samsara: A Continuous-Time Markov Chain Monte Carlo Sampler for Trans-Dimensional Bayesian Analysis

Gabriele Astorino [1,*] Lorenzo Valbusa Dall'Armi [1,2,†] Riccardo Buscicchio [3,4,5] Joachim Pomper [1,2] Angelo Ricciardone [1,2] and Walter Del Pozzo [1,2]

[1] *Dipartimento di Fisica "E. Fermi", Università di Pisa, I-56127 Pisa, Italy*
[2] *INFN, Sezione di Pisa, I-56127 Pisa, Italy*
[3] *Dipartimento di Fisica "G. Occhialini", Università degli Studi di Milano-Bicocca, Piazza della Scienza 3, 20126 Milano, Italy*
[4] *INFN, Sezione di Milano-Bicocca, Piazza della Scienza 3, 20126 Milano, Italy*
[5] *Institute for Gravitational Wave Astronomy & School of Physics and Astronomy, University of Birmingham, Birmingham, B15 2TT, UK*

(Dated: November 11, 2025)

Bayesian inference requires determining the posterior distribution, a task that becomes particularly challenging when the dimension of the parameter space is large and unknown. This limitation arises in many physics problems, such as Mixture Models (MM) with an unknown number of components or the inference of overlapping signals in noisy data, as in the Laser Interferometer Space Antenna (LISA) Global Fit problem. Traditional approaches, such as product-space methods or Reversible-Jump Markov Chain Monte Carlo (RJMCMC), often face efficiency and convergence limitations. This paper presents `samsara`, a Continuous-Time Markov Chain Monte Carlo (CTMCMC) framework that models parameter evolution through Poisson-driven birth, death, and mutation processes. `samsara` is designed to sample models of unknown dimensionality. By requiring detailed balance through adaptive rate definitions, CTMCMC achieves automatic acceptance of trans-dimensional moves and high sampling efficiency. The code features waiting time weighted estimators, optimized memory storage, and a modular design for easy customization. We validate `samsara` on three benchmark problems: an analytic trans-dimensional distribution, joint inference of sine waves and Lorentzians in time series, and a Gaussian MM with an unknown number of components. In all cases, the code shows excellent agreement with analytical and Nested Sampling results. All these features push `samsara` as a powerful alternative to RJMCMC for large- and variable-dimensional Bayesian inference problems.

I. INTRODUCTION

Bayesian inference essentially involves the computation of the posterior probability distribution for the quantities of interest, X, given a hypothesis, H, a logical context I, also known as the background information, and some observed data D. The entire procedure is encoded in Bayes' theorem:

$$p(X|D\,H\,I) = \frac{p(X|H\,I)p(D|X\,H\,I)}{p(D|H\,I)}\,. \tag{1}$$

Although, at least formally, every inference problem can be solved using Bayes' theorem [1], in practice, the calculations required to determine the posterior distribution $p(X|D\,H\,I)$ can be overwhelmingly challenging. For this reason, most studies dealing with the calculation of $p(X|D\,H\,I)$ rely on numerical methods that generate samples from the posterior distribution. The most widely used and powerful of these methods are Markov Chain Monte Carlo (MCMC) algorithms, in their many variants. In a nutshell, MCMC algorithms are designed to generate samples from a target distribution by stochastically exploring the parameter space. In its simplest implementation, given an initial point for the chain, an MCMC evolves it through a sequence of transitions governed by a user-defined specified proposal distribution. The evolution is modeled to resemble a diffusion process within the target space, and, thanks to ergodicity, the probability density in any given location is proportional to the time the chain spends there. The key principles behind MCMC algorithms are rooted in statistical mechanics. For instance, a necessary condition for MCMC algorithms to generate samples from the target distribution is that their transitions be stationary, that is, the forward and backward transition probabilities must balance. In analogy with thermal equilibrium, detailed balance ensures that the process probability distribution remains stationary.

MCMCs are guaranteed to converge to the target distribution[1], given a sufficient, sometimes infinite, number of steps. The successes of MCMC algorithms are ubiquitous, and many branches of modern science would not be as successful without their invention. Nonetheless, designing an MCMC algorithm is far from trivial, particularly when dealing with parameter spaces of large, and possibly unknown, dimensionality.

In essence every MCMC algorithm draws sets of discrete samples from an underlying continuous stochastic process. In doing so, it discretizes the "inference time axis". It

* gabriele.astorino@phd.unipi.it
† lorenzo.valbusadallarmi@df.unipi.it

[1] In other words, the samples collected by the MCMC follow the target distribution.

is useful to discuss this point further, since it plays an important role in the discussion that follows. A standard Metropolis-Hastings (MH) MCMC, used to draw samples from a target distribution $f(x)$ with proposal distribution $q(x|y)$ can be interpreted as follows:

> Given the current state $x_i \equiv x(t_i)$, the next state is the one generated from the proposal distribution $q(x_{i+1}|x_i)$ with a probability determined by the Metropolis-Hastings acceptance ratio [2, 3].

The above is equivalent to defining a time axis and checking for transitions to new states at *fixed* sampling intervals, given the state at an earlier time.

In a continuous-time (CT) representation, an MC aims at simulating a different process:

> Given the current state x_i, and a set of N possible states $\{x_j(t_{i+1})\}_{j=1}^{N}$, the next state is x_k with probability $\frac{N!}{x_1!\,x_2!\cdots x_N!}\prod_{j=1}^{N} p_j^{x_j}$, and $p_j = f(x_j)/\sum_j f(x_j)$.

Albeit apparently simple, the above interpretation represents a methodological paradigm shift: we represent the inference process as Poisson process, with rates given by our ability of exploring states, given the states we are currently exploring. In other words, the chain is *always* evolving[2] toward a more probable state, we just have to be able to find out which one it is. This is analogous to a path integral formulation of the problem of determining the most probable route to go from point A (the prior) to point B (the posterior) in the space of probabilities.

A CT representation of the exploration of parameter spaces also benefits from the theory of stochastic differential equations. A generic continuous-time stochastic process can be described in terms of i) a deterministic drift term, ii) a stochastic diffusion term, and iii) a discontinuous jump term, see [4]. Therefore, in the CT framework, one can naturally include both random mutations of the states (the diffusion) and changes in the number of parameters necessary to label that state (the jumps)[3]. This makes CT-based inference algorithms very appealing for the exploration of parameter spaces of potentially countably infinite dimension.

This paper presents `samsara`[4], a Continuous-Time Markov Chain Monte Carlo (CTMCMC) algorithm designed to explore countably infinite-dimensional parameter spaces. Popular examples are Dirichlet and infinite Mixture Models (MM) [6, 7], and models of signal inference in noisy data, where the number of signals is a priori unknown.

A widely used approach for handling such spaces is Reversible Jump MCMC (RJMCMC) [8] — an incarnation of a large class of methods often referred to as *product space methods* [9] — in which the chain is allowed not only to explore a given parameter space, but also to jump between spaces of different dimensionality – with an acceptance prescription as for MH —, thereby exploring models with variable numbers of parameters.

Our sampler models parameter space exploration as a birth-death-mutation Poisson process, yielding efficient trans-dimensional exploration, whose rates are set by the requirement of detailed balance, while in-dimension exploration relies on conventional MCMC updates.

We present in detail the algorithm behind `samsara`, we prove the rate prescriptions, and we develop estimators that weight samples by waiting times for improved accuracy. We present the modular implementation and the memory-efficient storage scheme that scales to large populations. We test the code on three benchmark examples: an analytic mixture, a joint recovery of sine waves and Lorentzians time-series, and a Gaussian Mixture model with unknown components, comparing our results with analytical and Nested Sampling outcomes. We show the capabilities of the code and we perform some comparisons about acceptance and efficiency with product-space and RJMCMC approaches.

The paper is organised as follows: in Section II we introduce in detail the general theory of CTMCMC and the theoretical choices behind `samsara`, whose implementation is discussed in Section III. In Section IV, we discuss the `samsara` solution to a few relevant tests cases in general inference. Finally, we conclude with a discussion and a summary of our results in Section V. Detailed calculations are presented in Appendices A, B, C, D, and E.

II. CONTINUOUS-TIME MARKOV CHAIN MONTE CARLO

A. Trans-dimensional parameter space

The CTMCMC algorithm implemented in `samsara` is designed to perform sampling of trans-dimensional distributions and Bayesian inference on problems involving an unknown number of sources or components of different classes. Among those, the Laser Interferometer Space Antenna (LISA) [10] so-called Global Fit [11, 12] and infinite MM [6, 7]. Our algorithm relies on the theory of birth-death processes [13, 14], originally developed to model the biological world and widely used in socio-economics, thus we opted to borrow from their nomenclature.

We shall refer to the state of the chain –the state y – as a

[2] In MCMC operative language, the acceptance probability for "a" transition is always equal to 1.

[3] We defer to future developments the inclusion of a deterministic drift based on Hamiltonian dynamics, as in HMC-based algorithms [5].

[4] In Buddhism, samsara is often defined as the endless cycle of birth, death, and rebirth.

society. Any society is made of a collection of *populations*[5] of *species*– the classes of competing families of models –, and any population is a collection of *individuals*, "atomic" parameter vectors for a given species. Individuals in a species share the same parameter space. We label a species with α, the number of parameters of its individuals as $N_{\mathrm{par},\alpha}$, the number of species in a society with N_{sp}, the number of individuals in the population of the species α with $N_{\mathrm{pop},\alpha}$. The parameters of the i-th individual of the species α are identified for simplicity as $\theta_{i,\alpha}$.

An individual belonging to species α lives in $\Theta_\alpha \subset \mathbb{R}^{N_{\mathrm{par},\alpha}}$, i.e., $\theta_{i,\alpha} \in \Theta_\alpha$. A population α made of $N_{\mathrm{pop},\alpha}$ individuals lives in the Cartesian product of the parameter spaces of the individual sources,

$$\Omega_{\alpha,N_{\mathrm{pop},\alpha}} = \left(\prod_{k=1}^{N_{\mathrm{pop},\alpha}} \Theta_\alpha\right) \subset \left(\mathbb{R}^{N_{\mathrm{pop},\alpha}}\right)^{N_{\mathrm{par},\alpha}} . \quad (2)$$

A society made of N_{sp} species, each comprising $N_{\mathrm{pop},\alpha}$ individuals lives in

$$\Lambda_{\{N_{\mathrm{pop},1},\ldots,N_{\mathrm{pop},N_{\mathrm{sp}}}\}} = \prod_{\alpha=1}^{N_{\mathrm{sp}}} \Omega_{\alpha,N_{\mathrm{pop},\alpha}} . \quad (3)$$

Nevertheless, being the CT Markov Chain formally represented by a point process, we should distinguish between the parameter space in which the full set of parameters lives, and the state space in which the Markov-Chain is represented as a point process.

Let $E_{\alpha,N_{\mathrm{pop},\alpha}}$ be the state space relative to the population $\Omega_{\alpha,N_{\mathrm{pop},\alpha}}$ composed by $N_{\mathrm{pop},\alpha}$ individuals of species α, ignoring the labeling of individuals. The full state space is therefore defined as

$$E = \prod_\alpha \left(\bigcup_{N_{\mathrm{pop},\alpha}=0}^{\infty} E_{\alpha,N_{\mathrm{pop},\alpha}}\right) \equiv \prod_\alpha E_\alpha, \quad (4)$$

where $E_{\alpha,N_{\mathrm{pop},\alpha}}$ are disjoint and E_α is the "full" state space for species α, i.e., the restriction of E on the subspace relative to species α.

As an example, we consider the case of a society, whose individuals represent plane waves. Each wave is characterized by its amplitude, frequency, and phase, denoted as $\theta_{i,\mathrm{sine}} = (A_i, f_i, \phi_i)$. The "sine" population is therefore the collection of $N_{\mathrm{pop,sine}}$ individuals, $\Omega_{\mathrm{sine},N_{\mathrm{pop,sine}}} = \{\theta_{1,\mathrm{sine}},\ldots,\theta_{N_{\mathrm{pop,sine}}}\}$. In the case of a single species, the society is uniquely identified by the "sine population", $\Lambda = \{\Omega_{\mathrm{sine}}\}$. In this case, in each subspace $E_{\mathrm{sine},N_{\mathrm{pop,sine}}}$, the elements are invariant under relabeling of the individuals. For instance, given two fixed sets of parameters θ_1, θ_2, the points $\{\theta_1,\theta_2\}, \{\theta_2,\theta_1\}$ in the space are equivalent in $E_{\mathrm{sine,N_{pop,sine}}}$, while in $\Omega_{\mathrm{sine},N_{\mathrm{pop,sine}}}$ they are distinct vectors.[6]

B. Birth-death-mutation processes

The dynamics of CTMCMC for each species is governed by three processes on E: *birth* of a new individual, *death* of an existing individual, and *mutation* of an existing one. In particular, they describe a birth-death process (discontinuous jump term) and a stochastic diffusion process, respectively. We label them with indices b, d, m, respectively. For convenience, we define y_α the restriction of the point $y \in E$ to the E_α subspace relative to species α, i.e., the point representation of the α population. In what follows, with $y_\alpha \to y'_\alpha$ we refer to the jump in E given by $y \to y' = (y \setminus y_\alpha) \cup y'_\alpha$. In full generality, we say that the initial state of the species α contains $N_{\mathrm{pop},\alpha}$ individuals, $y_\alpha = \{\theta_{1,\alpha},\ldots,\theta_{N_{\mathrm{pop},\alpha},\alpha}\} \in E_{\alpha,N_{\mathrm{pop},\alpha}}$.

- *birth*: A birth process for a species α evolves the state by increasing the number of individuals in the population, i.e., it jumps to a final state containing one or more new individuals, $y_\alpha \to y'_\alpha \in E_{\alpha,N'_{\mathrm{pop},\alpha}}$ with $N'_{\mathrm{pop},\alpha} > N_{\mathrm{pop},\alpha}$. In this work, we focus only on the case in which the final state contains the same individuals of the initial one and an additional, $N'_{\mathrm{pop},\alpha} = N_{\mathrm{pop},\alpha} + 1$

- *death*: A death process for a species α evolves the state by decreasing the number of individuals in the initial state. The state jumps to one containing one or less individuals, $y_\alpha \to y'_\alpha \in E_{\alpha,N'_{\mathrm{pop},\alpha}}$ with $N'_{\mathrm{pop},\alpha} < N_{\mathrm{pop},\alpha}$. Again, we focus on the case in which the final state is obtained simply by removing one individual from the initial population,

$$\begin{aligned} y'_\alpha &= y_\alpha \setminus \theta_{j,\alpha} \\ &\equiv \{\theta_{1,\alpha},\ldots,\theta_{j-1,\alpha},\theta_{j+1,\alpha}\ldots,\theta_{N_{\mathrm{pop},\alpha},\alpha}\}, \end{aligned} \quad (5)$$

  with $y'_\alpha \in E_{\alpha,N_{\mathrm{pop},\alpha}-1}$.

