

The dynamical memory of tidal stellar streams

Joint inference of the Galactic potential and the progenitor of GD-1 with flow matching

Giuseppe Viterbo^{1,2,*} and Tobias Buck^{1,2}

¹ Universität Heidelberg, Interdisziplinäres Zentrum für Wissenschaftliches Rechnen (IWR), Im Neuenheimer Feld 205, 69120 Heidelberg, Germany

² Universität Heidelberg, Zentrum für Astronomie, Institut für Theoretische Astrophysik, Albert-Ueberle-Straße 2, 69120 Heidelberg, Germany

Received 2 December 2025 / Accepted 18 February 2026

ABSTRACT

Context. Stellar streams offer one of the most sensitive probes of the Milky Way’s gravitational potential, as their phase-space morphology encodes both the tidal field of the host galaxy and the internal structure of their progenitors. In this work, we introduce a framework that leverages flow matching and simulation-based inference (SBI) to jointly infer the parameters of the GD-1 progenitor and the global properties of the Milky Way potential.

Aims. Our aim is to move beyond traditional techniques (e.g., orbit-fitting and action-angle methods) by constructing a fully Bayesian likelihood-free posterior over host galaxy parameters and progenitor properties, thereby capturing the intrinsic coupling between tidal stripping dynamics and the underlying potential.

Methods. To achieve this, we generated a large suite of mock GD-1-like streams using our differentiable N -body code `ODISSEO`, sampling self-consistent initial conditions from a Plummer sphere and evolving them in a flexible Milky Way potential model. We then applied conditional flow matching to learn the vector field that transports a base Gaussian distribution into the posterior $p(\theta | d)$, enabling efficient amortized inference directly from stream phase-space data.

Results. We demonstrate that our method successfully recovers the true parameters of a fiducial GD-1 simulation, producing well-calibrated posteriors and accurately reproducing parameter degeneracies arising from progenitor–host interactions. Our results highlight the power of modern generative models for dynamical inference and provide a scalable pathway toward jointly constraining Galactic structure and the origins of stellar streams.

Conclusions. Flow matching provides a powerful, flexible framework for Galactic archaeology. Our approach enables joint inference on progenitor and Galactic parameters, capturing complex dependencies that are difficult to model with classical likelihood-based methods. This work paves the way for fully simulation-driven dynamical inference using Gaia and upcoming surveys.

Key words. methods: data analysis – methods: numerical – methods: statistical – Galaxy: kinematics and dynamics – Galaxy: structure

1. Introduction

Assuming a hierarchical Lambda cold dark matter accretion history, the Galactic halo should be populated by tidal debris from accreted satellites, such as dwarf galaxies and star clusters (e.g., Buck et al. 2019, 2020). As described in Binney & Tremaine (2008), when the tidal forces acting on these objects are strong enough, stars are pulled out and end up on orbits that have slightly more or less energy than the progenitor’s orbit, respectively forming the so-called leading and trailing arms. These narrow structures, referred to as stellar streams, have proven to be excellent tracers of fundamentals unknown in Galaxy evolution, and they have been used to probe the dark matter halo in external galaxies (Walder et al. 2025) and our Galaxy (Küpper et al. 2015), chart dark matter subhalos in the Milky Way halo (Nibauer et al. 2025), and investigate the potential property of dark matter itself (Mestre et al. 2024). In particular, the long, dynamically cold GD-1 stream has been used to constrain the Milky Way potential with various techniques,

including orbit fitting (Koposov et al. 2010; Malhan & Ibata 2019), backwards time integration (Price-Whelan et al. 2014; Palau et al. 2025), action-angle modeling (Bovy et al. 2016), action-angle clustering (Reino et al. 2022), and particle-spray with stream-track (Bowden et al. 2015) modeling. More recently, new machine learning tools have been implemented to face this challenging problem, as in Nibauer et al. (2022) and Nibauer & Bonaca (2025), where they estimated the acceleration felt by the stars in the stream as a probe of the Milky Way potential without relying on smooth analytic approximation. Notably, machine learning tools have opened a new avenue for more realistic representation of the potential.

Recent developments in generative models have boosted the adoption of the simulation-based inference (SBI) technique (Cranmer et al. 2020) as a valid alternative to the classical Bayesian approaches used in the study of gravitational waves (Dax et al. 2025), Galactic chemical enrichment (Buck et al. 2025; Gunes et al. 2025), Galactic archaeology (Viterbo & Buck 2024; Sante et al. 2025), the dark matter density profile in dwarf galaxies (Nguyen et al. 2023), and cosmology (Saoulis et al. 2025). In this work, we set out to jointly recover the gravitational

* Corresponding author: giuseppe.viterbo@iwr.uni-heidelberg.de

potential of the Milky Way together with the parameters of the progenitor of the stellar stream in order to compare and extend the work presented in Alvey et al. (2023). This paper is structured as follows. In Sect. 2, we present the simulation choices, the N -body simulator ODISSEO used to create the training set, an introduction to the flow matching technique, and the details of the model architecture. In Sect. 3, we present the results of the inference for our reference GD-1 simulation and the test set results to validate the posterior calibration and accuracy. In Sect. 5, we summarize our findings and discuss limitations and future prospects.

2. Method

Our goal is to recover, in a statistically sound way, the physical parameters, θ , that govern the tidal stripping of a globular cluster evolving within a Milky Way-like gravitational field. Achieving this requires a set of modeling decisions regarding how we represent the host-galaxy potential (Sect. 2.1) and how we describe the progenitor system that seeds the stellar stream (Sect. 2.2). To infer θ , we adopted a fully Bayesian framework. More specifically, we adopted a likelihood-free approach (SBI; Sect. 2.4). The SBI technique enables us to learn directly from simulations how the observed data, d , carry information about the underlying physical parameters. Specifically, we trained a generative model¹ to approximate the posterior distribution, $p(\theta | d)$, by automatically extracting informative summary statistics from the observations. The training dataset for this approach consists of pairs (d^i, θ^i) , where each synthetic observation, d^i , is produced through forward modeling. We drew parameters, θ^i , and passed them through a simulator, S (ODISSEO, Sect. 2.3, yielding $d^i = S(\theta^i)$). The forward model must also incorporate observational caveats (e.g., uncertainty, selection effect) so that the forward process is equivalent to sampling from the likelihood $p(d | \theta)$ ². This simulation-inference amortized pipeline allowed us to connect theoretical models of tidal stripping with the observable properties of stellar streams in a principled and scalable way.

2.1. The host: Model of the Milky Way

The gravitational potential of the Milky Way remains a subject of active debate. In the method outlined in the following section, we exploit the cold stellar stream GD-1 as a sensitive tracer of the Galaxy's underlying density field. As illustrated in Fig. 1, even when the properties of the progenitor are fixed, tidal stripping can give rise to strikingly different present-day stream morphologies, depending on the assumed Galactic potential. We adopted the widely used BovyMWPotential2014 model (Bovy 2014), which provides a smooth and flexible yet tractable three-component representation of the Milky Way.

This framework consists of the following:

1. A spherical dark-matter halo following a Navarro-Frenk-White (NFW) profile with density profile

$$\rho(r) = \frac{\rho_0}{\frac{r}{r_s} \left(1 + \frac{r}{r_s}\right)^2}, \quad (1)$$

where r is the spherical radial coordinate, r_s is the scale radius, and ρ_0 is the central density.

¹ A neural network designed to learn conditional probability distributions.

² For this reason, the SBI technique is also referred to as ‘‘implicit likelihood inference.’’

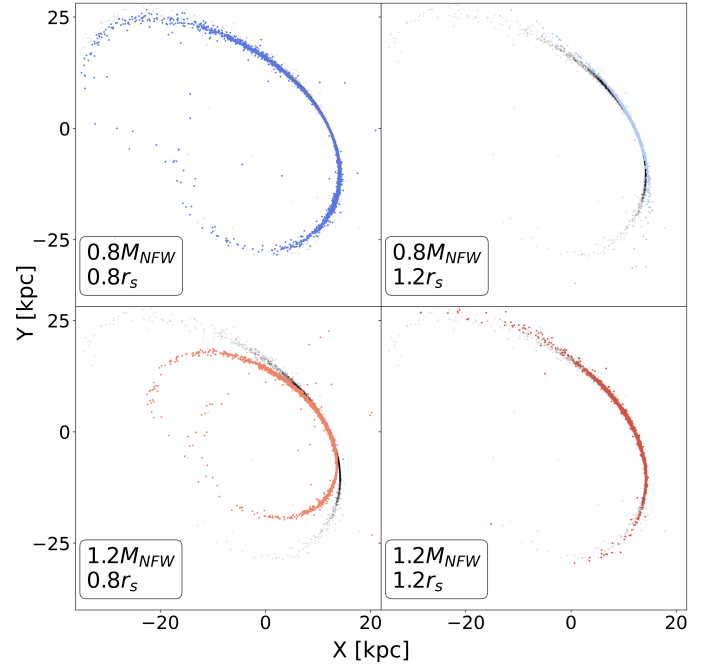


Fig. 1. Stream morphology in different Galactic potentials. The black scatter plot is the output of the simulation for the fiducial values of BovyMWPotential2014, and the same is shown in each subpanel. We show the results for various combinations of 20% offset of the mass of the NFW halo and its scale radius as indicated in the legend of each subpanel.

