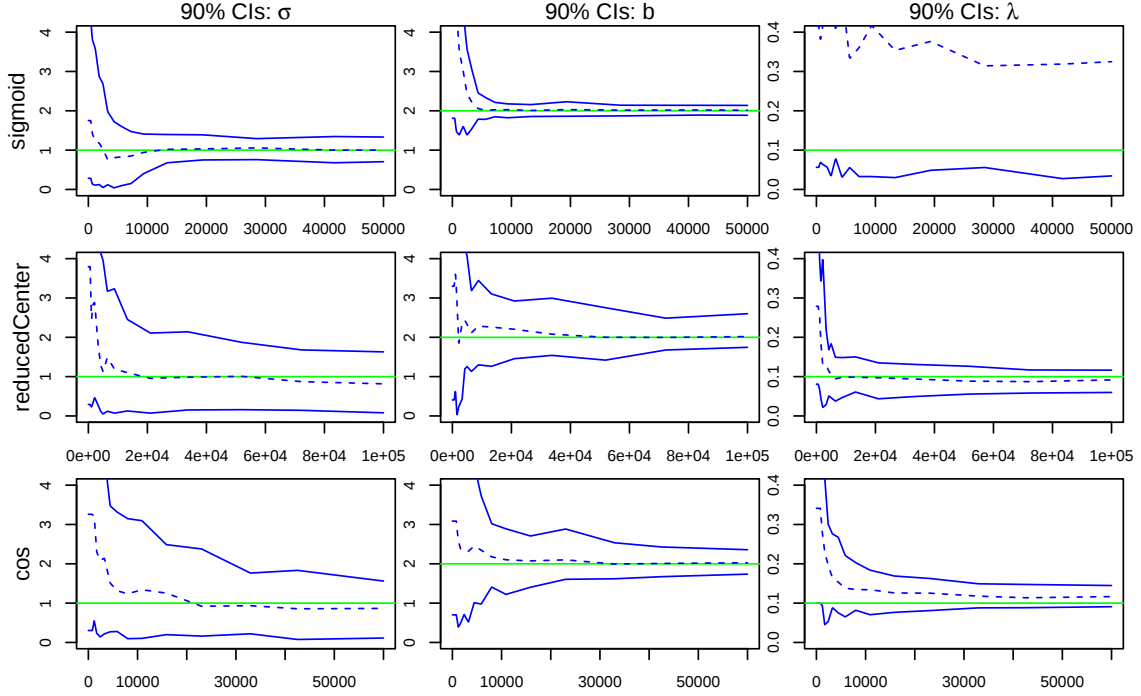


Figure 19: 90% CIs as functions of the computational budget of the marginal posterior densities of θ (10) for TP 3 (WPWD-PDifMP) with sigmoid, reducedCenter, and cos jump rate functions.

compared with the constant λ case (cf. the bottom panels of Figure 16 and the top panels of Figure 17, respectively). In contrast, under the sigmoid Λ , the parameter λ cannot be reliably estimated. This occurs because the jump rate decreases for smaller state values, causing sample paths to drift towards minus infinity and eventually preventing further jumps to happen, even for large λ values. As a consequence, the empirically observed ergodicity of the process breaks down in this scenario, which explains the inability to infer λ . Notably, the estimation of the remaining parameters, σ and b , remains very accurate, even in this non-ergodic regime.