- *mutation*: A mutation process for a species α evolves the state by modifying some parameters of some individuals in the population. The final state contains the same individuals, $y'_\alpha \in E_{\alpha,k}$. Hereafter, we will focus only on final states obtained by mutating only a single individual from the initial ones

$$y'_\alpha = \{\theta_{1,\alpha},\ldots,\theta_{j-1,\alpha},\theta'_{j,\alpha},\theta_{j+1,\alpha}\ldots,\theta_{N_{\mathrm{pop},\alpha},\alpha}\}. \quad (6)$$

Figure 1 shows a schematic representation of the trans-dimensional sampling performed by `samsara`.

[5] A society with only one species is formally equivalent to a single population.

[6] For example, in the case in which θ can assume only two discrete values a and b, $\Omega_{\mathrm{sine},2} = \{(a,a),(a,b),(b,a),(b,b)\}$, while $E_{\mathrm{sine},2} = \{(a,a),(a,b),(b,b)\}$.

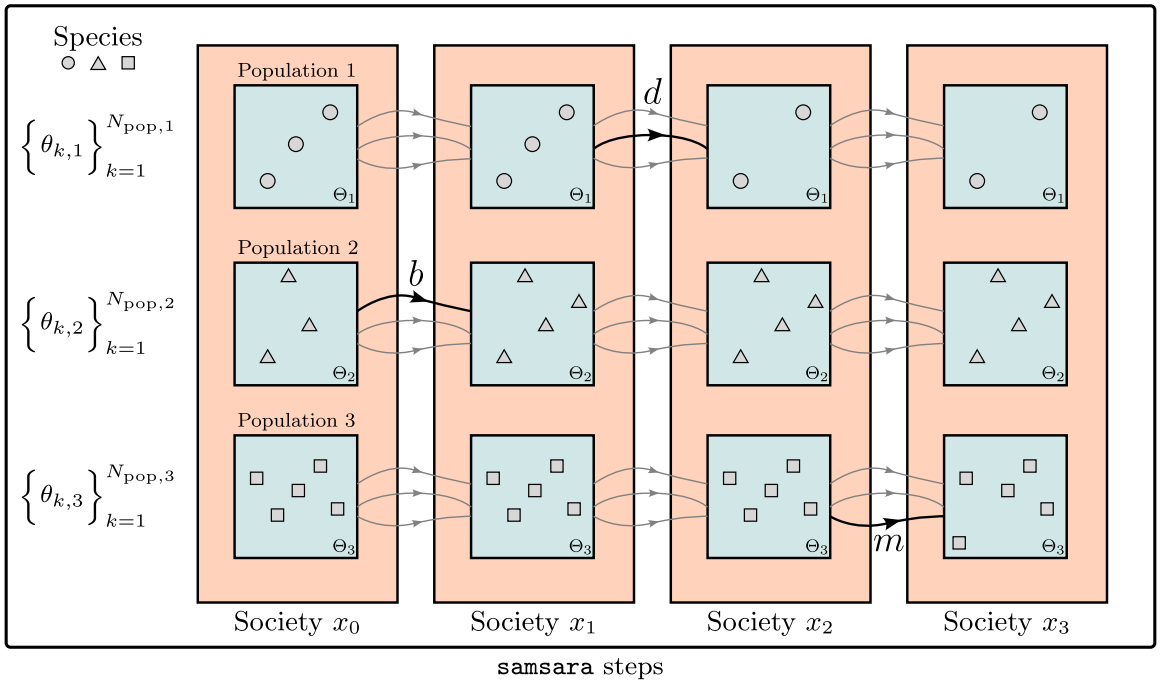

FIG. 1: Structure and evolution diagram of `samsara` trans-dimensional sampling. Each step x_i is identified by a society (orange boxes), whose elements are populations (teal boxes) of $N_{\rm pop,\alpha}$ individuals $\theta_{k,\alpha}$, each describing a point in its parameter space Θ_α of dimension $N_{par,\alpha}$. Individuals from a population are from the same species α. For illustrative purposes, we consider the case of three populations from three distinct species. Individuals from each species are denoted with circle, triangle, and square markers. At each society evolution step, `samsara` evaluates all possible birth, death, and mutation rates, following the decision process described in Sec. II (grey arrowed lines). Then, only one move is chosen and applied to the society, moving onto the next one. We show with a black arrowed line the example of a birth (death, mutation) acting on Population 2 (1, 3), annotated with b (d, m).

In `samsara`, for a given species α and process p, the algorithm proposes $M_{p,\alpha}$ independent moves to evolve the population. For the death process, we consider $M_{d,\alpha} = N_{\rm pop,\alpha}$ moves, each corresponding to the removal of one of the individuals in the population. For the birth process, multiple individuals could in principle be proposed for the addition to the society. However, we focus on the case $M_{b,\alpha} = 1$. For the mutation process, choosing $M_{m,\alpha} > 1$ would correspond to a sampling scheme analogous to the multiple-try proposal [15]. Also in this case, we restrict to $M_{d,\alpha} = 1$. We plan to extend to the case $M_{d,\alpha} = 1$ in future developments.

In Section II C, we discuss the procedure to select the species α, the process p, and the corresponding move j at each step of the MCMC, while a rigorous mathematical description of the moves is provided in Section II D.

C. Dynamics of the Poisson process

In the CTMCMC framework, birth, death, and mutation are treated at every step as independent Poisson processes, each governed by specific rates, representing the expected number of events occurring per unit time. For the process p, the rate associated with the j-th of the $M_{p,\alpha}$ proposed moves for species α is denoted as $R_{j,p,\alpha}$.

For each species, the total rate for a process is given by the sum over the rates of all the proposed moves,

$$R_{p,\alpha} \equiv \sum_{j=1}^{M_{p,\alpha}} R_{j,p,\alpha}\,. \tag{7}$$

This rate represents the total number of moves that take place per unit time associated with the process p for each species. Analogously, we define the total rate for a given species as

$$R_\alpha = \sum_{p=b,d,m} R_{p,\alpha}\,, \tag{8}$$

that represents the expected total number of transitions taking place per unit time, evolving the species α.

Since births, deaths, and mutations are independent Poisson processes, a society evolves over time as follows. For each given species, process, and move, we draw the arrival time $t_{j,p,\alpha}$ of the next Poisson count from its exponential distribution with mean $R^{-1}_{j,p,\alpha}$. We then evolve the society according to the move with earliest arrival time, i.e. the smallest $t_{j,p,\alpha}$.

Algorithmically, the Poisson-step dynamics implemented in `samsara` follows a more practical, equivalent route:

1. starting from the individual transition rates $R_{j,p,\alpha}$, we evaluate the rates per process of a given species $R_{p,\alpha}$ and the total rates for a species R_α using Eqs. (7), (8); we select the population to evolve by sampling from a categorical distribution with probabilities $p_\alpha = R_\alpha / \sum_{\alpha'} R_{\alpha'}$;
2. for the species selected, we choose the process by sampling from a categorical distribution with probabilities $p_{p,\alpha} = R_{p,\alpha}/R_\alpha$;
3. for the species and the process selected, we choose the transition by sampling from a categorical distribution with probabilities $p_{j,p,\alpha} = R_{j,p,\alpha}/R_{p,\alpha}$.

In this workflow, `samsara` selects the population to evolve according to the total rates associated to each species at the given state. However, the algorithm also supports a Gibbs-style sampling scheme, as discussed in Section II D, in which the species of the population to evolve is cycled through deterministically at each MCMC step, while only the process and the specific move are drawn from the rates as described above.

For each state, it is possible to compute an average waiting time, defined as the inverse of the sum of the total rates for the different species,

$$\tau = \frac{1}{\sum_\alpha R_\alpha} \,. \tag{9}$$

The waiting time represents the average lifetime of a state, given the rates $R_{i,p,\alpha}$, i.e., the stability of the state: the larger the waiting time, the larger the probability of such a state. In Section II E we show how the waiting times could be exploited to define more accurate and precise estimators – such as mean values and covariances – w.r.t. standard MCMC.

In the next section, we describe how the rates are either computed or fixed, in order to satisfy detailed balance and to guarantee, together with ergodicity, that the trans-dimensional MCMC yields the target distribution as its stationary one.

D. Detailed balance condition

All MCMC algorithms must satisfy the detailed balance condition and the ergodicity assumption to ensure that the Markov chain admits the target distribution as its stationary one.

Let $P(\cdot \mid D)$ be[7] the target posterior measure with density $p(y|D) = p(y_1 \dots y_{N_{sp}}|D)$, $y \in E$, and let us define $y_{-\alpha} \equiv y \setminus y_\alpha$. Instead of sampling directly from $p(y|D)$, we sample over the species as anticipated in Section II C with two prescriptions. The first follows the CTMCMC dynamics, where the population to evolve is selected according to the rates R_α defined in Eq. (8). The second approach consists of sampling in turn each species using blocked Gibbs sampling [16], evolving states with a deterministic, predetermined sequence over the species. At each step, both approaches require sampling from the conditional distribution $p(y_\alpha|y_{-\alpha}\, D)$[8], which is, in principle, a trans-dimensional problem in itself. Hence, we sample this conditional distribution using CTMCMC, too. We will see in Section III A how they are implemented in practice.

For the first part of this discussion, we focus on birth and death moves, which are each other's inverse process, hence connected through the detailed balance condition. Since the detailed balance for mutations is independent of births and deaths, and it is well known from MCMC theory, we will address it at the end of this section. Assume that a population of species α has been selected for the evolution at a given step of the chain, and denote with $\mu \equiv P(\cdot \mid y_{-\alpha}\, D)$ the target measure with density $p(y_\alpha|D)$, $y_\alpha \in F \subset E_{\alpha,N_{\text{pop},\alpha}}$. We also define[9] $z_\alpha \in G \subset E_{\alpha,N_{\text{pop},\alpha}+1}$ and denote the transition kernels for birth and death by $K_{b,\alpha}(y_\alpha; G)$, and $K_{d,\alpha}(z_\alpha; F)$ respectively.[10] More explicitly, they are the probabilities that the state y_α (z_α) transits to a point in the subspace G (F) given the prescription used for the birth (death).

As in Sec. II B, our birth process involves adding a single individual to the population, proposed from a probability density function h over Θ_α. No specific assumptions are imposed on the proposal distribution h; it can therefore be chosen freely and may also depend on the current configuration of the individuals in the population. This is the case for ensemble proposals, where the new point in the MCMC is proposed based in the location of all [17] – or a subset [18] – of the others. The birth of an individual θ_α occurs with rate $R_{b,\alpha}(y, \theta_\alpha)$. The transition kernel from an initial state y_α to the subspace G is the composition of the probability that a birth occurs and the probability that the new point is θ_α, sampled according to h, such that the arrival state is in G,

$$\begin{aligned} K_{b,\alpha}(y_\alpha; G) &= P\left(y_\alpha \to y'_\alpha \in G | \theta_\alpha\right) \,, \\ y'_\alpha &= y_\alpha \cup \theta_\alpha \,, \\ \theta_\alpha &\sim h(\cdot \mid y) \,. \end{aligned} \tag{10}$$

Similarly, our death process consists in the removal of a single individual from the population of species α.

[7] From now on, we simplify the notation by omitting the hypothesis and the background information in the posterior measure and density.

[8] For simplicity, from now on, we use the compact notation $p(y_\alpha|D) \equiv p(y_\alpha|y_{-\alpha}\, D)$.

[9] As remarked in Section II B, here we focus exclusively on birth processes that generate only a single new individual. However, in general, z could belong to $E_{\alpha,N_{\text{pop},\alpha}+M}$ with M any positive integer.

[10] The dimension of the space where they are applied is clear from the point to which they act and the arrival subspace.

Each individual in the population, $\theta_{j,\alpha} \in z_\alpha$, dies following a Poisson process, with rate $R_{j,d,\alpha}(z, \theta_{j,\alpha})$, with $j = 1, \dots, N_{\text{pop},\alpha}$. The transition kernel from an initial state z_α to a subspace F is given by the product of the probability of proposing a death move and the probability that the death of an individual yields an arrival state in F, summed over all the individuals in the population,

$$\begin{aligned} K_{d,\alpha}(z_\alpha; F) =& P\left(\bigvee_{\theta_{j,\alpha}\in z_\alpha} z_\alpha \to z_\alpha \setminus \theta_{j,\alpha} \in F\right) \\ =& \sum_{\theta_{j,\alpha}\in z_\alpha} P(z_\alpha \to z_\alpha \setminus \theta_{j,\alpha} \in F)\,, \end{aligned} \tag{11}$$

where we used the fact that the union (logical OR) in the first row corresponds to the sum of probabilities, since the events associated with the death of different individuals are mutually exclusive.

The transition kernels reads explicitly

$$K_{b,\alpha}(y_\alpha; G) = \int_{\Theta_\alpha\,:\, y_\alpha \cup \theta_\alpha \in G} \mathrm{d}\theta_\alpha \frac{R_{b,\alpha}(y, \theta_\alpha)}{R_{b,\alpha}(y)} h(\theta_\alpha|y_\alpha)\,, \tag{12}$$

$$K_{d,\alpha}(z_\alpha; F) = \sum_{\theta_{j,\alpha}\in z_\alpha\,:\, z_\alpha\setminus\theta_{j,\alpha}\in F} \frac{R_{j,d,\alpha}(z, \theta_{j,\alpha})}{R_{d,\alpha}(z)}\,. \tag{13}$$

According to [13, 19], the CT Markov Chain has p as its stationary distribution if the integrated death and birth rates balance

$$\int_F \mathrm{d}\mu(y_\alpha) R_{b,\alpha}(y_\alpha) = \int_{\mathcal{E}_{+1}} \mathrm{d}\mu(z_\alpha) \mathcal{R}_{d,\alpha}(z_\alpha; F)\,, \tag{14}$$

$$\int_G \mathrm{d}\mu(z_\alpha) R_{d,\alpha}(z_\alpha) = \int_{\mathcal{E}} \mathrm{d}\mu(y_\alpha) \mathcal{R}_{b,\alpha}(y_\alpha; G)\,. \tag{15}$$

In Eqs. (14), (15) we have used the compact notation $\mathcal{E}_{+1}$, $\mathcal{E}$ as shorthand for $E_{\alpha,N_{\text{pop},\alpha}+1}$, $E_{\alpha,N_{\text{pop},\alpha}}$, respectively. The quantities $\mathcal{R}_{b,\alpha}$, $\mathcal{R}_{d,\alpha}$ represent the transition rate for the birth and death, and are given by

$$\begin{aligned} \mathcal{R}_{d,\alpha}(z_\alpha; F) &\equiv R_{d,\alpha}(z_\alpha) K_{d,\alpha}(z_\alpha; F)\,, \\ \mathcal{R}_{b,\alpha}(y_\alpha; G) &\equiv R_{b,\alpha}(y_\alpha) K_{b,\alpha}(y_\alpha; G)\,. \end{aligned} \tag{16}$$

The detailed balance equations establish the relationship between the birth and death rates, the posterior distribution, the birth proposal, and the number of individuals in the population. As a result, we can determine the birth and death rates as a function of the other quantities that appear in the detailed balance conditions, without the need to implement an accept-reject check. Whenever a birth or a death is chosen, given the rates fixed by Eqs (14) and (15), the Poisson dynamics, see Section II C, the proposed point *is always accepted*, so that no calculation is wasted.