2. An axisymmetric disc described by a Miyamoto-Nagai (MN) potential

$$\Phi(R, z) = -\frac{GM_{MN}}{\sqrt{R^2 + (a + \sqrt{z^2 + b^2})^2}}, \quad (2)$$

with R, z being the radius and height in cylindrical coordinates, M_{MN} as the total mass of the disc, and a, b respectively as the scale length and height.

3. A spherical bulge with an exponential cutoff with a density profile of

$$\rho(r) = \rho_0 \left(\frac{r_1}{r_c}\right)^\alpha \exp\left[-(r/r_c)^2\right], \quad (3)$$

with r_c and α respectively as the cutoff radius and the power-law exponent.

2.2. The progenitor: Globular cluster

The progenitor of GD-1 is widely attributed to be a disrupted globular cluster. Such systems are well approximated as spherically symmetric self-gravitating stellar ensembles (Aarseth et al. 1974), for which a Plummer sphere provides an analytically convenient and physically motivated model. The Plummer potential captures both the central concentration and the finite spatial extent expected for low-mass, pressure-supported clusters, making it ideally suited for generating initial conditions in stream-formation simulations. A crucial aspect of modeling tidal disruption is that the internal phase-space distribution of stars within the progenitor directly shapes where and how stars are stripped. Stars located near the outer regions of the Plummer sphere – those with higher energies and larger orbital

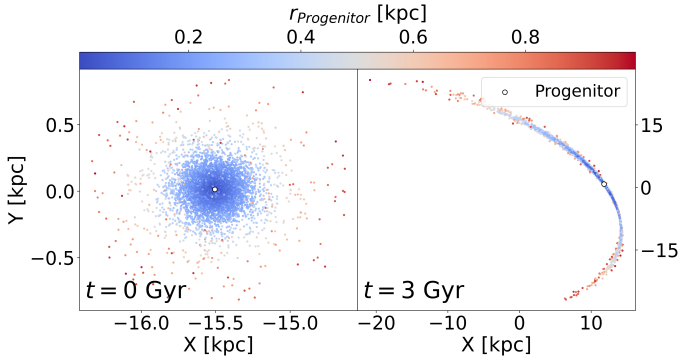


Fig. 2. Tidal stripping of $N=10^4$ stars from a Plummer sphere over a 3 Gyr evolution. The parameters to set this simulation were the fiducial BovyMWPotential2014 for the galaxy and $(M_{\text{Plummer}}, a_{\text{Plummer}}) = (10^{4.05} M_{\odot}, 100 \text{ pc})$ for the progenitor. The color bar indicates the initial radial distance from the progenitor.

radii – reach the tidal boundary earlier and are therefore removed first as the cluster orbits within the Galactic potential. Conversely, tightly bound inner stars escape later and typically with different initial velocities (see e.g., Skúladóttir et al. 2025; Buder et al. 2025a,b, for a recent exploitation of this effect for Galactic Archaeology). We show an example of this process by evolving a Plummer sphere in the fiducial BovyMWPotential2014 in Fig. 2.

These variations imprint distinct escape conditions along the orbit, influencing the width, density variations, and the energy–angle structure of the resulting stream. Consequently, even subtle differences in the progenitor’s density profile or velocity dispersion propagate into observable differences in the stream morphology, making an accurate generative model of the progenitor essential for robust inference. In the following section, we describe in detail how we sampled particles from a Plummer sphere to initialize the progenitor’s six-dimensional phase-space distribution. This sampling procedure forms one of the first steps of our forward model and ensures that the tidal stripping points – and thus the emergent stream structure – are physically consistent when generating the training data used for our SBI pipeline.

Differentiable initial condition. The initial positions and velocities of the progenitor depend inherently on the two parameters of the Plummer model: the total mass M_{Plummer} and its scale radius a_{Plummer} . To sample star particles from a Plummer sphere, we needed to sample (x, v) as follows:

1. Given the radial mass profile of the Plummer model,

$$M(r) = M_{\text{Plummer}} \left(\frac{r}{a_{\text{Plummer}}} \right)^3 \left(1 + \frac{r^2}{a_{\text{Plummer}}^2} \right)^{-3/2}, \quad (4)$$

we applied inverse sampling, using $F(r) = M(r)/M_{\text{Plummer}}$ as a cumulative distribution function to obtain r by simply evaluating

$$r = \sqrt{\frac{a_{\text{Plummer}}^2}{u^{-2/3} - 1}} \quad \text{where } u \sim \mathcal{U}(0, 1). \quad (5)$$

2. Given the radial distance r and assuming spherical symmetry, we generated (x, y, z) by sampling a random direction on the unit sphere

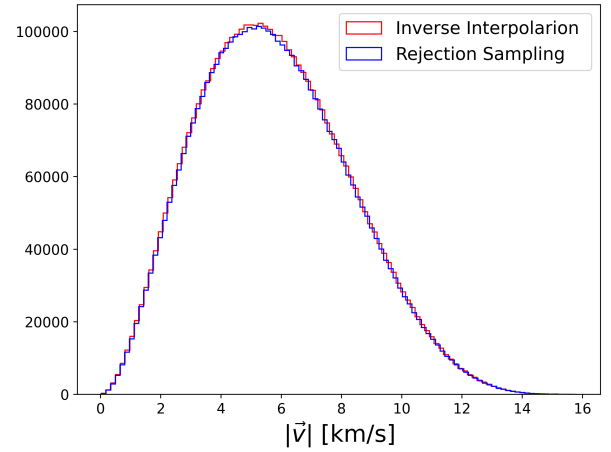


Fig. 3. Comparison between rejection sampling and inverse sampling of the interpolated inverse for the module of the velocity of particles sampled from a Plummer sphere.

3. Using the Plummer potential

$$\Phi_{\text{Plummer}}(r) = -\frac{GM_{\text{Plummer}}}{\sqrt{r^2 + a_{\text{Plummer}}^2}}, \quad (6)$$

we can find the associated escape velocity, $v_e(r)$, for each r by simply setting the total energy of the particle to be 0, obtaining $v_e(r) = \sqrt{-2\Phi(r)}$. This is the maximum velocity that a gravitationally bound particle can have. In order to obtain the velocity, we used the distribution function, which for the Plummer sphere has the closed form

$$g(v)dv \propto (-E)^{7/2} v^2 dv, \quad (7)$$

where $E = -v_e^2 + \frac{1}{2}v^2$. We then defined $q(v) = v/v_e$ so that the previous expression can be rewritten as the unnormalized probability density function:

$$g(q) = (1 - q)^{7/2} q^2 \quad \text{with } 0 \leq q \leq 1. \quad (8)$$

In this case, inverse sampling is not possible because the cumulative distribution function $G(q) = \int_0^q (1 - s)^{7/2} s^2 ds$ does not have an analytic inverse, F , such that $F(G(q)) = q$. However, since $G(q)$ can be evaluated numerically, we approximated its inverse, $\tilde{F}(u)$, with $u \sim \mathcal{U}(0, 1)$, by interpolating over a set of points $(G(q'), q')$ equally spaced values between zero and one. In practice, this amounts to swapping the input and output of G and constructing an interpolation of the inverse, which can then be sampled at the values of u to obtain our final samples for q . We report in Fig. 3 a comparison plot of sampling 10^5 particles, showing how close it matches the classical rejection sampling approach presented in Aarseth et al. (1974). An advantage over the classical rejection sampling approach is that the inverse interpolation sampling is differentiable, allowing for gradient-based approaches, which we could leverage easily with our differentiable simulator, as shown in Viterbo & Buck (2025). In this work however, we do not yet make use of the differentiability.

4. Once we had sampled q , we could rescale them using v_e to obtain the velocity module v . Finally, assuming spherical symmetry, we generated (v_x, v_y, v_z) by sampling a random direction on the unit sphere.

2.3. *N*-body simulation with `ODISSEO`

Although fast particle-spray algorithms exist (e.g., [Chen et al. 2025](#); [Fardal et al. 2015](#); [Nibauer et al. 2025](#)), we modeled tidal disruption directly using the direct `ODISSEO` *N*-body integrator. This self-consistent approach captures the ejection dynamics due to close encounters that particle-spray techniques approximate³, and it produces forward simulations well suited for a simulation-based inference pipeline. As presented in [Viterbo & Buck \(2025\)](#), `ODISSEO` is a *N*-body code developed to study particle systems in which external potentials play a crucial role. The code is implemented in a purely functional style in Jax ([Bradbury et al. 2018](#)), which ensures that the full simulation is trivially parallelizable on GPU, and the high-level Python interface enables easy setup, prototyping, community-driven development, and maintenance. In addition, `ODISSEO` natively supports just-in-time compilation and execution on CPUs, GPUs, and TPUs, ensuring both flexibility and computational efficiency. We used `ODISSEO` as our simulator to quickly generate the training set of mock GD-1 simulations that are needed to train the model used in our SBI approach. Additionally, as described in [Viterbo & Buck \(2025\)](#), the simulation can be differentiated by automatic differentiability (AD), allowing for gradient descent methods or variational inference for parameter estimation. In particular, the gradient can be pulled through the time integration and through the initial condition sampling, as described in Sect. 2.2. In this work, we do not use the differentiability.

Stellar stream simulations. In this section we describe the general steps that were used to generate simulations of a stellar stream. The steps are as follows:

1. *Cluster trajectory.* Given the present-day position and velocity ($\mathbf{x}^c, \mathbf{v}^c$) for the progenitor of the star cluster, we traced the trajectory back to t_{end} as a single particle with mass $M_{Plummer}$ in the chosen gravitational potential to the initial phase space position ($\mathbf{x}_0^c, \mathbf{v}_0^c$);
2. *Populate with star particles.* We drew $N=1000$ star particle positions and velocities $(\hat{x}_i, \hat{v}_i)_{i=0, \dots, N}$ from a Plummer potential centered on the origin, and then we shifted them in phase space by $(\mathbf{x}_0^c, \mathbf{v}_0^c)$;
3. *Stream evolution:* we evolved the star particles forward in time for a total integration time of t_{end} using the fifth order explicit Runge–Kutta method `Tsit5` ordinary differential equation (ODE) solver available in `difffrax` ([Kidger 2022](#)).

All the simulations were carried out using `ODISSEO`⁴. We decided to treat the particles as phase-space tracers, so we used a Plummer softening of 0.1 pc to avoid the formation of dynamical binaries.

2.4. Inference

Our aim was to jointly infer the parameters that describe the progenitor (θ_{prog}) of a fiducial GD-1 stream simulation and the parameters of the Milky Way potential (θ_{host}) in which the tidal stripping of its progenitor has happened. In practice, we aimed to both reproduce the results obtained by [Alvey et al. \(2023\)](#) and to extend the inference to also be able to use the stellar stream as a tracer for the gravitational potential of the host galaxy. We decided to face this challenging task by training

³ To keep the computational cost tractable, we nevertheless adopted a small Plummer softening length, as described in the following section.

⁴ The data are publicly available at <https://zenodo.org/records/17711491>.

Table 1. Prior ranges and true values for model parameters.

Parameter	Prior range	True value
Host parameters		
$M_{vir} (M_{\odot})$	$(0.5 M_{vir}^{true}, 2.0 M_{NFW}^{true})$	4.37×10^{11}
$r_{NFW} (\text{kpc})$	$(0.5 r_s^{true}, 2.0 r_s^{true})$	16.0
$M_{MN} (M_{\odot})$	$(0.5 M_{MN}^{true}, 2.0 M_{MN}^{true})$	6.82×10^{10}
$a_{MN} (\text{kpc})$	$(0.5 a_{MN}^{true}, 2.0 a_{MN}^{true})$	3.0
Progenitor parameters		
$t_{end} (\text{Gyr})$	(0.5, 5.0)	3.0
$M_{Plummer} (M_{\odot})$	$(10^{3.0}, 10^{4.5})$	1.12×10^4
$a_{Plummer} (\text{kpc})$	$(0.5 a_{Plummer}^{true}, 2.0 a_{Plummer}^{true})$	0.008
$x^c (\text{kpc})$	(10.0, 14.0)	11.8
$y^c (\text{kpc})$	(0.1, 2.5)	0.79
$z^c (\text{kpc})$	(6.0, 8.0)	6.4
$v_x^c (\text{km s}^{-1})$	(90.0, 115.0)	109.5
$v_y^c (\text{km s}^{-1})$	(−280.0, −230.0)	−254.5
$v_z^c (\text{km s}^{-1})$	(−120.0, −80.0]	−90.3

Notes. The prior ranges for t_{end} , $M_{Plummer}$, x_c^0 , v_c^0 are taken from [Alvey et al. \(2023\)](#).

a neural density estimator ([Cranmer et al. 2020](#)) of the posterior distribution $p(\theta | d_{obs})$, where $\theta = (\theta_{prog}, \theta_{host})$ and $d_{obs} = (\phi_1, \phi_2, r, v_{\phi_1} \cos \phi_2, v_{\phi_2}, v_r)$ is the phase space of all the stars in the GD-1 stellar stream projected on the plane of the stream, as in [Alvey et al. \(2023\)](#) and [Koposov et al. \(2010\)](#)⁵. Since we modeled the progenitor of the GD-1 with a Plummer sphere (Sect. 2.2), the parameters are the total time of integration, total mass, and the scale radius of the Plummer sphere, its present day position and velocity so that $\theta_{prog} = (t_{end}, M_{Plummer}, a_{Plummer}, \mathbf{x}^c, \mathbf{v}^c)$. Since we wanted to test the constraining power over the amplitude and shape of the host potential, we only varied the NFW halo mass and scale radius and the MN mass and scale length while fixing all the other parameters to the default fiducial parameters in `Galax` ([Starkman et al. 2024](#)) for `MWPotential2014` in order to contain the computational cost of sampling on a small but still informative parameter space, leaving for future work extension to models with higher degrees of freedom, such as triaxial NFW profiles or time-evolving potentials. The final host potential parameters were then $\theta_{host} = (M_{NFW}, r_{NFW}, M_{MN}, a_{MN})$. To train a neural density estimator we needed to generate pairs of parameters and an observation $(\theta, d) \sim p(\theta, d)$ by letting $\theta \sim p(\theta)$, with $p(\theta)$ being the prior over the parameters, and $d = S(\theta) \sim p(d | \theta)$, with S being the `ODISSEO` simulator that implicitly defines the likelihood $p(d | \theta)$. Moreover, we applied to the observation the same observational window and noise level reported in Table 2 in [\(Alvey et al. 2023\)](#), leaving the background contamination and selection functions effect for future work. The prior choice is reported in Table 1.

Flow matching posterior estimation. In recent years, many SBI applications, performed using normalizing flow architecture as a neural posterior estimator, have been proposed in the field

⁵ These are the standard set of co-ordinates used in the literature: (ϕ_1, ϕ_2) are two angles coordinate, the corresponding proper motion (v_{ϕ_1}, v_{ϕ_2}) , and radial distance and velocities (r, v_r) .

(Nguyen et al. 2023; Sante et al. 2025; Viterbo & Buck 2024; Ho et al. 2024). The core idea is to leverage the flexibility of a neural network to learn a series of invertible transformations, conditioned on the observations, that project the parameters in a latent space where it is easy to sample from. In this way, instead of relying on the expensive “gold standard” Mark chain Monte Carlo technique, the transformations, via the change of variable formula, take care of tracking how the parameter space has been changed through the flow to the sample and evaluate the posterior distribution. The expressivity of this technique is limited by the necessity of using an invertible transformation (usually spline functions as described in Durkan et al. 2019). For this reason, neural posterior score estimation (Geffner et al. 2022) and flow matching posterior neural estimation (FMPE; Wildberger et al. 2023) tackle the problem of approximating a conditional distribution with a different approach. As described in Wildberger et al. (2023), inspired by the promising results in generative tasks for which they were developed, these models transform noise into samples via trajectories parametrized by a continuous time variable t . In particular, for FMPE, the goal is to regress the vector field v_ϕ , parametrized by a neural network with weights ϕ , that describes these trajectories by solving an ODE. The key advantage over normalizing flows is that by regressing on the vector field v_t , we are free in the choice of the network architecture, at the cost of multiple network passages for sampling, since we need to solve the ODE for $t \in [0, 1]$. Moreover, contrary to neural posterior score estimation, which needs to solve a Stochastic differential equation, we are also able to track the posterior density directly. As described in Sect. 2.4, we respectively trained observations and parameters on tuples (d, θ) . Our objective was to learn a continuous transformation for θ (or equivalently a probability path q_t) with constraints $\theta \sim q_0$ and $\theta \sim q_1 = p(\theta | d)$, where q_0 is a sampling distribution, in our case a normal distribution with zero mean and identity covariance matrix. The ODE that controls this continuous process can be expressed as

$$d\theta_t = v_\phi(t, d, \theta_t)dt. \quad (9)$$

Flow matching actually falls in the paradigm of continuous normalizing flow, but with an alternative objective function. In fact, since continuous normalizing flows are trained using a negative log likelihood as objective, one must track at training time the computationally expensive $p(\theta | x)$ by

$$p(\theta | x) = q_1 = q_0 \exp\left(\int_0^1 \nabla \cdot v_\phi(t, d, \theta)dt\right), \quad (10)$$

making the training of these models infeasible. As described in (Wildberger et al. 2023), the key of flow matching is to directly regress the vector field v_ϕ on a vector field u_t that generates the desired probability path p_t while avoiding the integration of the ODE at training time⁶. The nontrivial solution on how to perform this task, presented in Lipman et al. (2022), is to choose this path depending on the sample θ_1 ⁷ for which we want to model the probability path. In practice, we modeled the conditioned probability path $q_t(\theta | \theta_1)$ and the corresponding vector field $u_t(\theta | \theta_1)$ with the sample conditional flow matching (CFM) loss

$$\mathcal{L}_{CFM} = \mathbb{E}_{t \sim U[0,1], d \sim p(d|\theta), \theta_t \sim q_t(\theta_t|\theta_1)} \|v_\phi(t, d, \theta_t) - u_t(\theta_t | \theta_1)\|^2. \quad (11)$$

⁶ The integration is still needed at inference time.

⁷ We indicate with θ_1 the samples θ to be consistent with the literature on flow matching.