In `samsara` it is possible to adopt two different prescriptions for the rates calculation. The first, based on [13], fixes the birth rate $R_{b,\alpha}(y_\alpha)$ and computes the death rates according to

$$R_{j,d,\alpha}(y) = \frac{\mathcal{Z}(N_{\text{pop},\alpha}, \theta_{j,\alpha})}{N_{\text{pop},\alpha}} \frac{p(x_\alpha|D)}{p(y_\alpha|D)} R_{b,\alpha}(x_\alpha) h(\theta_{j,\alpha}|x_\alpha)\,, \tag{17}$$

where $x_\alpha \equiv y_\alpha \setminus \theta_{j,\alpha}$, and $\mathcal{Z}(N_{\text{pop},\alpha}, \theta_{j,\alpha})$ is a factor describing the prior measure. The second, proposed in [14], lets both birth and death rates to vary, computing them according to the relations derived from the CTMCMC framework

$$R_{j,d,\alpha}(y) = \min\left\{1, \frac{\mathcal{Z}(N_{\text{pop},\alpha}, \theta_{j,\alpha})}{N_{\text{pop},\alpha}} \frac{p(x_\alpha|D)}{p(y_\alpha|D)} h(\theta_{j,\alpha}|x_\alpha)\right\}, \tag{18}$$

$$R_{b,\alpha}(y) = \min\left\{1, \frac{N_{\text{pop},\alpha}+1}{\mathcal{Z}(N_{\text{pop},\alpha}, \theta_{j,\alpha})} \frac{p(z_\alpha|D)}{p(y_\alpha|D)} \frac{1}{h(\theta_{j,\alpha}|y_\alpha)}\right\}. \tag{19}$$

with $z_\alpha \equiv y_\alpha \cup \theta_{j,\alpha}$. We provide mathematical proof for the Eq. (17), Eq. (18) and Eq. (19) and a derivation of the factor $\mathcal{Z}(N_{\text{pop},\alpha}, \theta_{j,\alpha})$ for the test cases discussed in Section IV in Appendix A.

As anticipated in Section II B, our mutation proposals change only the parameters of a single individual in the society. Mutations are formally defined as transitions with $N_{\text{pop},\alpha} = N'_{\text{pop},\alpha}$. The detailed balance condition for transitions at fixed dimensionality is

$$p(y_\alpha|D) Q(y_\alpha \to y'_\alpha) = p(y'_\alpha|D) Q(y'_\alpha \to y_\alpha)\,, \tag{20}$$

where the transition probability is given by

$$Q(y_\alpha \to y'_\alpha) \equiv \xi(y_\alpha) q(y'_\alpha|y_\alpha)\,, \tag{21}$$

with ξ the acceptance and q the proposal distribution. For instance, in the MH algorithm, the acceptance is computed using

$$\xi(y_\alpha) = \min\left\{1, \frac{p(y'_\alpha|D)}{p(y_\alpha|D)} \frac{q(y_\alpha|y'_\alpha)}{q(y'_\alpha|y_\alpha)}\right\}. \tag{22}$$

For fixed-dimensional MCMC samplers, the proposed new point y'_α is accepted with probability ξ, while the current state y_α is retained with probability $1 - \xi$. Within the Poisson dynamics defined in Section II C, this corresponds to assigning the following rates for leaving the state y_α or remaining in the current state,

$$\begin{aligned} R_{m,\alpha}(y_\alpha \to y'_\alpha) =& R_{m,\alpha}(y_\alpha) \xi(y_\alpha)\,, \\ R_{m,\alpha}(y_\alpha \to y_\alpha) =& R_{m,\alpha}(y_\alpha) \left[1 - \xi(y_\alpha)\right]\,, \end{aligned} \tag{23}$$

with $R_{m,\alpha}(y_\alpha)$ the total mutation rate per species α, Eq. (7). The total mutation rate per species α might depend on the state of the sampler. Hereafter, we fix the rate to a constant: $R_{m,\alpha} = 1$. This allows to design and include mutations based on any MCMC sampling schemes, such as MH variants, ensemble sampling, Hamiltonian Monte Carlo, or any variant of Gibbs sampling.

E. Estimators in CTMCMC

An important difference between CTMCMC and standard MCMC is the way they weight states. In CTMCMC each state is associated with a lifetime determined by Poisson dynamics, Section II C. This section reviews how to compute expectation values in CTMCMC, accounting for unequal and finite lifetimes of the states.

Given a function f of the state y, the ergodic theorem ensures that the ensemble average of f can be approximated by averaging $f(y_t)$ along the chain trajectory. In practice, once the chain has reached its stationary distribution, ensemble averages may be replaced by time averages

$$\langle f \rangle = \lim_{T\to\infty} \int_0^T \frac{dt}{T} f\left[y(t)\right] . \tag{24}$$

In CTMCMC, the time average is performed using the chain samples according to

$$\hat{f} = \frac{1}{\sum_{j=1}^{N_{\rm samples}} \Delta t_j} \sum_{i=1}^{N_{\rm samples}} \Delta t_i f\left(y_i\right) , \tag{25}$$

where Δt_i is the time that the chain spends in the state y_i, which corresponds to the minimum between the $t_{i,p,\alpha}(y_i)$, see Section II C. Note that, when all Δt_i are equal, the estimator defined in Eq. (25) is reduces to the standard estimator in equal-time MCMC. In `samsara`, we do not compute estimates as in Eq. (25), but rather we make use of the Rao-Blackwell theorem [20, 21] and the Rao-Blackwellization scheme described in [22, 23], which was applied in the context of CTMCMC in [14]. The Rao-Blackwell theorem ensures that the estimator defined as the expectation value over each Δt_j in Eq. (25) is an unbiased estimator with lower variance than the original one. The Rao-Blackwellized estimator of $\hat{f}$ is therefore

$$\hat{f}_{\rm RB} = \frac{1}{\sum_{j=1}^{N_{\rm samples}} \tau_j(y_j)} \sum_{i=1}^{N_{\rm samples}} \tau_i(y_i) f(y_i) , \tag{26}$$

with τ_j the waiting times associated with the state y_j defined in Eq. (9). The Rao-Blackwellized estimator allows us to not sample each time from an exponential distribution, and it could be helpful in reducing the error on estimates of quantities from the CTMCMC. However, the above discussion is valid when the waiting times are representative of all possible transitions in the parameter space given the state in which they are computed. When $M_{b,\alpha}$ are not fixed, the waiting times are related to a transition w.r.t. one state, thus they could fluctuate around their true value. More details on Rao-Blackwellization and the role of waiting times are provided in Appendix B.

III. IMPLEMENTATION

A. Algorithm

`samsara` samples from trans-dimensional distributions, such as a posterior distribution $p(y|D)$, where the data D have fixed dimensionality, while the parameters y live in the trans-dimensional space described in Section 3. For example, D may represent a time or frequency series, and y the set of parameters characterizing the signals present in the data, as in the case of the LISA Global Fit. Alternatively, D could consist of data samples from a certain distribution, and y the parameters of that distribution, as in Gaussian Mixture Models (GMM).

Our algorithm starts by loading the injected data D, if the problem requires them, and initializing the initial state of the sampler y. This is achieved by defining a society with an initial number of individuals per species, $N^{\rm ini}_{{\rm pop},\alpha}$ with parameters y_α. The algorithm then proceeds for $N_{\rm gen}$ iterations, repeating the following steps:

1. compute the rates $R_{j,p,\alpha}$ for birth and death processes, unless the birth rate is fixed by the user. If the evolution of the population follows the Gibbs-style scheme, the rates are computed only for the population of species α selected to evolve at the current step;

2. determine the species α' of the population to evolve. If the species is selected by the Poisson process, we sample according to $\alpha' \sim {\rm Mult}(1|\tau R_1, \ldots, \tau R_{N_{sp}})$, where the total rates per species and the waiting time are defined in Eqs. (8), (9). If the populations evolve via Gibbs-style scheme, α' is chosen deterministically and the code iterates over all the species at each step;

3. choose the process p' by using the Poisson evolution according to

$$p' \sim {\rm Mult}\left(\cdot \middle| 1, \frac{R_{b,\alpha'}}{R_{\alpha'}}, \frac{R_{b,\alpha'}}{R_{\alpha'}}, \frac{R_{b,\alpha'}}{R_{\alpha'}}\right) ; \tag{27}$$

4. select the move of a given process according to

$$j' \sim {\rm Mult}\left(\cdot \middle| 1, \frac{R_{1,p',\alpha'}}{R_{p',\alpha'}}, \ldots, \frac{R_{M_{p',\alpha'},p',\alpha'}}{R_{p',\alpha'}}\right) . \tag{28}$$

   As discussed in Section II B, in the current version of the algorithm, we fix $M_{b,\alpha} = M_{m,\alpha} = 1$ and $M_{d,\alpha} = N_{{\rm pop},\alpha}$, therefore the selection of the move, i.e., of the individual to remove from the population, is done only for the death;

5. if p' is a mutation, the new point is accepted according to the prescription described in Section II D. If $p' = b$ or $p' = d$ the new point is always accepted;

6. update the society and store it.

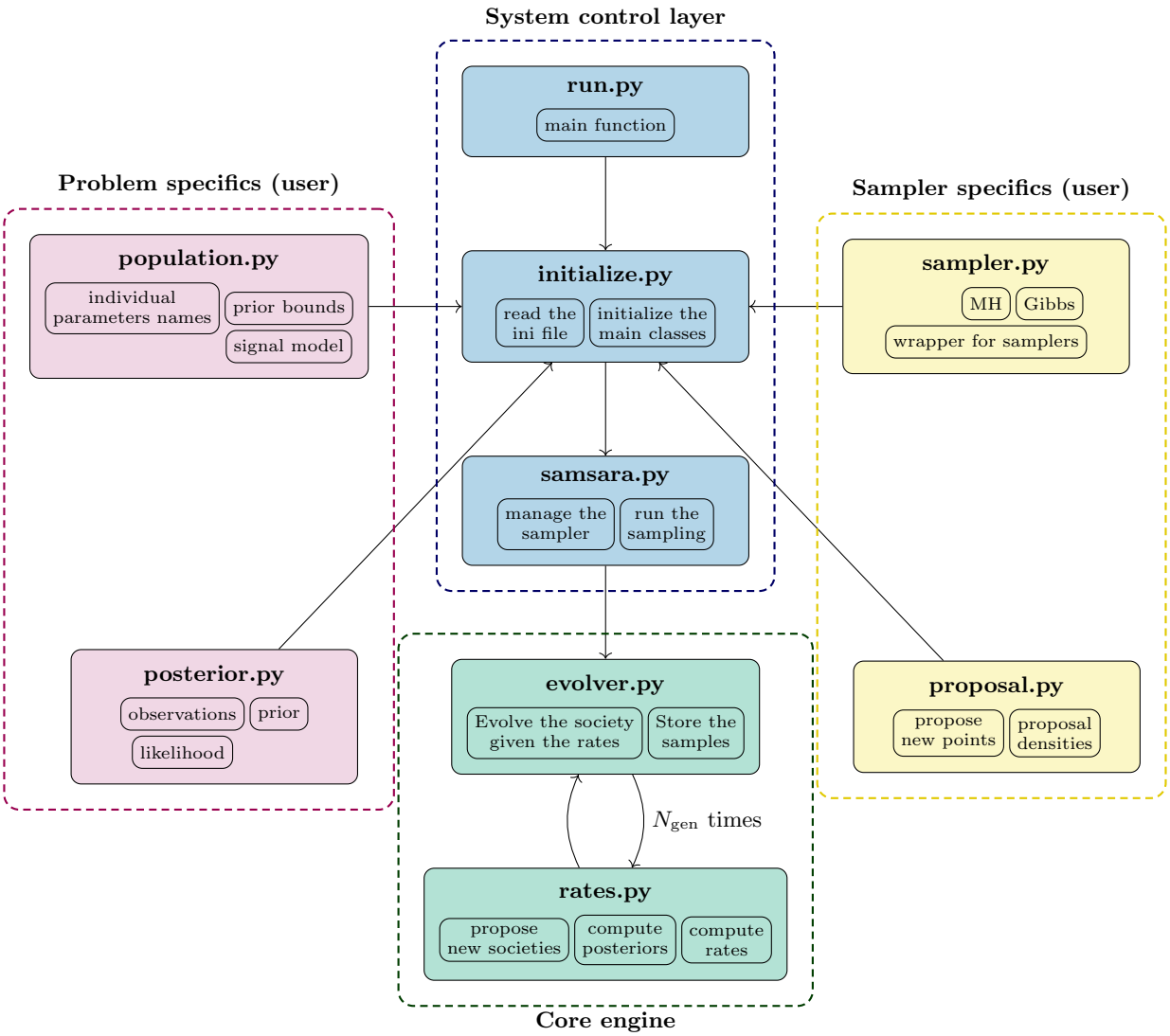

FIG. 2: Code architecture diagram of `samsara`. The teal blue boxes represent the system control layer of the code, which manages internal processes related to input handling and sampler management. The emerald green boxes are the core engine of the code, related to the Poisson dynamics of the CTMCMC. The violet and golden yellow boxes are the input provided by the user related to the inference problem (population and posterior) and sampler (sampler used and proposals).

B. Overall structure of the code

`samsara` is designed as a sampler for a wide range of trans-dimensional problems, such as inference of multiple overlapping signals in noisy data, sampling from trans-dimensional distributions, and mixture models. For this reason, the code is developed with a highly modular architecture, allowing users to implement custom populations, signals models, posteriors, and proposals, without modifying the core of the sampler.