One simple choice for the conditioned probability path is the Gaussian path family

$$q_t(\theta | \theta_1) = \mathcal{N}(\theta | t\theta_1, (1 - (1 - \sigma_{min})t)\mathbb{I}), \quad (12)$$

which generates the velocity field

$$u_t(\theta | \theta_1) = \frac{\theta_1 - (1 - \sigma_{min})\theta}{1 - (1 - \sigma_{min})t}, \quad (13)$$

with $\sigma_{min} > 0$. These choices lead to this problem coinciding with the optimal transport (Lipman et al. 2022) between two Gaussian distributions: the sampling distribution and linear trajectory $t\theta_1$, ending in θ_1 and with a smoothing constant given by σ_1 . The steps to be followed to train boil down to sampling $\theta_1 \sim p(\theta)$ and transporting a point $\theta_0 \sim \mathcal{N}(0, \mathbb{I})$ from the sampling distribution to the posterior distribution on the linear trajectory $t\theta_1$ and ending in θ_1 .

In the upper part of Fig. 4, we summarize the SBI task. We first sampled from the prior parameters θ , then forward modeled them using `ODISSEO` to get the observation d , and used the couples (d, θ) as our training set. In the lower part of Fig. 4, we show a representation of what the inference task looks like. We modeled the vector field v_ϕ with a neural network that is called for each integration step ($t \in [0, 1]$) of the ODE that transport the sampling distribution $\mathcal{N}(0, \mathbb{I})$ into the posterior distribution $p(\theta | d)$.

2.5. Architecture: SetTransformer

Given the permutation-invariant nature of our data – the set of N -body particles – we adopted the SetTransformer architecture introduced by (Lee et al. 2018). Our model extends this framework to infer the vector field v_ϕ , conditioned on the flow matching integration time t , the model parameters θ_t , and the particle observations d .

Each input was first embedded into a latent representation via multilayer perceptrons (MLPs). The integration time t was further encoded through a sinusoidal time embedding following (Ho et al. 2020). The embedded particle features are processed by a stack of self-attention blocks (SABs) with residual connections, which capture interactions among particles. To incorporate temporal conditioning, the output of each SAB was modulated via feature-wise linear modulation (FiLM; Perez et al. 2017) using the embedded time representation. Before producing the vector field, a cross-attention block (CAB) was applied, where the query Q corresponds to the embedded parameters θ_t , and the keys K and values V are given by the SAB plus FiLM representation of the particle set. In this configuration, the model parameters act as queries that guide the attention toward relevant features within the particle embeddings. The cross-attention output was then modulated again through a FiLM layer conditioned on the embedded time. Finally, the predicted vector field v_ϕ was obtained via a last MLP layer. The architecture is described in Fig. 5.

2.6. Training details

We generated 2×10^5 stellar streams, each containing a fixed number of stars, $N=1000$ ⁸. Of these, 5×10^4 samples were used for validation and 10^3 for testing. In Appendix A we report an

⁸ We chose this number to have the same order of magnitude of stars as in GD-1 membership studies, such as the one presented in Ibata et al. (2024).

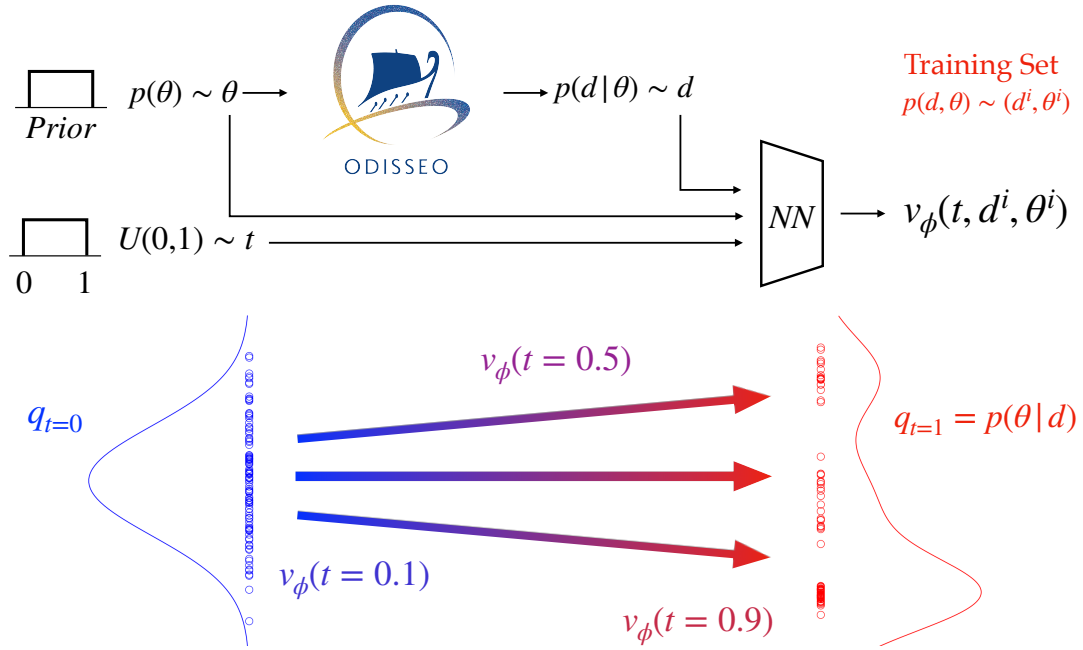


Fig. 4. Schematic of flow matching for posterior estimation with `ODISSEO`. The training set was generated by sampling $i = 0, \dots, N$ parameters θ^i from the prior $p(\theta)$ and then forward modeled using `ODISSEO` to obtain the observation $d^i \sim p(d|\theta)$. Following the flow described in Sect. 2.4, we sampled $t \sim U(0, 1)$ to train a neural network to approximate the vector field $v_\phi(t, d^i, \theta^i)$. In the lower section, we report a simplified flow matching objective for a 1D case. The neural network is called for different t to regress the vector field that governs the ODE in order to transform the sampling distribution $q_{t=0} = \mathcal{N}(0, I)$ into the posterior distribution $q_{t=1} = p(\theta|d)$. Note that in this schematic, we refer to θ_1 described in Sect. 2.4 as θ .

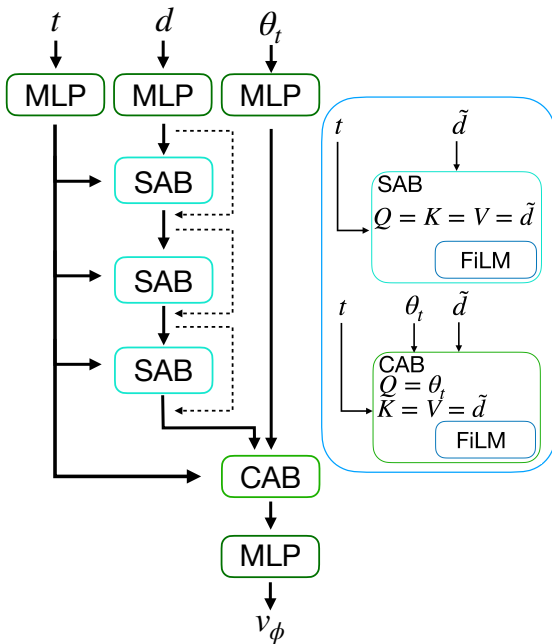


Fig. 5. Flow matching SetTransformer. We indicate with \tilde{d} both the observation (d) and the intermediate output of the attention blocks. To embed each of the stars in the observation (d) the parameter state θ_t at the ODE time t , we used a simple MLP with 128 neurons with a SiLU activation function. We then passed \tilde{d} through three stacked SABs with a skip connection (dashed line) to encode the correlations between the particles while modulating the output of each block on t using FiLM. Then we used a CAB with a FiLM modulation to focus the attention mechanism on finding the relevant feature in \tilde{d} to regress the parameters θ . The vector field v_ϕ was obtained by compressing the output of the CAB through an MLP with output dimensions equal to the dimensionality of θ , in our case 13 dimensions.

ablation study used to verify that the number of simulations is enough to reach convergence of the training. The training employed early stopping with a patience of 30 epochs, resulting in a total of 123 training epochs. We used a batch size of 500 and optimized the model with AdamW, starting from a learning rate of 10^{-3} and applying a reduce-on-plateau schedule with a patience of five epochs. The final architecture, illustrated in Fig. 5, was selected after extensive experimentation with different design choices and hyperparameters, including the number of neurons, the number of SABs and CAB blocks, activation functions, modulation schemes, and related configurations. The training set generation took ~ 7.5 GPU hours on an NVIDIA (Hopper) H200 (140 GB). The training and testing of the FMPE took ~ 8 hours on an NVIDIA (Ampere) A100 (40 GB). The entire inference pipeline was carried out using the `sbi-sim`⁹ package presented in Holzschuh & Thuerey (2024).

3. Results

We demonstrate the validity of the results of our inference by reporting a few metrics obtained on the test set. This set of (d, θ) was not used during training. For each of the test set couples, we sampled 10^3 samples from the posterior distribution. In order to evaluate the accuracy and calibration over the whole test set, as we report in Sects. 3.1.1 and 3.1.2, respectively, we predicted-true plots and percentile-percentile plots. In Sect. 3.2 we report the posterior's samples obtained from inferences based on our mock observation of the GD-1 stream. For this test, we decided to sample 10^4 times the posterior. Moreover, we performed posterior predictive checks to show that by forward passing these posterior samples, we obtain an observation that resembles the mock observation of GD-1.

⁹ <https://github.com/tum-pbs/sbi-sim/tree/dev>

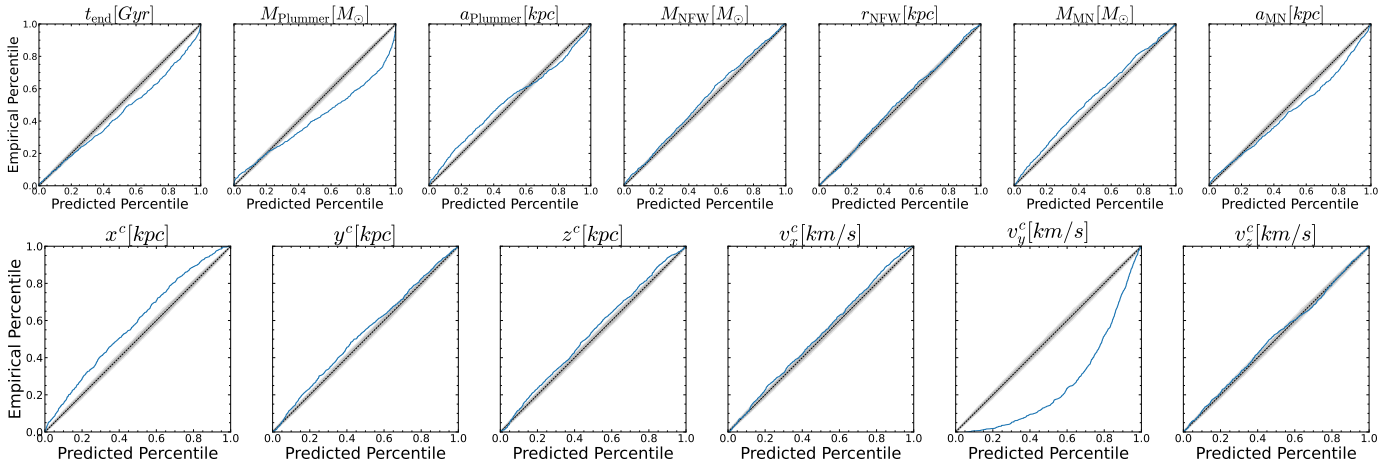


Fig. 6. Percentile-percentile plot for the marginal posterior distributions over the test set.

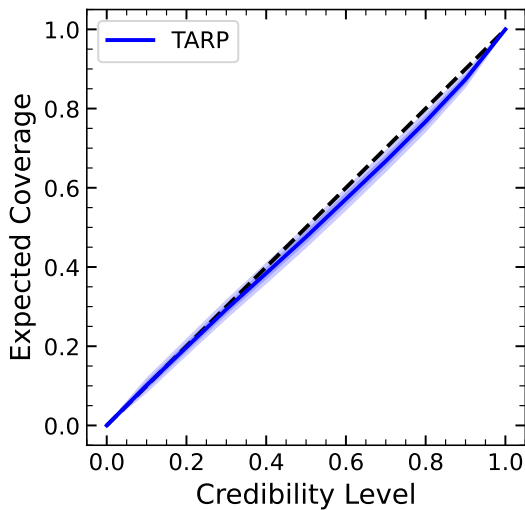


Fig. 7. Tarp plot for the joint posterior distribution over the test set.

3.1. Model evaluation

3.1.1. Calibration

Percentile–percentile (P–P) plots are a diagnostic tool used to evaluate the calibration of posterior distributions in a probabilistic inference. The key concept is that a well-calibrated model should produce posteriors whose credible intervals contain the true parameter values with the correct empirical frequency. For the marginal distributions of the predicted posterior, the P–P plot visualizes this property by comparing the predicted percentiles under the inferred posterior against the empirical percentiles (the fraction of true parameters that fall below the corresponding posterior quantile). Ideally, if the posterior is perfectly calibrated, these two quantities coincide, and the curve follows the diagonal line. By inspecting these plots and their deviations from the diagonal, we can spot deviations from the true posterior distributions, such as over- or underconfidence and positive and negative biases. We report the marginal cover plots for individual parameters in Fig. 6 and the joint coverage plot in Fig. 7. One can appreciate that for the θ_{host} , the P–P plots of marginal cover suggest a good calibration, while the two S-shape behavior in the Plummer parameters suggests a possible bias,

which for the case of $M_{Plummer}$ is a positive bias as reflected by the over-estimation shown in Fig. 8. Moreover, a significant underconfidence (overconfidence) is shown for x^c (v_y^c).

However, marginal posterior coverages do not tell the whole story. We adopted the “tests of accuracy with random point” (TARP) to study the behavior of the joint posterior distribution. As described in Lemos et al. (2023), TARP provides a necessary and sufficient condition for the joint posterior coverage. The method constructs spherical credible regions around randomly chosen points in parameter space and measures the fraction of posterior samples contained within each region. By repeating this process over many random points, TARP estimates the empirical coverage as a function of the nominal credibility level. A perfectly calibrated joint posterior yields a one-to-one correspondence between nominal and empirical coverage, represented by a diagonal trend in the resulting TARP curve. We report our Tarp in Fig. 7 to show we have good coverage on the joint distribution evaluated over the test set. TARP is sensitive to the geometry of the joint parameter space. Marginal P–P plots, on the other hand, assess one-dimensional projections obtained by integrating over all other parameters. These marginal checks can reveal parameter-specific biases or calibration issues that may not be immediately apparent in the joint coverage statistics, even when the overall joint posterior is well calibrated. Therefore, both diagnostics are complementary and indispensable for comprehensively assessing posterior coverage.

3.1.2. Accuracy

Having assessed that our posterior is well calibrated on almost all the marginals and the joint, we report the accuracy that we can expect by using our model in the form of true-predicted plots. In Fig. 8, the circles represent the median value, and the error bars represent the 16th to 84th percentile intervals. In the lower plots of the figure, we report the distribution of residuals to capture any possible residual trends. Our model seems to be extremely capable of capturing the posterior distribution for most of the parameters θ , except for the $M_{Plummer}$ and $a_{Plummer}$, for which a mostly uniform trend seems to emerge when we inspect the distribution of the residuals. We interpret the mass bias as being due to the fact that under $\sim 10^3 M_\odot$, the feature that the progenitor can imprint of the stars is limited, and the stars are mostly dominated by the host potential. The Plummer scale $a_{Plummer}$ seems to play a less important role compared to other quantities, and the results

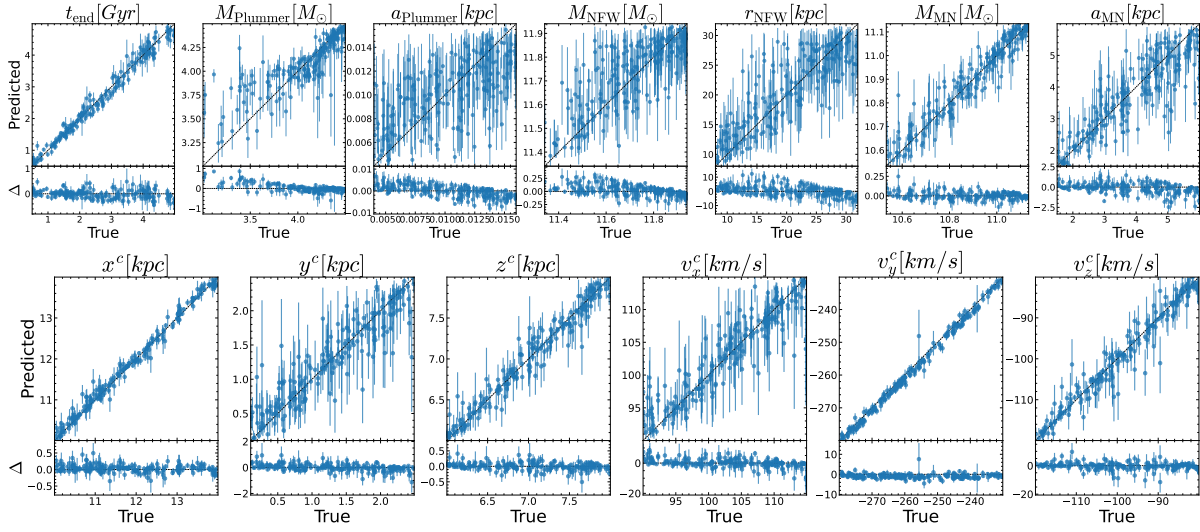


Fig. 8. True-predicted plots for the test set. We subsampled the test set to 500 for visual clarity. The circles represent the median, while the error bars report the 16th to 84th percentiles. We also report the residuals Δ , with error bars at the 16th to 84th percentiles.