Figure 2 illustrates the main modules of the code and briefly outlines their role within the algorithm. In particular, the module `evolver.py` manages the selection of the species, process, and move used to evolve the society, following the Poisson dynamics described in Section II C. If the selected process evolving the society is a mutation, the evolution is done by calling the `run_mcmc` method of the `sampler` class for the corresponding population. The module `rates.py` is used to compute the rates or the acceptance rules derived in Section II D.

To implement a new trans-dimensional problem, the user needs to define few functions and classes. For the inference, the user must define a likelihood function, and, optionally, a custom prior, otherwise the code will use a uniform prior for each individual and an unbound improper uniform prior over integer numbers. It is impor-

tant to stress that as the posterior will always be proper, this presents no difficulties in interpreting the results [24]. The prior, along with other attributes, are passed as inputs to the `posterior` class in the `posterior.py` module of `samsara`, while the likelihood must be implemented as a `posterior`'s method. The user can either use one of the predefined posterior classes provided in the code or define a custom posterior by inheriting from the `posterior_generator` class in the `posterior.py` module. The custom likelihood function should take as input a `society`, defined as a list of `population` objects defined in the `population.py` module, one per each species.

To define a population object, the user must initialize it providing the names of the individual parameters and their corresponding prior bounds. If the individuals contribute to the likelihood through some signals that combine with the data, a `signal` class must be defined in the `signals` directory. This class should implement the methods `compute_template` and `compute_realization`, depending on whether the signals are deterministic or stochastic.

The `data` in the likelihood are passed by the user as an external input and are stored by the algorithm during the run.

The user can further customize `samsara` by specifying the proposals for each process and the sampler used for the mutations. The code currently contains the following proposals: a prior-based proposal and two types of Gaussian proposals—a standard displacements and a mitosis-like proposal[11]. Alternatively, users can define and implement their own custom proposals. Further details on the available proposals are provided in Appendix C.

In addition, the user can select one of the built-in samplers available in `samsara` which include the MH or Gibbs and Collapsed Gibbs [16] samplers for GMM. Alternatively, the user may implement a wrapper for their own sampler or for any external sampler of choice. `samsara` already include a wrapper for `zeus` [25, 26] and all built-in sampler, together with the classes required to set up custom wrappers, are contained in `sampler.py`.

Finally, the modules `run.py`, `initialize.py`, and `samsara.py` form the internal workflow of the code and are not meant to be accessed or modified by the user. Specifically, `run.py` runs the CTMCMC algorithm, `initialize.py` sets up all the quantities specified by the users in the INI files, and `samsara.py` manages the overall run.

[11] In biology, mitosis is a process in which a eukaryotic cell divides into two genetically identical daughter cells. The mitosis-like proposal implements this idea by proposing the birth or a mutation of an individual being the copy of one individual from the population, plus a small mutation. See Appendix C for details.

C. Optimized storage of the samples

In `samsara`, we keep track of all individuals in the society in each of the $N_{\rm gen}$ iterations of the sampler. If we assume that each species has an average number of individuals $\bar{N}_\alpha$ during the chain, the memory required to store the full society at each iteration scales as

$$\mathcal{M}_{\rm full} \approx N_{\rm gen} \sum_{\alpha=1}^{N_{\rm sp}} \bar{N}_\alpha N^\alpha_{\rm par} \times 8\,{\rm B}\,. \tag{29}$$

Such an amount of memory could be very challenging in some cases. For example, in the case of Double White Dwarfs (DWD) systems in LISA [27], where $\bar{N}_{\rm DWD} \approx 10^3$, $\bar{N}_{\rm par} \approx 10$, and $\bar{N}_{\rm gen} \approx 10^8$, the required memory would be $\mathcal{M}_{\rm full} \approx 10\,{\rm TB}$. Furthermore, any kind of post-processing would need to allocate $1-10\,\mathcal{M}_{\rm full}$ for the time/frequency series of the signals in the Markov Chain.

For this reason, we have optimized the memory used in the algorithm by avoiding redundant copies of the individuals in the society. More specifically, an individual survives in the population either when its proposed drift is not accepted or when other individuals in the society are evolved (through any kind of process). Therefore, the individual's history is fully specified by its birth and death generations. Our algorithm therefore requires just to store the parameters associated with the individual one time and two integers, which refer to the generation of birth, g_b, and to the generation of death, g_d, of the individual. In this scheme, the memory cost equals the number of unique individuals that appear in the society, multiplied by the memory needed to store their parameters and the two generation indices

$$\mathcal{M}_{\rm opt} \approx N_{\rm gen} \sum_{\alpha=1}^{N_{\rm sp}} \bar{R}^{\rm accept}_\alpha \,(N^\alpha_{\rm par}+16){\rm B}\,, \tag{30}$$

with $\bar{R}^{\rm accept}_\alpha$ the average acceptance rate, while the 16 B are used to store birth and death dates. Since $\bar{R}^{\rm accept}_\alpha \in [0,1]$, $\mathcal{M}_{\rm opt} \leq \mathcal{M}_{\rm full}$ when $\bar{N}_\alpha \geq 3$. Such an indexing storage system is particularly convenient when a large number of individuals in the society are present or when the acceptance rate of the mutation is low.

In `samsara` all the information is stored in a dictionary `saved_samples`, whose keys correspond to the species in the society. For each species, the value associated with its key is itself a dictionary containing the following entries:

- `values`, corresponding to an array of shape $(N^\alpha_{\rm unique}, N^\alpha_{\rm par})$, with $N^\alpha_{\rm unique}$ the number of unique individuals appeared in the society, i.e., the sum of the number of births for the species α and of the accepted drifts;

- `lifetime`, corresponding to an array of shape $(N^\alpha_{\rm unique}, 2)$ where the entry $[i,0]$ and $[i,1]$ are the generations at which the individual with values i entered and left the society, i.e, the birth and death dates.

Generation	**0**	**1**	**2**	**3**	**4**	**5**
Event	Empty society	Birth (θ_0)	Unaccepted drift	Birth (θ_1)	$\theta_0 \to \theta_0'$	Death (θ_1)
Parameters	[]	$[\theta_0]$	$[\theta_0]$	$[\theta_0, \theta_1]$	$[\theta_0, \theta_1, \theta_0']$	$[\theta_0, \theta_1, \theta_0']$
Birth Dates	[]	[1]	[1]	[1, 3]	[1, 3, 4]	[1, 3, 4]
Death Dates	[]	$[-1]$	$[-1]$	$[-1, -1]$	$[4, -1, -1]$	$[4, 5, -1]$

TABLE I: Example of the storage indexing system of CTMCMC in the case of one species, starting from an empty society. "Generation" represents the step of the chain considered, while "Event" the move that takes place at that time.

In Table I, we provide an example of how the parameters and the birth and death dates are updated in the algorithm, in the case of one species, starting from an empty population.

Nevertheless, this specific saving scheme is not always applicable. For instance, in cases where multiple individuals are updated within the same iteration, such as in GMM problems that mutate via Gibbs samplers. For these situations, `samsara` also provides a standard saving option based on HDF5 (`.h5`) files to store the samples. The user can freely choose which saving method to employ.

D. Diagnostic

The output of `samsara` is a chain of $N_{\rm gen}$ samples, in which both the number of individuals per species and the parameter values associated with each individual may vary between samples. The parameters $\theta^{(i)}_{j,\alpha}$ of the j-th individual of the α-th species of the i-th sample form an array of length $N^{\alpha}_{\rm par}$.

Since the number of individuals per species is not fixed, the correlation length of the chain and other diagnostics cannot be computed as in standard MCMC, because the expectation values of the parameters are ill defined. For this reason, we introduce a scalar function ρ of the society

$$\rho : \Lambda_{\{N_{\rm pop,1} \dots N_{\rm pop,N_{sp}}\}} \to \mathbb{R}\,, \tag{31}$$

where the parameter space of the society is defined in Eq. (3). The function ρ allows us to extract a scalar quantity from the parameters of the society, bypassing the trans-dimensional nature of the sampling problem. Several choices of ρ are possible, depending on the type of correlation expected in the samples or the diagnostic considered.

For the autocorrelation function we choose ρ to be the log-posterior, as discussed in more detail in Appendix D 1, but other choices are possible, such as the number density of the individual species. Similarly, the convergence of the chain is assessed by introducing a set of reference points for each population in the Θ_α space, and build scalar quantities based on the distance of the population itself from these points [28, 29]. Additional details on the distance measures used and the specifics of the convergence tests are given in Appendix D 2.

E. Post-processing

The post-processing in `samsara` is done using the thinned samples with correlation length defined in Section III D. We identify here with $N'_{\rm gen}$ the number of samples after the thinning.

The first quantity we extract is the posterior on the number of individuals per each species in the society, which is obtained by the weighted average of the number of individuals per species, normalized by their mean value,

$$p\left(N_{{\rm pop},\alpha}\right) \equiv \frac{\sum_{i=1}^{N'_{\rm gen}} \tau^{(i)} \mathbf{1}_{N^{(i)}_{{\rm pop},\alpha} = N_{{\rm pop},\alpha}}}{\sum_{i=1}^{N_{\rm gen}} \tau^{(i)}}\,, \tag{32}$$

with $\tau^{(i)}$ the waiting time associated to the i-th sample and $\mathbf{1}_{\rm condition}$ the indicator function which is one when the condition is true and zero when the condition is false.

The second quantity we extract from the samples is the distribution of the parameters with the population of each species. This quantity could correspond, for instance, to the distribution of angular position in the sky or frequency of DWD. For distribution of the k-th parameter of the α-th species, we flatten into a single array all the values of the k-th parameters from the individuals in the α-th population across the $N'_{\rm gen}$ samples. We then plot the normalized histogram of this array, weighting each sample by its corresponding waiting time. We notice that this quantity becomes informative in two regimes. First, when the spacing between source parameters is larger than the uncertainty in the estimates of individual parameters, the resulting distribution reflects the posteriors of individual sources. Second, when the number of sources is large, the distribution reveals the underlying population features, with minimal contamination from the shot noise associated with a single realization.

The post-processing of `samsara` computes also the median and the confidence intervals of functions of the form

$$f : \Upsilon \subset \Lambda_{\{N_{\rm pop,1} \dots N_{\rm pop,N_{sp}}\}} \to \mathbb{R}^M\,. \tag{33}$$

For example, in the case of GW sources in LISA, `samsara` can reconstruct the total signal emitted by a given species, such as DWD, in each LISA channel. This is possible because, although the input is trans-dimensional, the output is not - as discussed in Section III D for the computation of the autocorrelation length.

As already stressed, the output of `samsara` are $N'_{\rm gen}$ lists of $N_{\rm sp}$ populations, containing $N_{{\rm pop},\alpha}$ individuals, whose number can vary between samples. However, one may be interested in getting for each of the $N_{\rm sp}$ species a list of $N^{\rm eff}_{{\rm pop},\alpha}$ catalogs of length $N_{{\rm gen},j,\alpha}$, where each catalog refers to a specific source. Since a catalog should contain at least one sample, the maximum number of catalog one can construct for the species α is $N^{\rm eff}_{{\rm pop},\alpha} \leq \max(N^{(i)}_{{\rm pop},\alpha})$. In addition, since some samples could have $N^{(i)}_{{\rm pop},\alpha} < N^{\rm eff}_{{\rm pop},\alpha}$, the length of the samples for a given catalog is bounded by $N_{{\rm gen},j,\alpha} \leq N'_{\rm gen}$. Schematically, we perform a reordering of the data structure: from a list of samples, each containing a list of populations, each in turn containing a list of individuals, to a list of populations, each containing a list of individuals, each associated with a list of samples. In a coding-style array notation, the unordered and reordered samples can be expressed as

$$\theta^{(i)}_{j,\alpha} = \begin{cases} \texttt{samples}[i][\alpha][j] \\ \texttt{catalog}[\alpha][j][i] \end{cases} \tag{34}$$

To produce a catalog from the trans-dimensional samples, we use `PETRA` [30], which performs a relabeling of the samples to minimize the statistical divergence between the trans-dimensional samples and a factorized representation of the catalog.

IV. TEST RESULTS

A. Analytic Test Case

As a first check, before addressing more realistic inference problems, we demonstrate that the CTMCMC techniques, used within `samsara`, faithfully sample trans-dimensional distributions, which in itself is a non-trivial task. This test guarantees that we can use the waiting times $\tau^{(i)}$ to accurately estimate the posterior for the number of signals or the population posterior of the model parameter, as described in section III E, for a real inference problem. For this, we replace the likelihood function, dictated by the data, with an analytically understood trans-dimensional distribution

$$p(N,\theta|\bar{N},\omega) = \mathcal{P}(N|\bar{N})p(\theta|N,\omega)\,, \tag{35}$$

and use `samsara` to sample from it. To mimic a simplified version of the trans-dimensional posterior landscape, as it could be encountered, we choose

$$\mathcal{P}(N|\bar{N}) = \frac{\bar{N}^N}{N!}e^{-\bar{N}}\,, \tag{36}$$

$$p(\theta|N,\omega) = \prod_{i=1}^{N} p(\theta_i|\omega)\,. \tag{37}$$

$$p(\theta_i|\omega) = \sum_{j=1}^{3} \frac{w_j}{2\pi\sqrt{\det\Sigma_j}} e^{-\frac{1}{2}(\theta_i-\mu_j)^T\Sigma_j^{-1}(\theta_i-\mu_j)} \tag{38}$$

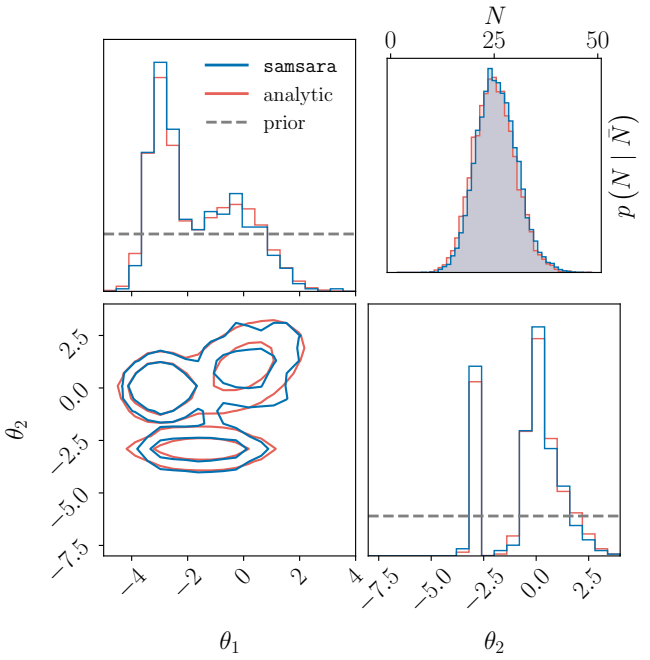

FIG. 3: Posterior distribution for the number of sources (upper right panel) and parameter contours (corner plot) for the analytical model described in Eq. (35). `samsara` (analytical) results and priors are shown with deep blue (coral red) and dashed gray lines, respectively. No true values are presented since we are sampling from the trans-dimensional distribution in Eq. (35) and not performing an inference problem, as discussed in Section IV A.