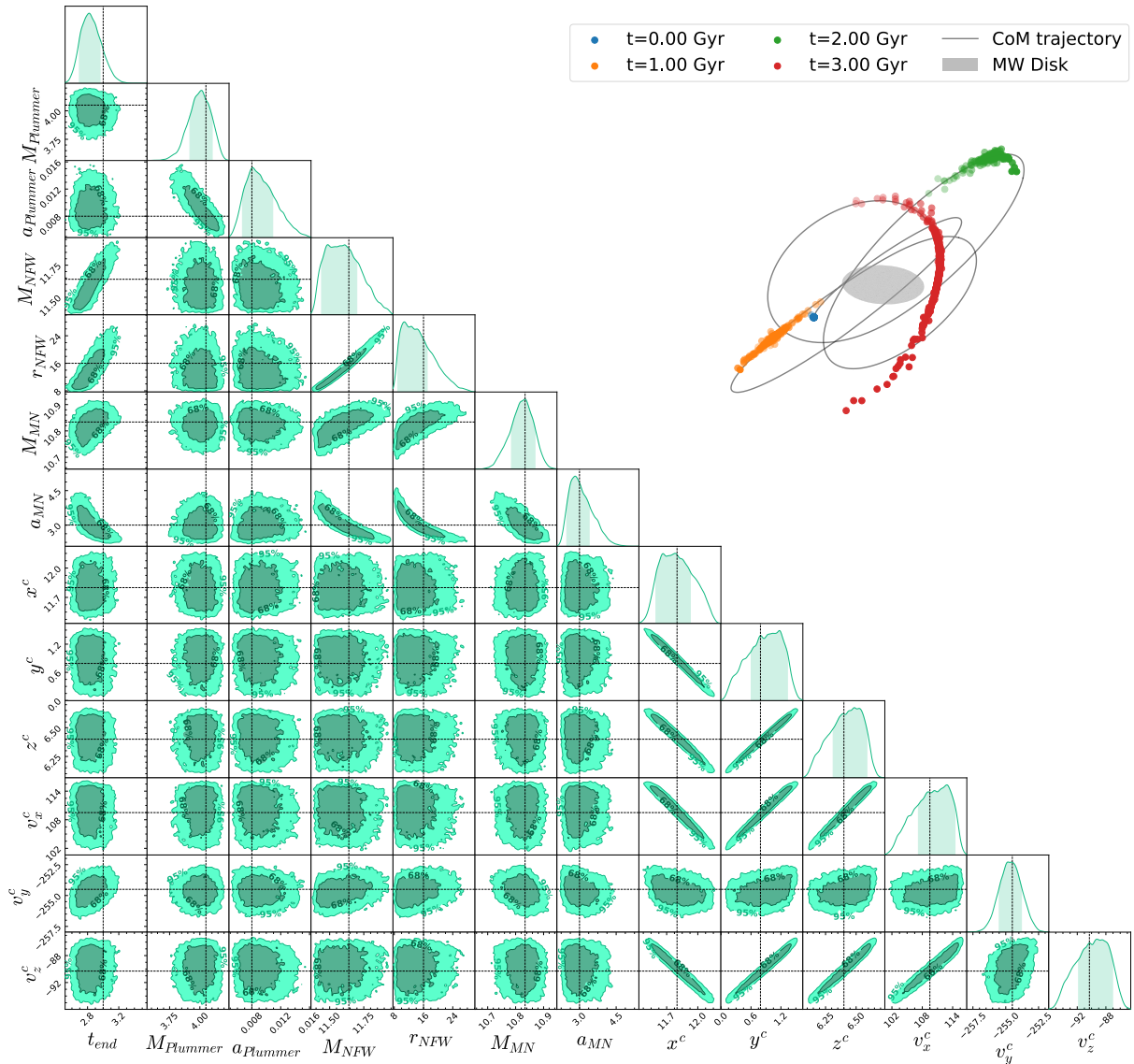


Fig. 9. Lower left: posterior samples from the fiducial GD-1 observation. The dashed line corresponds to the true value column in Table 1. Upper right: snapshots of our fiducial GD-1 simulation. We reported an 8 kpc Milky Way disc for reference.

of our analysis could be limited by the choice of a insufficiently broad prior.

3.2. GD-1

In the following section we show the posterior distribution for our mock true observation of the GD-1 stream. This simulation is meant to allow for a direct comparison with the results of Alvey et al. (2023), even though the set of parameters common between the two approaches is limited to $(t_{end}, M_{Plummer}, \mathbf{x}^c, \mathbf{v}^c)$. In Fig. 9, we report the corner plot for the posterior. The results suggest that we can strongly constrain these parameters since the true value always lies inside the 16th to 84th percentile and that we can reproduce the results on Alvey et al. (2023). We also captured the correlation between the host potential parameters, for example, (M_{NFW}, r_{NFW}) and (M_{MN}, a_{MN}) . These correlations are expected since different combinations lead to the same enclosed mass at a given radius. In Fig. 9, one can also appreciate the correlations in the phase space $(\mathbf{x}^c, \mathbf{v}^c)$, a reflection of the spherical symmetry of the problem. In agreement with the literature (Koposov et al. 2010; Nibauer et al. 2022; Palau et al. 2025), these results show that stellar streams can probe the shape and amplitude of the gravitational potential.

Posterior predictive check. As a complementary check, we implemented a forward pass on the posterior samples from Fig. 9 to obtain $d \sim P(d | \theta_{\text{Posterior}})$. Our aim was to capture biases introduced by the inference, which would be reflected as a deviation of $d \sim P(d | \theta_{\text{Posterior}})$ from d_{True} . For each of the samples in Fig. 9, we generated a stellar stream with 1000 stars and simulated the tidal disruption as described in Sect. 2.3. We then combined all the streams and checked if the distribution in observable space d obtained by this procedure deviates from the stars in our mock observation. We report the results in Fig. 10. In this plot, the ‘‘ground truth’’ is not a single point but a distribution of particles, so the black contours are given by displaying 1000 test set simulations, each with 1000 star particles. We note that all major stream features are captured nicely by our model.

Moreover, in Fig. 11 we report the observation obtained by forward modeling our estimate of the parameters and the posterior mean, which was obtained by the $N = 1000$ samples in our posterior as $\langle \theta_{\text{Posterior}} \rangle = 1/N \sum_{i=0}^N \theta_i$. Also in this test, we observed strong agreement with the mock observation.

4. Discussion

4.1. Comparison to classical GD-1 analyses

The GD-1 stellar stream has been widely used to constrain the Milky Way gravitational potential while using a variety of dynamical techniques (see Sect. 1), each making different approximations about stream formation and evolution. Orbit-fitting and backwards integration approaches have demonstrated that GD-1 provides strong constraints on large-scale properties of the Galactic potential, such as the circular velocity and overall flattening, but it typically approximates the stream as tracing a single orbit and therefore neglects its internal phase-space structure (e.g., Koposov et al. 2010; Malhan & Ibata 2019; Price-Whelan et al. 2014). Action-angle based methods partially relax this assumption by modeling the stripping process in angle space, enabling improved recovery of stream morphology and yielding precise constraints on the shape of the gravitational force field at the stream location, although they often rely on analytic potential models and simplified prescriptions for mass loss (e.g., Bovy 2014). Particle-spray and stream-track techniques

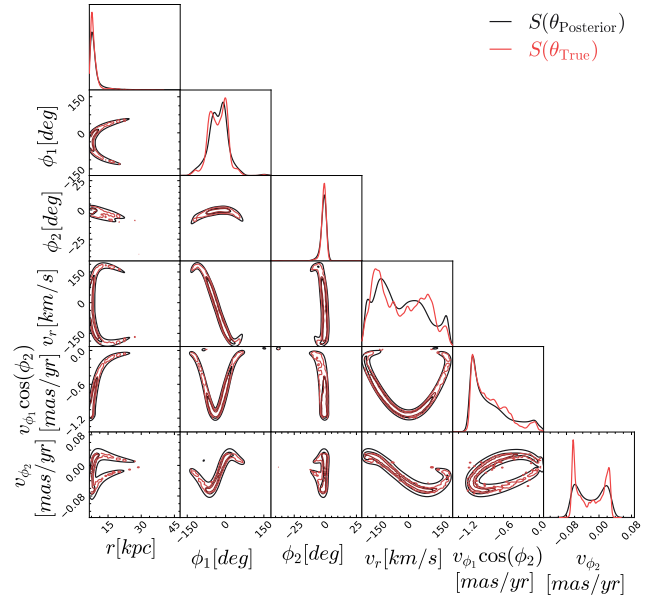


Fig. 10. Posterior predictive check. We report in red our mock observation of the GD-1 stream obtained using the true values in Table 1. In black we report the joint results of forward modeling the 1000 samples in the posterior distribution.

provide a compromise between physical fidelity and computational cost by approximating the escape of stars from the progenitor, but they do not capture the fully self-consistent dynamical evolution of the progenitor–stream system (e.g., Bowden et al. 2015).

In contrast, the framework presented here performs fully joint Bayesian inference on both progenitor properties and host-galaxy potential parameters using self-consistent N -body forward modeling combined with simulation-based inference. Rather than fitting a simplified representation of the stream, the method learns the mapping between physical parameters and the full phase-space distribution of stream stars. In controlled experiments, we have demonstrated accurate recovery of the full 13-dimensional parameter vector describing the Galactic potential and progenitor phase-space properties (see Fig. 9). We report a calibrated joint posterior (Fig. 7) and calibrated marginal posteriors for the host and for most progenitor parameters (see Fig. 6 and Sect. 3.1.1). This joint inference naturally captures degeneracies between host and progenitor parameters that are typically fixed or marginalized over in classical analyses. The principal trade-off is the upfront computational cost associated with generating training simulations and training the neural posterior estimator. However, once trained, the amortized nature of the method enables rapid posterior inference without additional simulations or likelihood evaluations. In addition, our GPU accelerated N -body code ODYSSEO (Viterbo & Buck 2025) presents a very efficient way of performing stream simulations. In this sense, our approach complements classical techniques by prioritizing modeling fidelity, joint inference, and posterior calibration over analytic tractability.

4.2. Model misspecification and robustness

A fundamental assumption underlying SBI is that the observed data are generated by a process that lies within, or sufficiently close to, the family of simulators used for training. In the present work, the neural posterior estimator is trained exclusively on

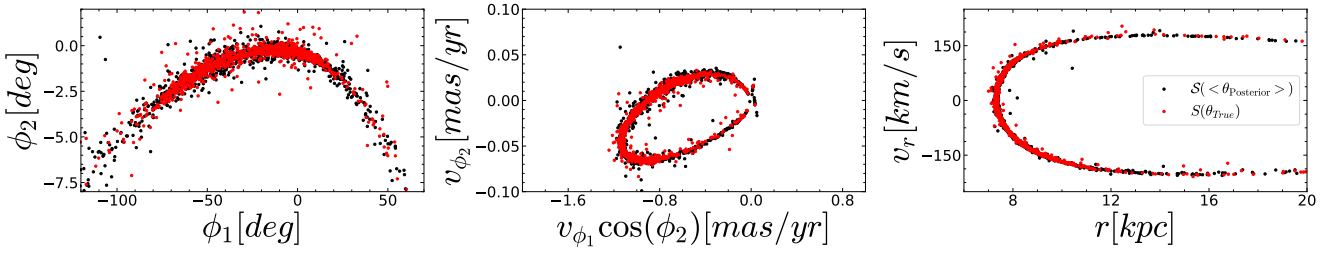


Fig. 11. Posterior predictive check. We report in red our mock observation of the GD-1 stream just as in Fig. 10, and in black we show the forward model of the mean of the posterior samples, which is our estimate of the parameters θ .

simulations generated using the `BovyMWPotential2014` functional form (Bovy et al. 2016). If the true Milky Way potential deviates from this family, for example through halo triaxiality, time-dependent mass growth, or perturbations induced by massive satellites such as the Large Magellanic Cloud, the inference problem becomes formally misspecified or out-of-distribution. In such cases, the learned posterior is no longer guaranteed to be calibrated and may exhibit biased parameter estimates, underestimated uncertainties, or failures of posterior coverage, even if it performs well on in-distribution test simulations.

Recent work in the SBI literature has shown that model misspecification leads to characteristic statistical signatures that can be diagnosed using principled validation tools (e.g., Schmitt et al. 2021; Zhou et al. 2025). Simulation-based calibration, posterior predictive checks, and classifier-based two-sample tests can be used to assess whether the observed data are statistically compatible with the assumed simulator family. In our framework, posterior predictive simulations drawn from the inferred parameter distribution (e.g., Figs. 10 and 11) can be directly compared to the observed stream morphology and phase-space structure, allowing systematic residuals or mismatches to be identified as indicators of forward-model incompleteness.

From an astrophysical perspective, unmodeled perturbations to the Galactic potential are expected to primarily affect localized or higher-order features of the stream, such as density variations, track offsets, or kinematic asymmetries, rather than uniformly shifting all global potential parameters. Consequently, model misspecification may manifest as broadened posteriors or altered parameter degeneracies rather than simple point-estimate biases. Importantly, the simulation-based nature of the present framework makes it straightforward to extend the training distribution to include more flexible or perturbed potential models (e.g., time-dependent halos, non-axisymmetric components, or explicit satellite perturbations) without requiring analytic likelihoods or changes to the inference machinery. Similarly, our N -body simulation based approach will be able to make use of prior constraints in the potential shape as informed by cosmological simulations (e.g., Obreja et al. 2022) and will not be limited to stellar streams alone. It can similarly make use of any halo substructure as found by state-of-the-art structure finders (Oliver et al. 2024; Oliver & Buck 2024). Furthermore, our approach will not be limited to the Milky Way alone but might as well be applied to the large sample of external galaxies hosting stellar streams (e.g., Miró-Carriero et al. 2025). The present work therefore provides a controlled baseline within which robustness to model misspecification can be systematically explored in future studies.

4.3. Pathway toward application to real GD-1 data

While our current work focuses on an idealized simulation setting, the framework is designed with application to real GD-1

observations in mind. Extending the analysis to realistic survey data primarily requires augmenting the forward model rather than modifying the inference procedure. In practice, this involves sampling stellar magnitudes from the observed luminosity function of GD-1, applying survey-specific selection functions, and resampling phase-space coordinates with magnitude-dependent uncertainties appropriate for Gaia and complementary spectroscopic surveys. Background contamination can be incorporated by forward modeling a mixture of stream and field stars, allowing the neural posterior estimator to marginalize over membership uncertainty. Conceptually, these extensions correspond to replacing the idealized observation block in the SBI pipeline shown in Fig. 4 with a survey-aware observational model while leaving the remaining components unchanged, as also demonstrated in recent SBI-based dynamical analyses (e.g., Nibauer et al. 2022; Alvey et al. 2023).

Specifically, the choice to fix the number of stars per simulation to $N = 1000$ was motivated by computational considerations and by the desire to match the order of magnitude of the current GD-1 membership samples. This choice does not represent a fundamental limitation of the method. The `SetTransformer` architecture employed here is permutation invariant and can, in principle, handle variable-size inputs. In future applications, robustness to varying sample sizes can be achieved by drawing the number of stars per simulation from a distribution informed by the survey selection function or by randomly subsampling from larger simulated streams during training. These strategies enable the model to learn invariance to completeness and membership size, which will be essential when analyzing real observational data.

5. Conclusion

With this work, we have presented the first application of SBI to the GD-1 stream to jointly constrain parameters of the progenitor and the host potential. With the `ODISSEO` simulator, we ran a large set of mock GD-1 stream analogs to reproduce the tidal stripping of a globular cluster, which we used to train our neural density estimator. We adopted a `SetTransformer` architecture to automatically extract summary statistics from the observations x , enabling a robust inference implemented via a flow matching strategy. Our results are summarized below:

- We have produced a large set of publicly available mock GD-1 streams using the `ODISSEO` simulator;
- We have extended the work of (Alvey et al. 2023) to also study the gravitational potential of the host galaxy, achieving an amortized inference that does not require sequential training;
- We have shown the validity of our model with extensive testing in the form of P-P plots for coverage, true-predicted plots for accuracy, and posterior predictive checks for self-consistency;

- We recovered the fiducial parameters over our fiducial GD-1 observation in Fig. 9 for this controlled experiment. However, performance under more realistic survey conditions remains to be tested. We captured strong correlations between mass and scale length parameters ($M_{NFW} - r_{NFW}$; $M_{MN} - a_{MN}$; $r_{NFW} - a_{MN}$) and, surprisingly, a weak positive correlation between the masses ($M_{NFW} - M_{MN}$). We also captured a strong correlation between the present-day phase space of the progenitor of GD-1 (x^c, v^c), a result of the axisymmetry of the problem. A notable exception is v_y^c , for which our model is also not well calibrated, as can be observed in Fig. 6;
- We leveraged the freedom given by the flow matching technique to adopt a robust transformer architecture capable of handling this high dimensional inference task.

Our aim was to quantify, in a controlled simulation setting, the information that a single cold stellar stream (GD-1 analogs) carries about the progenitor properties and the host galaxy potential. We did so by combining self-consistent N -body forward modeling (ODISSEO) with amortized likelihood-free inference (flow matching) and by validating the calibration and accuracy on held-out synthetic data.

In future work, we will extend this method with a more robust, realistic, and survey-dependent handling of the observational errors, which we have simplified in this work; magnitude-dependent selection functions; and background contamination, which were both ignored. We also intend to extend the pipeline to leverage multistream observation, inspired by the results obtained in Bovy et al. (2016). Lastly, since ODISSEO is capable of calculating the gradient of the simulation with respect to the input parameters, we aim to incorporate this as additional information to guide the inference pipeline, as shown in Holzschuh & Thuerey (2024).

Data availability

We publicly release our code to reproduce all the figures via GitHub: https://github.com/vepe99/sbi-sim/tree/odisseo_branch.

Acknowledgements. This project was made possible by funding from the Carl-Zeiss-Stiftung. This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC 2181/1 – 390900948 (the Heidelberg STRUCTURES Excellence Cluster). We acknowledge the usage of the AI-clusters Tom and Jerry funded by the Field of Focus 2 of Heidelberg University. We thank William H. Oliver, Lorenzo Branca, Leonard Storcks and Anna Lena Schaible for insightful discussions and stimulating conversations that helped shape the ideas presented in this work. We are grateful to Paola Ziero for the design of the project logo and her creative support.