Here, $\mathcal{P}(N|\bar{N})$ represents the posterior of the number of signals via a Poisson distribution, and $p(\theta|N,\omega)$ mimics the population posterior on θ. Therefore, $\{\theta_i\}$ are assumed independently sampled from the population distribution $p(\theta|\omega)$. Modeling the latter as a Gaussian mixture, we can introduce bi-modalities as well as two probability phases separated by a region of low probability. Such features typically obstruct sampling and allow us to further demonstrate the robustness of the CTMCMC implementation in `samsara`. Specifically, we choose

$$\begin{aligned} w &= \left\{\frac{8}{18}, \frac{4}{18}, \frac{6}{18}\right\}, \\ \mu &= \left\{\begin{pmatrix}-3\\0\end{pmatrix}, \begin{pmatrix}-1.5\\-3\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix}\right\}, \\ \Sigma &= \left\{\begin{pmatrix}0.2 & 0\\0 & 0.2\end{pmatrix}, \begin{pmatrix}1.3 & 0\\0 & 0.01\end{pmatrix}, \begin{pmatrix}1 & 0.5\\0.5 & 1\end{pmatrix}\right\}. \end{aligned} \tag{39}$$

Adopting uniform priors, the replaced likelihood model mimics the full posterior. The priors on θ_1 and θ_2 are listed in Table II, while for N we adopt the default improper unbound uniform prior over integer numbers. Further, $\mathcal{Z}(N_{{\rm pop},\alpha},\theta_{j,\alpha}) = 1$. The distribution (35), while capturing the main features of a trans-dimensional sampling problem, is simple enough to be sampled efficiently

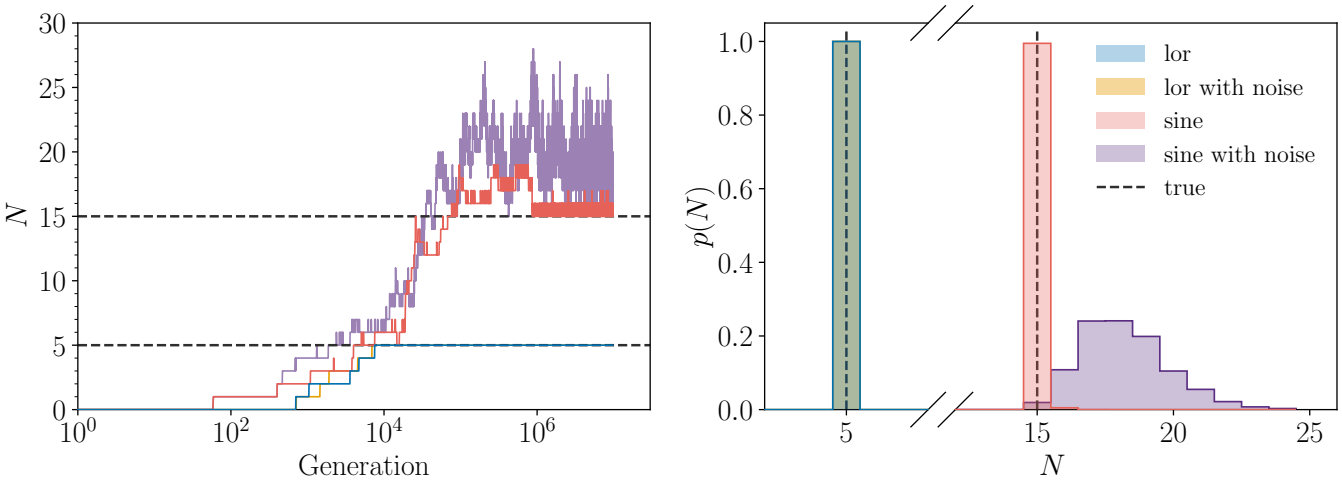

FIG. 4: Plot of the trace of the number (left panel) and posterior on the number (right panel). In deep blue (coral red) the Lorentzians (sine waves) for the zero noise case, while in amber (deep purple) the Lorentzians (sine waves) for the noisy case. Truths are in dashed drak gray.

by other means and thus allows comparison with the results of our algorithm. In Figure 3, we show the marginal posteriors for both the number of sources and the population distributions of the model parameters. We compare the post-processed output of `samsara` with the results of sampling (35) directly, choosing a number of components and then drawing them independently according to the Gaussian mixture distribution $p(\theta|N,\omega)$ in (36). For the samples depicted in Figure 3 we ran the code for $N_{\rm gen} = 10^7$ iterations, with Gaussian proposals for θ_1 and θ_2 with $\sigma_{{\rm proposal},\theta_1} = 0.005$, $\sigma_{{\rm proposal},\theta_2} = 0.002$. We conclude that the output of `samsara` is in excellent agreement with the test samples, providing validation for the deployed CTMCMC sampling technique.

Finally, we take advantage of this simple test case to apply the convergence test implemented within `samsara`. The details are in Appendix D 2.

B. Sine waves and Lorentzians

As a second test, we study the case in which two different kinds of signals may be present in the data. The two species are *sine waves* and *Lorentzians* named "sine" and "lor", respectively. The sinusoidal signals are described by parameters $\theta_{\rm sine} = \left(\log A_{\rm sin},\ \log f,\ \log \dot{f},\ \phi\right)$, their template is a sine wave with frequency $e^{\log f}$ that evolves in time with constant derivative $e^{\log \dot{f}}$,

$$s_{\rm sine}(t,\theta_{\rm sine}) = e^{\log A_{\rm sin}} \cos\left(2\pi e^{\log f} t + \pi e^{\log \dot{f}} t^2 + \phi\right) . \tag{40}$$

The Lorentzians are defined by parameters $\theta_{\rm lor} = \left(A_{\rm lor},\ w,\ t_0\right)$, their template is a Lorentzian centered at time t_0 and width w,

$$s_{\rm lor}(t,\theta_{\rm lor}) = A_{\rm lor} \frac{1}{1+\left(\frac{t-t_0}{w}\right)^2} . \tag{41}$$

As data, we generate a time series with duration $T_{\rm obs} = 0.1\,{\rm yr}$, with cadence for data collection $\Delta t = 500\,s$. In order to resolve the frequency of the simulated signals and observe few modulations of sinusoids, the frequency and its time derivative are constrained by $1/T_{\rm obs} \leq f \leq 1/2\Delta t$, $1/T^2_{\rm obs} \leq \dot{f} \leq 10/T^2_{\rm obs}$. In order to accurately reconstruct the widths of the Lorentzians, we also impose the condition $w \geq 5\Delta t$.

The data are given by

$$d(t) = \sum_{\alpha=\{\rm sine,lor\}} \sum_{i=1}^{N_{{\rm inj},\alpha}} h_{\alpha\,,i}(t) + n(t)\,, \tag{42}$$

where $h_{\alpha\,,i}(t)$ is the signal from the $i-$th injected source of type α, with $N_{\rm inj,sine} = 15$ and $N_{\rm inj,lor} = 5$, and $n(t)$ is the Gaussian noise realization in time domain with covariance $C(t)$. The corresponding log-likelihood of the data is a sum of independent one-dimensional Gaussian terms:

$$-2\log\mathcal{L} = \sum_t \left\{ \frac{[d(t)-h(t)]^2}{C(t)} + \log\left[2\pi C(t)\right] \right\} . \tag{43}$$

where $h(t) = \sum_{\alpha,i} h_{\alpha,i}(t)$. We assume known white noise with constant variance $C(t) = C = 10^{-45}$.

The priors adopted for the parameters of both classes of signals are listed in Table II. We remark that for both the number of sine waves and Lorentzians we use a uniform prior over all the positive integers. Furthermore, our inference starts from the initial point with $N_{\rm sine} = N_{\rm lor} = 0$.

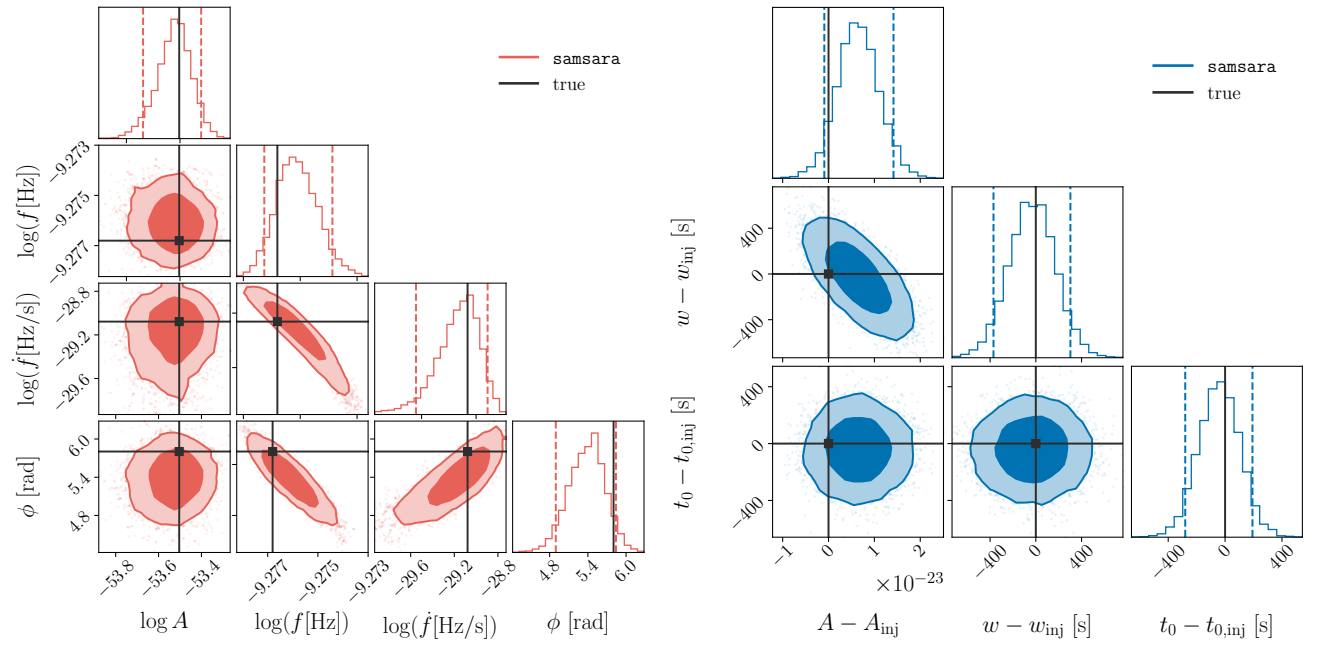

FIG. 5: Posterior of one of the injected sine waves (left panel) and one of the injected Lorentzians (right panel). The injected parameters are represented in solid dark gray.

For the mutation, we use adaptive Gaussian proposals, changing the widths accordingly to the signal-to-noise ratio (SNR) of the mutating source. We set $\mathcal{Z}(N_{\mathrm{pop},\alpha}, \theta_{j,\alpha}) = 1$ and $N_{\mathrm{gen}} = 10^7$, giving a wall time of approximately 15 hours on a single Intel Xeon Gold core.

Figure 4 shows the so-called trace plots for the number of sources per species as a function of the generations and the posterior on the number performed by `samsara`. The figure illustrates that the algorithm starts from an initial state with empty society, moves through a low posterior probability region and rapidly converges toward regions of high posterior density, effectively exploring the typical set. In the initial stages of the sampling, `samsara` typically tends to overfit the total signal by introducing an excess number of sources w.r.t. the true values. These excess sources are progressively removed in the later stages of the evolution. In zero noise, in which we consider the specific realization $n(t) = 0$, the numbers of reconstructed sine waves and Lorentzians exactly match the injected values. Conversely, for a generic noise realization, `samsara` tends to overfit the data with an excess number of sinusoids. This behavior may arise because the sinusoids are a basis, and the effect of the frequency derivative is not strong enough to prevent the algorithm from fitting the spurious contribution due to noise. However, the catalog reconstructed with the `PETRA` algorithm [30] described in Section III E contains 15 sources (the true number) with a probability of being in the catalog greater than 0.8 and 10 sources with a probability lower than 0.3. Therefore, in this case, `PETRA` is able to distinguish the injected sources from those fitting the noise, which `samsara` explores through the numerous birth and death moves that make N_{sine} oscillate in Figure 4. In Figure 5 we show the corner plots of a sine wave (left panel) and a resonance (right panel) reconstructed from the `samsara` samples using `PETRA`. All parameters for both signal types have been correctly recovered. In Figure 6, we plot the posterior over the total signal and its 90% C.I. from the sine wave plus Lorentzians society model. Also in this case, the injected signal is faithfully reconstructed.

C. Gaussian Mixture Model with unknown number of components

Finally, we apply `samsara` to a common problem in inference: clustering of data. Specifically, we focus on the determination of the number of components, and their parameters, in a GMM with an unknown number of components N. For the sake of simplicity and ease of comparison, we apply it to the case of uni-variate samples, although `samsara` can handle arbitrary dimensional cases. We model this problem as a single species society, named *mix*, where an individual is a Gaussian components of the mixture with parameters $\theta_{\mathrm{mix}} = \{\pi, \mu, \sigma^2\}$ and density

$$f_{\mathrm{mix}}(x|\theta_{\mathrm{mix}}) = \pi \mathcal{N}(x|\mu, \sigma^2)\,. \tag{44}$$

We simulate 1000 independent and identically distributed (i.i.d.) samples from the mixture density

$$p(x|\vec{\pi}, \vec{\mu}, \vec{\sigma}^2) = \sum_{i=1}^{3} \pi_i \mathcal{N}(x|\mu_i, \sigma_i^2)\,, \tag{45}$$

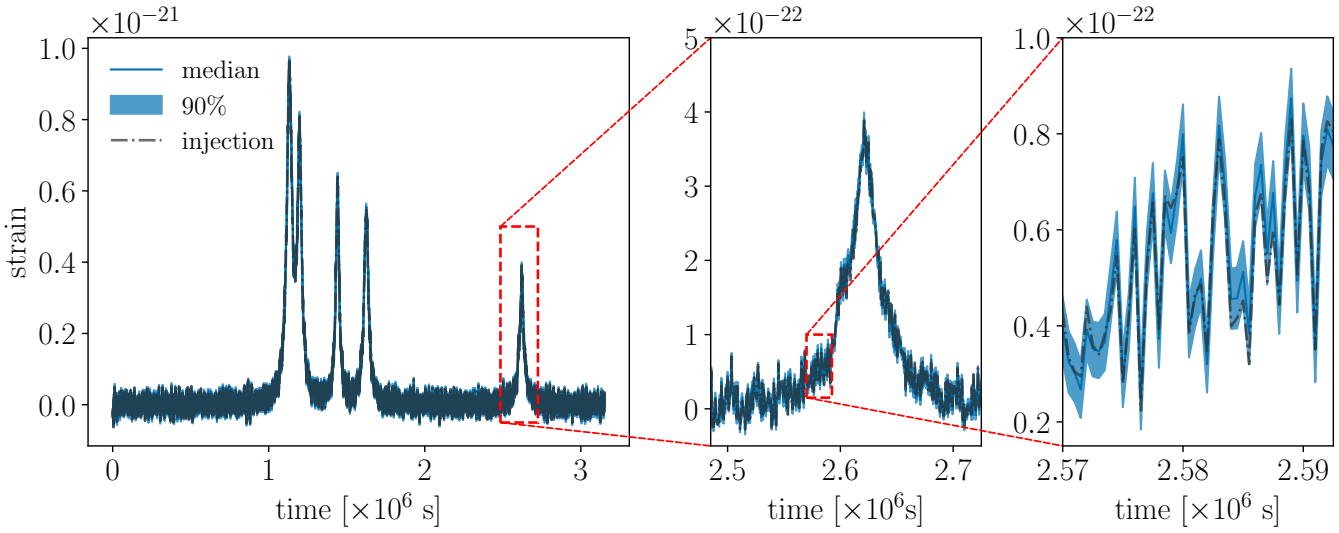

FIG. 6: Posterior on the total reconstructed signal (deep blue). The shaded area shows the 90% C.I., while the dash-dot line shows the injected total signal (dark gray).

with

$$\begin{aligned}\vec{\pi} &= \left\{\frac{1}{10}, \frac{6}{10}, \frac{3}{10}\right\} , \\ \vec{\mu} &= \{2, 3, 0\} , \\ \vec{\sigma} &= \{0.05, 1, 0.4\} .\end{aligned} \tag{46}$$

We set `samsara` for GMM problems, in which

$$\mathcal{Z}(N_{\mathrm{pop},\alpha}, \theta_{j,\alpha}) = \frac{1}{(N_{\mathrm{pop},\alpha}+1)(1-\pi_{j,\alpha})^{N_{\mathrm{pop},\alpha}}} , \tag{47}$$

and we mutate using the default blocked Gibbs sampler implementation for GMM. In this case, we choose to start with an initial population composed by one individual randomly extracted from the priors. The priors used are discussed again in Appendix E and Tabel II.

We then compare `samsara` results and performances with the product space prescription using the parallel Python implementation of the nested sampling (NS) `raynest` [31]. The results are summarized in Figures. 7 and 8.

The GMM posterior recovered by `samsara` accurately describes the samples drawn from the true distribution. Both the true distribution and the median corresponding to the most probable number of components inferred by the NS, lie within the 68% C.R. of the posterior and are fully consistent with the median value. Finally, the posterior on the number of components peaks at three, indicating that this is the most probable number of Gaussian components. Its behavior closely matches that of the Bayes' factor obtained from the product-space procedure.

V. DISCUSSION

We have presented `samsara`, a Continuous-Time Markov Chain Monte Carlo sampler for trans-dimensional Bayesian inference. Our framework models the parameter space in terms of a society, which is a collection of populations of different species that are lists of individuals with homogeneous properties, e.g., the number of parameters. The algorithm is based on the Poisson dynamics of birth-death-mutation processes, where species, process, and the transitions used to evolve the society are chosen according to the rates inferred from the proposed new states. A crucial point in our algorithm is that the death and birth rates, unless explicitly fixed by the user, are computed adaptively according to posterior values in the new states. While the mutation rate is fixed, birth and death rates are optimally balanced by the values of the posterior in the new states; the Poisson processes favor a change in the number of individuals in the society only if this provides a substantial gain in the posterior. This feature is particularly important, because, at each step, given the birth-death prescriptions, all the possible transitions are considered and the CTMCMC dynamics moves the chain towards the state of higher probability most of the time.

Our approach presents some advantages w.r.t. RJ-MCMC, where the changes in the number of models used to represent the observations are randomly proposed and accepted with probability given by the acceptance ξ_{RJ}, which can be very low. In our framework instead, the exploration of the posterior is self-controlled and optimized at each step by the Poisson process and a birth-death transition is always accepted. The acceptance in CTMCMC ξ_{CT} is typically much larger than ξ_{RJ}, because the birth of only a new individual is not affected by the

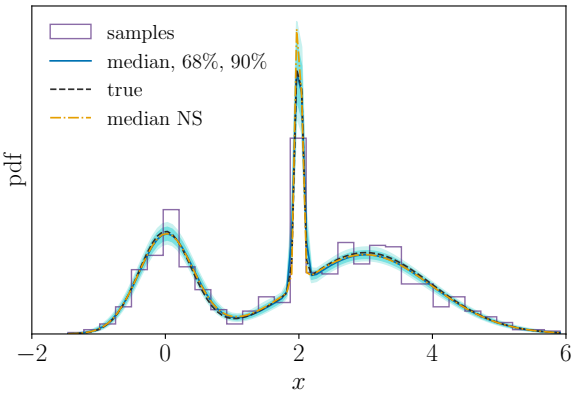

FIG. 7: Posterior reconstruction of the GMM both with `samsara` (median in solid deep blue and 68%, 90% C.I. in turquoise) and the `raynest` run corresponding to the maximum of the evidence (median in dashdot amber). Truth mixture model in dashed dark gray. The reconstructed GMMs are essentially indistinguishable.

curse of dimensionality. This means that to reach a state with $N_{\rm ind}$ the number of steps required in CTMCMC are approximately $N^{\rm CT}_{\rm steps} = N_{\rm ind}/\xi_{\rm CT}$, while for a RJ $N^{\rm RJ}_{\rm steps} = N_{\rm ind}/\xi_{\rm RJ}$ steps. Given that the acceptance of the RJ could be very low, especially for extremely highly-dimensional problems, it is very plausible that $N^{\rm RJ}_{\rm steps} \gg N^{\rm CT}_{\rm steps}$, meaning that we expect our framework to be more efficient in general. This is in stark contrast with the assertions in [32], where the authors find no efficiency improvements in CTMCMC compared to RJMCMC in very low dimensional problems.

In addition, our algorithm is capable of well exploring the trans-dimensional parameter space not only when starting from a state with a randomly generated society, but even when starting from an empty one. Moreover, it is able to do so with any kind of prior, proper or improper as the default one, being the posterior always proper [24].

`samsara` is a flexible and highly modular code, designed with a clear separation between the modules responsible for proposing new points, evaluating rates, and evolving the society according to the dynamics of the Poisson process. Moreover, the framework offers users the freedom to fully customize their Bayesian inference setup, by defining their own posterior, specifying the rate prescription, configuring custom proposal mechanisms, or even integrating an external sampler. Alternatively, `samsara` includes a variety of built-in methods that can be readily used and combined with user-defined components.

We tested the validity of our algorithm with three different illustrative cases that could be relevant both in Bayesian inference, physics, or other research fields. The first case involves sampling from a population whose number of individuals follows a Poisson distribution and whose two-dimensional parameters follow a bi-modal distribution with two probability phases and a sharp peak. The second problem is the joint analysis of sine waves and Lorentzians injected in a noisy time series, where the number and the parameters of both species are unknown. The third one concerns the reconstruction of the probability density function of a Gaussian mixture model with an unknown number of components. In all three cases, `samsara` successfully samples from the target distribution, accurately recovering both the posterior on the number of components and the posterior on their parameters. These results demonstrate the capability and reliability of `samsara`, even when compared with Bayes factor statistics obtained from the product-space approach using Nested Sampling.

Nonetheless, several aspects still require improvement and deeper understanding. In particular, we have seen that `samsara` sometimes can exhibits long autocorrelation lengths, so that we may lose a substantial number of iterations before obtaining statistically independent samples. For instance, this is the case of the test in Section IV B, but it is not for the GMM test of Section IV C where it is short. We think this arises mostly by the used proposals. Future work will focus on further investigating this issue, and exploring possible improvements which are: investigate the potential advantages or disadvantages of proposing jumps greater than one in the dimensionality of the parameter space, improve `samsara`'s performances through more sophisticated proposals schemes and more efficient Gibbs and Collapsed Gibbs samplers and study the effect of a deterministic drift for the mutation via HMC.

`samsara` is designed to tackle large inference problems in physics and astrophysics. As such, we begun to explore its capabilities to solve the inference over the large number of sources expected in future gravitational waves observatories such as LISA [10] and Einstein Telescope [33]. We plan to report our findings in future publications.

ACKNOWLEDGMENTS

The authors are grateful to A. Raimondi for valuable inputs. L. V. acknowledges financial support from the project "LISA Global Fit" funded by the ASI/Università di Trento Grant No. 2024-36-HH.0 - CUP F63C24000390001. RB acknowledges support from the ICSC National Research Center funded by NextGenerationEU, and the Italian Space Agency grant Phase B2/C activity for LISA mission, Agreement n.2024-NAZ-0102/PER.

The PhD fellowship of J.P. is funded by the Italian Ministry of University through the project 'Nano-Meta-Materials and Devices: New Frontier Concepts for Particle and Radiation Detection' (Grant 'Dipartimento di Eccellenza' 2023-2027, CUP I57G22000720004) at the Department of Physics of the University of Pisa.

Computations were performed using University of Birmingham BlueBEAR High Performance Computing facility and CINECA with allocations through INFN, and through EuroHPC Benchmark access call grant EHPC-

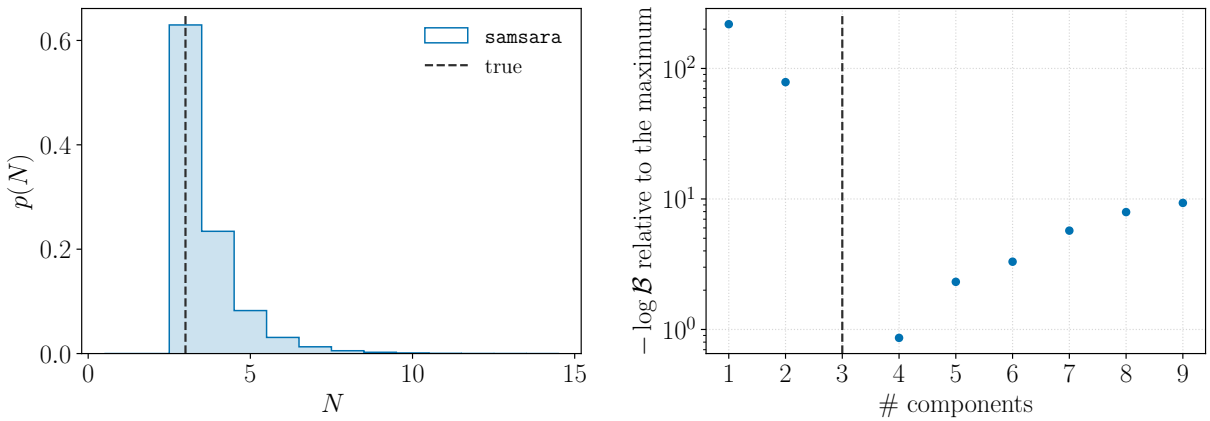

FIG. 8: Left panel: posterior distribution on the number of active components from the analysis with `samsara`. Right panel: negative logarithmic Bayes factor from the nested sampling runs on the product space relative to its maximum value. On both panels, the dashed line indicates the true number of components used to generate the data.

BEN-2025B08-042.

Data availability statement The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

Code availability statement The code/software generated during and/or analysed during the current study is available from the corresponding author on reasonable request.

Funded by SCOAP^3.

Appendix A: Rates derivation from detailed balance

In this appendix, we derive Eqs. (17), (18), (19) from the detailed balance equations, ensuring together with the ergodicity assumption, that the CT Markov Chain has the target posterior as stationary distribution. We will mainly follow here the computations of [13].

As shown in Eq. (1), Bayes' theorem allows to write the target posterior measure, $P(\cdot \mid D)$, as a function of the likelihood, $P(D|y)$, the evidence, $p(D)$, and the prior measure, $\Pi(\cdot)$,

$$\mathrm{d}P(y|D) = \frac{p(D|y)}{p(D)}\mathrm{d}\Pi(y) \equiv f(y)\mathrm{d}\Pi(y)\,, \tag{A1}$$

where y lives in the space E defined in Section II A and we have defined $f(y)$ as the ratio between the likelihood and evidence.

As discussed in Section II D, it is possible to sample the posterior from the conditional distribution of a single species α. We therefore derive the condition on the rates in this case, fixing the species different from α in the CTMCMC at a given step. For clarity of notation, we use the same conventions of Section II B, defining $y_\alpha \in F \subset E_{\alpha,N_{\mathrm{pop},\alpha}}$ and $y'_\alpha = y_\alpha \cup \theta_\alpha \in G \subset E_{\alpha,N_{\mathrm{pop}}+1}$. To highlight the quantities that are conditioned in this computation, we define $y_{-\alpha} \equiv y \setminus y_\alpha$. To simplify the notation we use the compact notation for the restrictions of the posterior and prior measures on $E_{\alpha,N_{\mathrm{pop},\alpha}}$, via

$$\begin{aligned}\mathrm{d}\mu(y_\alpha) &\equiv \mathrm{d}P(y_\alpha|y_{-\alpha}\,D)\,,\\ \mathrm{d}\nu(y_\alpha) &\equiv \mathrm{d}\Pi(y_\alpha|y_{-\alpha})\,.\end{aligned} \tag{A2}$$

It is useful to note that Bayes' theorem allows us to write

$$\int_{\mathcal{E}} \mathrm{d}\mu(y_\alpha) = \int_{\mathcal{E}} \mathrm{d}\nu(y_\alpha)\, f(y|y_{-\alpha})\,, \tag{A3}$$

where we have explicitly written that f is conditioned on the value of $y_{-\alpha}$.

For a point process on $E_{\alpha,N_{\mathrm{pop}}}$, where the labeling of the individuals is not taken into account, we can write the relation between the prior measure and its density π as

$$\mathrm{d}\nu(y_\alpha) = \pi(y_\alpha)\mathrm{d}\theta_{1,\alpha}\cdots\mathrm{d}\theta_{N_{\mathrm{pop},\alpha},\alpha}\,, \tag{A4}$$

where the number dependence is omitted, being clear by $y_\alpha \in E_{\alpha,N_{\mathrm{pop},\alpha}}$.