References

Aarseth, S. J., Hénon, M., & Wielen, R. 1974, *A&A*, 37, 183
 Alvey, J., Gerdes, M., & Weniger, C. 2023, *MNRAS*, 525, 3662
 Binney, J., & Tremaine, S. 2008, *Galactic Dynamics*, 2nd edn. (Princeton University Press)
 Bovy, J. 2014, *ApJ*, 795, 95

Bovy, J., Bahmanyar, A., Fritz, T. K., & Kallivayalil, N. 2016, *ApJ*, 833, 31
 Bowden, A., Belokurov, V., & Evans, N. W. 2015, *MNRAS*, 449, 1391
 Bradbury, J., Frostig, R., Hawkins, P., et al. 2018, *JAX: composable transformations of Python+NumPy programs*
 Buck, T., Macciò, A. V., Dutton, A. A., Obreja, A., & Frings, J. 2019, *MNRAS*, 483, 1314
 Buck, T., Obreja, A., Macciò, A. V., et al. 2020, *MNRAS*, 491, 3461
 Buck, T., Günes, B., Viterbo, G., Oliver, W. H., & Buder, S. 2025, *A&A*, 702, A184
 Buder, S., Buck, T., Skúladóttir, Á., et al. 2025a, arXiv e-prints [arXiv:2510.11284]
 Buder, S., Buck, T., Skúladóttir, Á., et al. 2025b, arXiv e-prints [arXiv:2510.20233]
 Chen, Y., Valluri, M., Gnedin, O. Y., & Ash, N. 2025, *ApJS*, 276, 32
 Cranmer, K., Brehmer, J., & Louppe, G. 2020, *PNAS*, 117, 30055
 Dax, M., Green, S. R., Gair, J., et al. 2025, *Nature*, 639, 49
 Durkan, C., Bekasov, A., Murray, I., & Papamakarios, G. 2019, arXiv e-prints [arXiv:1906.04032]
 Fardal, M. A., Huang, S., & Weinberg, M. D. 2015, *MNRAS*, 452, 301
 Geffner, T., Papamakarios, G., & Mnih, A. 2022, arXiv e-prints [arXiv:2209.14249]
 Günes, B., Buder, S., & Buck, T. 2025, arXiv e-prints [arXiv:2507.05060]
 Ho, J., Jain, A., & Abbeel, P. 2020, arXiv e-prints [arXiv:2006.11239]
 Ho, M., Bartlett, D. J., Chartier, N., et al. 2024, *Open J. Astrophys.*, 7, 54
 Holzschuh, B., & Thuerey, N. 2024, arXiv e-prints [arXiv:2410.22573]
 Ibata, R., Malhan, K., Tenachi, W., et al. 2024, *ApJ*, 967, 89
 Kidger, P. 2022, arXiv e-prints [arXiv:2202.02435]
 Koposov, S. E., Rix, H.-W., & Hogg, D. W. 2010, *ApJ*, 712, 260
 Küpper, A. H. W., Balbinot, E., Bonaca, A., et al. 2015, *ApJ*, 803, 80
 Lee, J., Lee, Y., Kim, J., et al. 2018, arXiv e-prints [arXiv:1810.00825]
 Lemos, P., Coogan, A., Hezaveh, Y., & Perreault-Levasseur, L. 2023, *40th International Conference on Machine Learning*, 202, 19256
 Lipman, Y., Chen, R. T. Q., Ben-Hamu, H., Nickel, M., & Le, M. 2022, arXiv e-prints [arXiv:2210.02747]
 Malhan, K., & Ibata, R. A. 2019, *MNRAS*, 486, 2995
 Mestre, M. F., Argüelles, C. R., Carpintero, D. D., Crespi, V., & Krut, A. 2024, *A&A*, 689, A194
 Miró-Carretero, J., Gómez-Flechoso, M. A., Martínez-Delgado, D., et al. 2025, *A&A*, 700, A176
 Nguyen, T., Mishra-Sharma, S., Williams, R., & Necib, L. 2023, *Phys. Rev. D*, 107, 043015
 Nibauer, J., & Bonaca, A. 2025, *ApJ*, 985, L22
 Nibauer, J., Belokurov, V., Cranmer, M., Goodman, J., & Ho, S. 2022, *ApJ*, 940, 22
 Nibauer, J., Bonaca, A., Spergel, D. N., et al. 2025, *ApJ*, 983, 68
 Obreja, A., Buck, T., & Macciò, A. V. 2022, *A&A*, 657, A15
 Oliver, W. H., & Buck, T. 2024, arXiv e-prints [arXiv:2411.03229]
 Oliver, W. H., Elahi, P. J., Lewis, G. F., & Buck, T. 2024, *MNRAS*, 530, 2637
 Palau, C. G., Wang, W., & Han, J. 2025, *MNRAS*, 539, 2718
 Perez, E., Strub, F., de Vries, H., Dumoulin, V., & Courville, A. 2017, arXiv e-prints [arXiv:1709.07871]
 Price-Whelan, A. M., Hogg, D. W., Johnston, K. V., & Hendel, D. 2014, *ApJ*, 794, 4
 Reino, S., Rossi, E. M., Sanderson, R. E., et al. 2022, *MNRAS*, 512, 4455
 Sante, A., Kawata, D., Font, A. S., & Grand, R. J. J. 2025, *MNRAS*, 542, 1776
 Saoulis, A. A., Piras, D., Jeffrey, N., et al. 2025, *MNRAS* [arXiv:2505.21215]
 Schmitt, M., Bürkner, P.-C., Köthe, U., & Radev, S. T. 2021, arXiv e-prints [arXiv:2112.08866]
 Skúladóttir, Á., Ernanandes, H., Feuillet, D. K., et al. 2025, *ApJ*, 986, L21
 Starkman, N., Price-Whelan, A., & Nibauer, J. 2024, <https://doi.org/10.5281/zenodo.11553485>
 Viterbo, G., & Buck, T. 2024, arXiv e-prints [arXiv:2411.17269]
 Viterbo, G., & Buck, T. 2025, arXiv e-prints [arXiv:2511.22468]
 Walder, M., Erkal, D., Collins, M., & Martínez-Delgado, D. 2025, *ApJ*, 994, 36
 Wildberger, J. B., Dax, M., Buchholz, S., et al. 2023, *Machine Learning for Astrophysics*, 34
 Zhou, L., Radev, S. T., Oliver, W. H., et al. 2025, *A&A*, 701, A44

Appendix A: Ablation study

Figure A.1 presents an ablation study examining how training and validation loss evolve as a function of simulation budget, defined as the fraction of the available training set used for model training. The ablation study demonstrates clear convergence behavior across all simulation budgets. Both training and validation losses decrease monotonically and stabilize after approximately 100–150 epochs, indicating successful model optimization. As expected, larger simulation budgets (50% and 100%) achieve lower final validation losses (~ 0.4 – 0.5), confirming that increased training data improves generalization performance. While comprehensive evaluation of an SBI pipeline requires more sophisticated metrics (see Sects. 3.1.1 and 3.1.2), this ablation study justifies our choice of training set size.

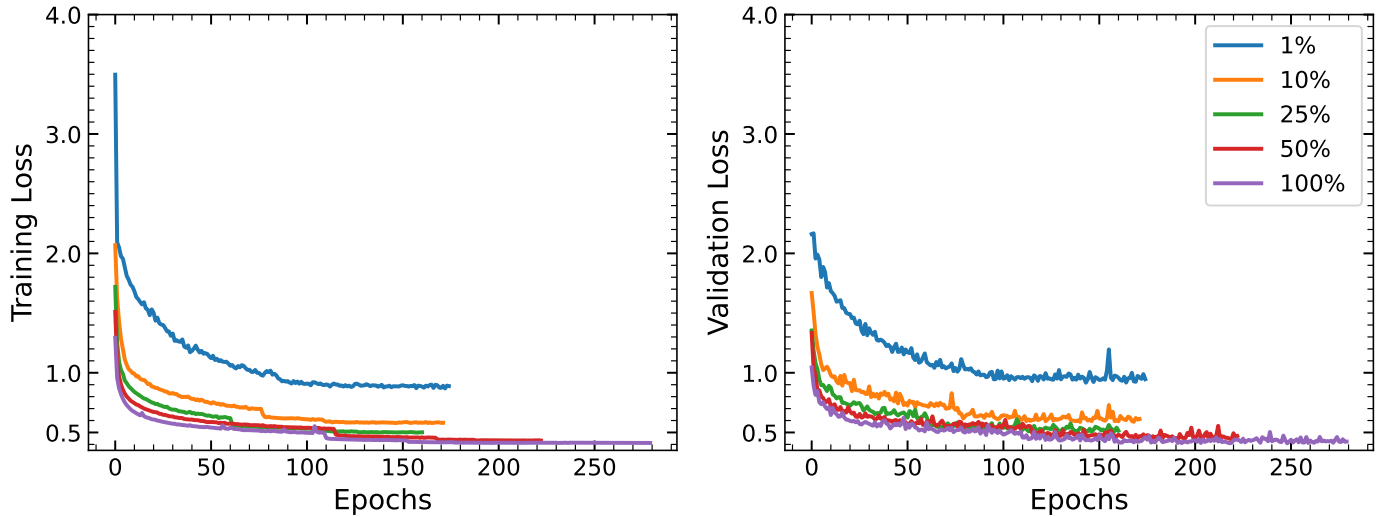


Fig. A.1. Ablation study showing training loss (left) and validation loss (right) as a function of training epochs for different simulation budgets (1%, 10%, 25%, 50%, and 100% of the full training set). All models converge within 100–150 epochs, with larger simulation budgets achieving better final performance.