For what follows, it is useful to note that

$$\begin{aligned}\pi(y_\alpha)\mathrm{d}\nu(y'_\alpha) &= \pi(y_\alpha)\pi(y'_\alpha)\prod_{i=1}^{N_{\mathrm{pop},\alpha}}\mathrm{d}\theta_{i,\alpha}\mathrm{d}\theta_\alpha\\ &= \pi(y'_\alpha)\mathrm{d}\nu(y_\alpha)\mathrm{d}\theta_\alpha\,.\end{aligned} \tag{A5}$$

To recover the rates from the detailed balance equations, we write the left-hand side (LHS) and the right-hand side (RHS) of Eqs. (14), (15) explicitly, substituting the transition kernels of Eqs. (12), (13) and integrating on the same integration domain. For simplicity, we focus just on (14).

The LHS can be written as

$$\begin{aligned}\mathrm{LHS} &= \int_F \mathrm{d}\mu(y_\alpha)\, R_{b,\alpha}(y_\alpha)\\ &= \int_F \mathrm{d}\nu(y_\alpha)\, f(y_\alpha|y_{-\alpha}) R_{b,\alpha}(y_\alpha)\\ &= \int_F \mathrm{d}\nu(y_\alpha)\, f(y_\alpha|y_{-\alpha}) \int_{\Theta_\alpha} \mathrm{d}\theta'_\alpha\, R_{b,\alpha}(y,\theta'_\alpha) h(\theta'_\alpha|y_\alpha)\\ &= \int_F \int_{\Theta_\alpha} \mathrm{d}\nu(y_\alpha)\mathrm{d}\theta'_\alpha\, f(y_\alpha|y_{-\alpha})\, R_{b,\alpha}(y,\theta'_\alpha) h(\theta'_\alpha|y_\alpha)\,,\end{aligned} \tag{A6}$$

The RHS can then be written as

$$\begin{aligned}\mathrm{RHS} &= \int_{\mathcal{E}_{+1}} \mathrm{d}\mu(z_\alpha) \sum_{\theta_{j,\alpha}\in z\,:\, z\,\setminus\,\theta_{j,\alpha}\in F} R_{j,d,\alpha}(z,\theta_{j,\alpha})\\ &= \int_{\mathcal{E}_{+1}} \mathrm{d}\mu(z_\alpha) \sum_{j=1}^{N_{\mathrm{pop},\alpha}+1} R_{j,d,\alpha}(z,\theta_{j,\alpha}) \mathbb{1}\{z_\alpha \setminus \theta_{j,\alpha} \in F\}\\ &= \int_{\mathcal{E}_{+1}} \mathrm{d}\nu(z_\alpha)\, f(z_\alpha|z_{-\alpha})\\ &\qquad \sum_{j=1}^{N_{\mathrm{pop},\alpha}+1} R_{j,d,\alpha}(z,\theta_{j,\alpha}) \mathbb{1}\{z_\alpha \setminus \theta_{j,\alpha} \in F\}\\ &= \int_{\mathcal{E}_{+1}} \mathrm{d}\nu(z_\alpha)\, (N_{\mathrm{pop},\alpha}+1) f(z_\alpha|z_{-\alpha})\\ &\qquad R_{j,d,\alpha}(z,\theta_{j,\alpha}) \mathbb{1}\{z_\alpha \setminus \theta_{j,\alpha} \in F\}\\ &= \int_F \int_{\Theta_\alpha} \mathrm{d}\nu(y_\alpha)\mathrm{d}\theta'_\alpha\, \frac{\pi(y'_\alpha)}{\pi(y_\alpha)}(N_{\mathrm{pop},\alpha}+1)\\ &\qquad f(y'|y_{-\alpha}) R_{j,d,\alpha}(y',\theta'_\alpha)\,,\end{aligned} \tag{A7}$$

with $\mathcal{E}_{+1}$ the compact notation for $E_{\alpha,N_{\mathrm{pop},\alpha}+1}$ introduced in Section II D, and $y'_\alpha \equiv y_\alpha \cup \theta'_\alpha$. To get the expressions in the second, fourth, and sixth rows, we have used Eq. (A3), (A4), and (A5). The factor $N_{\mathrm{pop},\alpha}+1$ in the fifth row comes from the symmetry of the prior under the exchange of two individuals. Equating the LHS and RHS, we get

$$\begin{aligned}&p(y_\alpha|y_{-\alpha}\,D) R_{b,\alpha}(y,\theta'_\alpha) h(y,\theta'_\alpha)\\ &\quad = (N_{\mathrm{pop},\alpha}+1) p(y'_\alpha|y_{-\alpha}\,D) R_{j,d,\alpha}(y',\theta'_\alpha)\,,\end{aligned} \tag{A8}$$

with $R_{b,\alpha}(y,\theta'_\alpha), R_{j,d,\alpha}(y',\theta'_\alpha)$ being the unknowns. By repeating the calculation for Eq. (15) similarly, we recover again Eq. (A8). Say the chain is in state y, if we fix $R_{b,\alpha}(y,\theta'_\alpha) = R_{b,\alpha}(y_\alpha)$, we have a closed form for $R_{j,d,\alpha}$ and we find Eq. (17). Instead, if both rates are free to vary, we find Eqs. (18), (19).

The factor $\mathcal{Z}(N_{\mathrm{pop},\alpha},\theta_{j,\alpha})$ introduced in Eq. (17) depends on the relation between the prior measure and the prior density. In this case, where we consider a point

process with a prior measure given by Eq. (A4), we have $\mathcal{Z}(N_{\mathrm{pop},\alpha}, \theta_{j,\alpha}) = 1$. However, such a factor could change when the label of the individuals has to be taken into account and the parameters in the states are ordered. For example, this applies to GMM, since the weights of the components must live in the simplex. In fact, $N_{\mathrm{pop},\alpha} - 1$ individuals (mixture components) can permute, but the remaining individual is completely determined by the others.

In GMM, we can represent the state as a marked point process [13] on E, where the individuals are written as $\theta_\alpha = (w_\alpha, \phi_\alpha)$, with w_α the weight of the component and $\phi_\alpha = \{\mu_\alpha, \Sigma_\alpha\}$ its parameters.

In this case, Eq. (A4) becomes

$$\mathrm{d}\nu(y_\alpha) = (N_{\mathrm{pop},\alpha} - 1)!\, \pi(y_\alpha) \prod_{j=1}^{N_{\mathrm{pop},\alpha}-1} \mathrm{d}w_j \mathrm{d}\phi_{j,\alpha} d\phi_{N_{\mathrm{pop},\alpha}} \,. \tag{A9}$$

Similarly, we get

$$\mathrm{d}\nu(y'_\alpha) = \frac{\pi(y'_\alpha)}{\pi(y_\alpha)} N_{\mathrm{pop},\alpha} \mathrm{d}\nu(y_\alpha) \mathrm{d}\phi'_\alpha \mathrm{d}w'_\alpha \,. \tag{A10}$$

By repeating the computations done in this appendix keeping into account this extra factor in the prior measure and the fact that the components live in the simplex, we find Eqs. (17), (18), (19) with $\mathcal{Z}(N_{\mathrm{pop},\alpha}, \theta_{j,\alpha})$ computed as in Eq. (47).

Appendix B: Rao-Blackwellization of estimators in CTMCMC

1. Rao-Blackwell theorem

In this appendix, we follow the enunciation of the Rao-Blackwell theorem given in [23]. Let X be a random observable, distributed according to $p_\theta(X = x)$, and be f a function of θ to estimate. Let then $\delta(X)$ be an estimator for f with finite risk,

$$R(\theta, \delta) = \mathbb{E}_X \left[L\left(\theta, \delta(X)\right) \right] \,, \tag{B1}$$

with L a convex loss function.

T is said to be a sufficient statistic for X if the conditional distribution of X given T is independent of θ. The Rao-Blackwell theorem states that if T is a sufficient statistic for p_θ, then the Rao-Blackwellized estimator

$$\eta(T) \equiv \mathbb{E}_T \left[\delta(X) | T \right] \tag{B2}$$

satisfies

$$R(\theta, \eta) \leq R(\theta, \delta) \,, \tag{B3}$$

with the inequality saturated if $\delta(X) = \eta(T)$ with probability one. In [22] the Rao-Blackwellization of estimators has been applied to some sampling schemes, like the Metropolis-Hastings and Accept-Reject algorithm.

2. Rao-Blackwellization with the waiting times

In CTMCMC, the lifetime of a state is distributed according to an exponential distribution,

$$p\left(\Delta t_i | \tau(y_i)\right) = \frac{1}{\tau(y_i)} e^{-\Delta t_i / \tau(y_i)} \,, \tag{B4}$$

where we have defined

$$\Delta t_i \equiv \min_{j,p,\alpha} t_{j,p,\alpha} \,, \tag{B5}$$

where the $t_{j,p,\alpha}$ are drawn from exponential distributions with average $1/R_{j,p,\alpha}(y_i)$. In this case, y_i is a sufficient statistic for Δt_i, because the knowledge of y_i completely determines the exponential distribution, therefore the Rao-Blackwellized estimator of Eq. (25) is obtained by taking the expectation value of Δt_i given y_i, i.e., given $\tau(y_i)$, finding Eq. (26).

Appendix C: Proposals

Currently, `samsara` includes only a few built-in proposal distributions. Two of them are standard: the prior distribution and a Gaussian displacement. Given the starting individual with parameters θ_0, the prior-based proposal reads

$$q_{\mathrm{prior}}(\theta_t | \theta_0) = \pi(\theta_t) \,, \tag{C1}$$

the Gaussian displacement is

$$q_{\mathrm{Gaussian}}(\theta_t | \theta_0) = \mathcal{N}(\theta_t | \mu = \theta_0, \sigma) \,. \tag{C2}$$

For instance, the default proposal for a birth move is the prior proposal defined in Eq. (C1). In the GMM case, the default birth proposal is the prior-based following the mutations' Gibbs conjugate priors: Normal-Inverse-Wishart (NIW) for the means and covariances, and Beta distribution for the weight

$$\begin{aligned} h(\mu', \Sigma', \pi'_\alpha) = \mathrm{NIW}(\mu', \Sigma' | \bar{D}, k, \mathrm{Cov}(kD), \nu) \times \\ \times \mathrm{Beta}(\pi'_\alpha | 1, N_{\mathrm{pop},\alpha}) \,, \end{aligned} \tag{C3}$$

where

$$\begin{aligned} &\mathrm{NIW}(\mu, \Sigma | \mu_0, k, \Lambda, \nu) = \mathcal{N}\left(\mu | \mu_0, \Sigma/k\right) \mathcal{W}^{-1}(\Sigma | \Lambda, \nu) \,, \\ &\mathcal{W}^{-1}(\Sigma | \Lambda, \nu) = \frac{\det(\Lambda)^{\nu/2}}{2^{\nu M/2} \Gamma_M(\nu/2)} \times \\ &\qquad \times \det(\Sigma)^{-(\nu+M+1)/2} \exp\left\{ -\frac{1}{2} \mathrm{tr}\left(\Lambda \Sigma^{-1}\right) \right\} \,, \\ &\mathrm{Beta}(\pi | \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} \pi^{\alpha-1} (1 - \pi)^{\beta-1} \,, \end{aligned} \tag{C4}$$

and Γ_M the multivariate gamma function.The default choice is $k = 1/5$ and $\nu = M + 2$, with M the dimension

of a single data sample in the observations D and $\bar{D}$ is the mean of D.

Another possible proposal is more peculiar and genetically inspired. In analogy with biological mitosis, we propose a new individual in the society with parameters θ_t, obtained by applying a multiplicative Gaussian shift to the parameters of another individual θ_0 in the society,

$$q_{\rm Mitosis}(\theta_t|\theta_0) = \mathcal{N}\left(\theta_t|\theta_0, \xi\theta_0\right) . \tag{C5}$$

The strength ξ is chosen according to the specific problem considered and may vary across different parameters. In addition, we introduce a *mutation rate* which sets the probability that a component θ_t remains unaffected by the mutation. In the current version of `samsara`, the individual θ_0 involved is drawn randomly from the population.

Appendix D: Diagnostic

1. Correlation length

According to the discussion in Section III D, to overcome the issues related to the trans-dimensional nature of the problem, we introduce a scalar function ρ to evaluate the autocorrelation function of the chain. In this work, for simplicity, we consider ρ to be the log-posterior in a given generation, but other choices are possible, such as the number of sources per species. Once the scalar quantity has been chosen, the autocorrelation function is computed according to

$$\mathrm{ACF}(\rho,\delta) = \frac{N_{\rm gen}}{N_{\rm gen}-\delta} \frac{\sum_{i=1}^{N_{\rm gen}-\delta}\left(\rho^{(i)}-\bar{\rho}\right)\left(\rho^{(i+\delta)}-\bar{\rho}\right)}{\sum_{i=1}^{N_{\rm gen}}\left(\rho^{(i)}-\bar{\rho}\right)^2} , \tag{D1}$$

where $\bar{\rho}$ is the average value of the scalar function,

$$\bar{\rho} \equiv \frac{1}{N_{\rm gen}} \sum_{i=1}^{N_{\rm gen}} \rho^{(i)} . \tag{D2}$$

A conservative assumption for the computation of the autocorrelation time is to choose $\delta_{\rm corr}$ as the location of the fifth zero of the autocorrelation function. In Eq. (D1) we have neglected the impact of the waiting times introduced in Section II E. Although waiting times are properly taken into account in post-processing, their contribution to the correlation length is negligible. Including them in the computation would generate a nontrivial mode coupling in the correlations at different lags δ, resulting in a complicated expression for the correlation time, which would slow down the code considerably.

2. Convergence

In this section, we discuss the convergence tests within `samsara`, which are based on [28, 29].

The problem of quantifying the convergence of trans-dimensional MCMC consists of comparing samples from spaces with different dimensions. This is analogous to the problems encountered in defining an auto-correlation length, as discussed in Section III D. Also in this case, the idea is to construct an *ad hoc* scalar [28] used to compare multiple chains of `samsara`.

We consider C independent chains, obtained for instance by using different sampler seeds or initial conditions. Within our framework, for a given species α, the m-th sample of the trans-dimensional sample path related of the chain c, is a population with $N_{{\rm pop},\alpha}$ individuals[12] denoted by $y^{(m)}_{\alpha,c}$. Each individual has parameters $\theta^{(m)}_{j,\alpha,c} \in \Theta_\alpha \subset \mathbb{R}^{n_\alpha}$.

We then consider a set of reference points $\{v_{\alpha,\ell} : \ell = 1,\ldots,N_v,\ v_{\alpha,\ell} \in \mathcal{V}_\alpha \subset \Theta_\alpha\}$, and construct scalar quantities to compare different chains. The scalar[13] we adopt here is the minimum distance of an individual in the society from a chosen reference point

$$x^{(m)}_{\alpha,c,\ell} = \min_{i=1,\ldots,N_{{\rm pop},\alpha}} \left\{ ||\theta^{(m)}_{i,\alpha,c} - v_{\alpha,\ell}|| \right\} , \tag{D3}$$

with $||\cdot||$ the Euclidean norm. Starting from $x^{(m)}_{\alpha,c,\ell}$, it is possible to define the *empirical Cumulative Mass Function* (empirical CMF) for a given reference point $v_{\alpha,\ell}$ as

$$F^c_\alpha(x|v_{\alpha,\ell}) = \frac{1}{N'_{\rm gen}} \sum_{m=1}^{N'_{\rm gen}} \mathbb{1}\{x^{(m)}_{\alpha,c,\ell} \le x\} , \tag{D4}$$

where $N'_{\rm gen}$ is the number of thinned chain samples.

With these two quantities three convergence indicators were constructed in [28]: the *pairwise comparison*, the *Monte Carlo test*, and the *potential scale reduction factor (PSRF)*. If $||\cdot||_p$ denotes the L^p-norm, the pairwise comparison between two chains reads

$$u^{c_1c_2}_\alpha \approx \frac{1}{N_v} \sum_{\ell=1}^{N_v} ||F^{c_1}_\alpha(x|v_{\alpha,\ell}) - F^{c_2}_\alpha(x|v_{\alpha,\ell})||_p . \tag{D5}$$

If $C \ge 3$, an overall distance measure can be defined by averaging the pairwise comparison between all the couples,

$$u_\alpha = \binom{C}{2}^{-1} \sum_{c_1=1}^{C} \sum_{c_1\neq c_2} u^{c_1c_2}_\alpha . \tag{D6}$$

A convergence criterion for the chains is $u_\alpha \to 0$.

For the MC test, we define the quantity

$$w^c_\alpha \approx \frac{1}{N_v} \sum_{\ell=1}^{N_v} ||F^c_\alpha(x|v_{\alpha,\ell}) - \bar{F}^c_\alpha(x|v_{\alpha,\ell})||_p, \tag{D7}$$

[12] Ref. [28] refers to individuals as components.

[13] With *scalar*, we mean here a quantity independent on the number of society individuals.

where

$$\bar{F}^c_\alpha(x|v_{\alpha,\ell}) = \frac{1}{N'_{\rm gen}-1}\sum_{k\neq c} F^k_\alpha(x|v_{\alpha,\ell}). \tag{D8}$$

In this case, chains have converged if $w^c_\alpha \to 0$.

The third convergence indicator involves the Gelman-Rubin diagnostic [29], using $x^{(m)}_{\alpha,c,\ell}$ as the scalars to be monitored for each $v_{\alpha,\ell}$. In this case, we define the PSRF as

$$\hat{R}_{v_{\alpha,\ell}} = \left(\frac{\frac{N'_{\rm gen}-1}{N'_{\rm gen}}W_{v_{\alpha,\ell}} + \frac{1}{N'_{\rm gen}}B_{v_{\alpha,\ell}}}{W_{v_{\alpha,\ell}}}\right)^{1/2}, \tag{D9}$$

where

$$\begin{aligned} B_{v_{\alpha,\ell}} =& \frac{N'_{\rm gen}}{C-1}\sum_{c=1}^{C}(\bar{x}_{\alpha,c,\ell}-\bar{x}_{\alpha,\ell})^2, \\ W_{v_{\alpha,\ell}} =& \frac{1}{C}\sum_{c=1}^{C} s^2_{\alpha,c,\ell} \end{aligned} \tag{D10}$$

with

$$\begin{aligned} \bar{x}_{\alpha,c,\ell} =& \frac{1}{N'_{\rm gen}}\sum_{m=1}^{N'_{\rm gen}} x^{(m)}_{\alpha,c,\ell}, \\ \bar{x}_{\alpha,\ell} =& \frac{1}{C}\sum_{c=1}^{C}\bar{x}_{\alpha,c,\ell} \\ s^2_{\alpha,c,\ell} =& \frac{1}{N'_{\rm gen}-1}\sum_{m=1}^{N'_{\rm gen}}(x^{(m)}_{\alpha,c,\ell}-\bar{x}_{\cdot,c,\ell})^2. \end{aligned} \tag{D11}$$

For $N'_{\rm gen}\to\infty$, the convergence is reached in each chain if $\hat{R}_{v_{\alpha,\ell}}\to 1$.

There is no prescription on the number of reference points that should be used for the convergence test. In principle, this choice should depend on the problem considered, but in [28] $N_v \approx 100$ is suggested as a sufficient number of points. For the actual values of points chosen, we randomly extract N_v from points of the different chains.

a. *Application to the analytic test case*

In Figure 9, we show the PSRF test applied to the analytic case of Section IV A. To compute the PSRF defined in Eq. (D9), we collect samples of the distribution (35) from 5 independent chains, obtained by initializing `samsara` with different seeds. After post-processing and thinning the samples as described in Section III E, we randomly extract 30 reference points per chain (150 total), computing the PSRF defined in Eq. (D9) per each point. To be conservative, we estimate the convergence by looking at the deviation from 1 of the maximum of the PSRF over the ensemble points. Figure 9 clearly shows that, after the burn-in cut performed, the chains are already consistent with the PSRF test being below 0.3% deviation from 1.

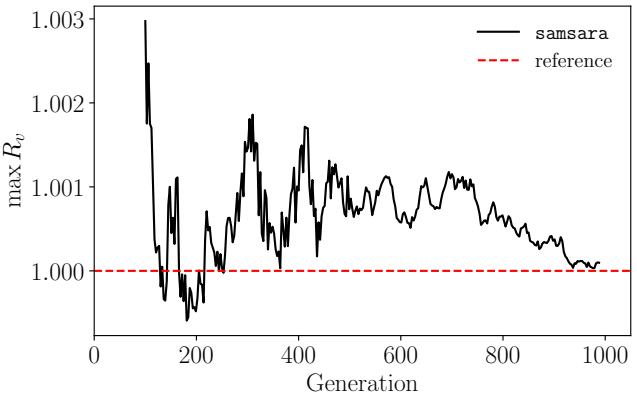

FIG. 9: Plot of the maximum of the PSRF (solid black line), defined in Eq. (D9), as a function of the generation for the analytic test case discussed in Section IV A. The dashed red line shows the reference value of $\hat{R}_v$ used as a criterion for the convergence.

Appendix E: Priors

In all the test cases discussed in Sections IV A, IV B, and IV C, we adopted an implicit unbounded prior over the number, with each number being equally-likely. Hence, we are fully agnostic on the number of sources, and the algorithm is free to explore any possible dimension of the parameter space. This emphasizes the efficiency of our algorithm in reconstructing the true number of individuals. In Table II, we list all the priors used in this work.

For the analytic case, we choose a uniform prior in θ_1, θ_2 large enough to contain all the Gaussian distributions in the mixture.

For the sine waves, in order to resolve the frequency and observe few modulations of sinusoids, the frequency and its time derivative are constrained by $1/T_{\rm obs} \leq f \leq 1/2\Delta t$ and $1/T^2_{\rm obs} \leq \dot{f} \leq 10/T^2_{\rm obs}$. For the Lorentzians, the prior bounds are chosen such that the peaks lie within the observation window and $w \geq 5\Delta t$, ensuring that each bump includes a sufficient number of sampling points.

For the GMM, we maintain faith to the blocked Gibbs sampler for the mutation using the Normal-Inverse-Gamma (NIG) for means and covariances, and the Dirichlet distribution (Dir) for the weights, being conjugate priors. Here, the NIG is the one-dimensional equivalent of the Normal-Inverse-Wishart (NIW),

$$\Gamma^{-1}(\sigma^2|a,b) = \frac{b^a}{\Gamma(a)}\frac{e^{-b/\sigma^2}}{(\sigma^2)^{a+1}}, \tag{E1}$$

with Γ the gamma function. We set the NIG hyperparameters to be their default initialization of the Gibbs sampler implemented in `samsara` as already presented in Section C. We set the concentration parameter of the Dirichlet to be $1/N_{\rm pop,mix}$ for each weight, with $N_{\rm pop,mix}$ the number of individuals (components) in the mixture.

Test case	Parameter	Prior
Analytic	N	improper unbound uniform on integers
	θ_1	$\mathcal{U}(-5, 4)$
	θ_2	$\mathcal{U}(-8, 4)$
Sinusoids and Lorentzians	$N_{\rm sine}$	improper unbound uniform on integers
	$\log A_{\rm sine}$	$\mathcal{U}(-54.5, -52.5)$
	$\log(f/{\rm Hz})$	$\mathcal{U}(\log(3\times10^{-5}), \log(10^{-3}))$
	$\log\left(\dot{f}/({\rm Hz/s})\right)$	$\mathcal{U}(\log(10^{-13}), \log(10^{-12}))$
	ϕ	$\mathcal{U}(0, 2\pi)$
	$N_{\rm lor}$	$\mathcal{U}(0, \infty)$
	$A_{\rm lor}$	$\mathcal{U}(10^{-22}, 10^{-20})$
	$w\,[s]$	$\mathcal{U}(10^4, 2\times10^4)$
	$t_0\,[s]$	$\mathcal{U}(0, T_{\rm obs})$
Gaussian Mixture Model	N	improper unbound uniform on integers
	π	${\rm Dir}(1/N_{\rm pop,mix}, \ldots, 1/N_{\rm pop,mix})$
	μ	$\mathcal{N}(\bar{D}, \sigma/k)$
	σ	$\Gamma^{-1}({\rm Cov}(D), \nu)$

TABLE II: Prior range of the parameters for the three different test cases considered in Section IV A, IV B, and IV C.

[1] E. T. Jaynes, *Probability Theory: The Logic of Science*, edited by G. L. Bretthorst (Cambridge University Press, Cambridge, UK, 2003).
[2] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, Journal of Chemical Physics **21**, 1087 (1953).
[3] W. K. Hastings, Biometrika **57**, 97 (1970).
[4] C. W. Gardiner, *Handbook of stochastic methods for physics, chemistry and the natural sciences*, 3rd ed., Springer Series in Synergetics, Vol. 13 (Springer-Verlag, Berlin, 2004) pp. xviii+415.
[5] R. M. Neal, Mcmc using hamiltonian dynamics, in *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. Jones, and X.-L. Meng (Chapman & Hall/CRC, New York, 2011) pp. 113–162.
[6] R. D. Gupta and D. S. P. Richards, International Statistical Review **69**, 433 (2001).
[7] C. Rasmussen, in *Advances in Neural Information Processing Systems*, Vol. 12, edited by S. A. Solla, T. K. Leen, and K.-R. Müller (MIT Press, 1999) pp. 554–560.
[8] P. J. Green, Biometrika **82**, 711 (1995).
[9] B. P. Carlin and S. Chib, Journal of the Royal Statistical Society. Series B (Methodological) **57**, 473 (1995).
[10] M. Colpi *et al.*, Lisa definition study report (2024), arXiv:2402.07571 [astro-ph.CO].
[11] N. J. Cornish and J. Crowder, Phys. Rev. D **72**, 043005 (2005), arXiv:gr-qc/0506059.
[12] M. Vallisneri, Class. Quant. Grav. **26**, 094024 (2009), arXiv:0812.0751 [gr-qc].
[13] M. Stephens, Annals of Statistics **28**, 40 (2000).
[14] R. Mohammadi, M. Pratola, and M. Kaptein, Continuous-time birth-death mcmc for bayesian regression tree models (2020), arXiv:1904.09339 [stat.ML].
[15] J. S. Liu, F. Liang, and W. H. Wong, Journal of the American Statistical Association **95**, 121 (2000).
[16] S. Geman and D. Geman, IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-6**, 721 (1984).
[17] J. Goodman and J. Weare, Communications in Applied Mathematics and Computational Science **5**, 65 (2010).
[18] D. Foreman-Mackey, D. W. Hogg, D. Lang, and J. Goodman, Publications of the Astronomical Society of the Pacific **125**, 306 (2013).
[19] C. Preston, Advances in Applied Probability **7**, 465–466 (1975).
[20] D. Blackwell, The Annals of Mathematical Statistics **18**, 105 (1947).
[21] C. R. Rao, Information and the accuracy attainable in the estimation of statistical parameters, in *Breakthroughs in Statistics: Foundations and Basic Theory*, edited by S. Kotz and N. L. Johnson (Springer New York, New York, NY, 1992) pp. 235–247.
[22] G. Casella and C. P. Robert, Biometrika **83**, 81 (1996).
[23] E. L. Lehmann and G. Casella, *Theory of Point Estimation*, 2nd ed. (Springer, New York, 1998).
[24] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian Data Analysis*, 3rd ed., Chapman & Hall/CRC Texts in Statistical Science Series (CRC, Boca Raton, Florida, 2013).
[25] M. Karamanis and F. Beutler, arXiv e-prints , arXiv:2002.06212 (2020), arXiv:2002.06212 [stat.ML].
[26] M. Karamanis, F. Beutler, and J. A. Peacock, MNRAS **508**, 3589 (2021), arXiv:2105.03468 [astro-ph.IM].
[27] N. Cornish and T. Robson, J. Phys. Conf. Ser. **840**, 012024 (2017), arXiv:1703.09858 [astro-ph.IM].
[28] S. A. Sisson and Y. Fan, Statistics and Computin **17**, 357 (2007).
[29] A. Gelman and D. B. Rubin, Statistical Science **7**, 457 (1992).
[30] A. D. Johnson, J. Roulet, K. Chatziioannou, M. Vallisneri, C. G. Trejo, and K. A. Gersbach, Phys. Rev. D **112**, 024045 (2025), arXiv:2502.14818 [gr-qc].
[31] W. Del Pozzo and J. Veitch, raynest: Parallel nested

sampling based on ray, Astrophysics Source Code Library, record ascl:2405.003 (2024), ascl:2405.003.

[32] O. Cappe, R. Christian P, and T. Ryden, *Reversible Jump MCMC Converging to Birth-and-Death MCMC and More General Continuous Time Samplers*, Working Papers 2001-29 (Center for Research in Economics and Statistics, 2001).

[33] M. Maggiore, M. Branchesi, A. Ghosh, *et al.*, Journal of Cosmology and Astroparticle Physics (3), 050, arXiv:2003.01113 [astro-ph.IM].