A COMPASS TO BAYESIAN MODEL COMPARISON IN SIMULATOR-BASED SETTINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

Bayesian Model Comparison (BMC) is a cornerstone of scientific discovery, yet it remains a formidable challenge for complex, simulation-based models where likelihoods are intractable. Existing Simulation-Based Inference (SBI) methods primarily focus on parameter inference and can be computationally prohibitive to adapt for comparing multiple model hypotheses. While recent works have introduced highly flexible and powerful inference methods, a dedicated framework for robust and scalable BMC is still lacking. We introduce `compass`, a novel framework that leverages a conditional diffusion transformer to create an efficient, end-to-end pipeline specifically for BMC. By strategically masking inputs, `compass` uses a single, flexibly conditioned model per hypothesis to perform posterior estimation for parameter inference and likelihood estimation for model ranking. It incorporates a principled method for jointly inferring shared parameters from multiple observations, leading to a highly robust estimate of the maximized likelihood for model comparison. We demonstrate `compass` on two benchmark tasks and a challenging real-world astrophysics application, showing that it correctly identifies the ground-truth data-generating model and provides robust parameter constraints, even under model misspecification. Furthermore, we show that the model's internal attention mechanism is interpretable, providing novel scientific insights into the learned physical relationships that drive model selection. Our work provides a powerful, general-purpose tool for scientific discovery. The code is publicly available at `https://anonymous.4open.science/r/COMPASS-6CC6/`.

1 INTRODUCTION

In numerous scientific disciplines, computational simulations serve as indispensable tools for modelling complex systems. These simulators encapsulate our mechanistic understanding of the world, but their complexity often renders their likelihood functions intractable. This "likelihood-free" setting has given rise to Simulation-Based Inference (SBI) (Cranmer et al., 2020), a vibrant field of machine learning (ML) dedicated to inferring model parameters θ that could explain observed data x. While significant progress has been made in parameter inference, a more fundamental scientific question often precedes it: *"Given a set of competing simulators (or hypothesis) $\mathcal{M}_i$, which one best explains the data x?"* This is the problem of Bayesian Model Comparison (BMC). Solving it is crucial for adjudicating between scientific hypotheses, for example in ecology, where different models of population dynamics can be tested against time-series data or in astrophysics, where competing models of Galactic Chemical Evolution (GCE) based on different nucleosynthetic yield tables can lead to vastly different predictions (Buck et al., 2021; Günes et al., 2025). However, BMC is notoriously difficult in the SBI context. Traditional methods rely on estimating the marginal likelihood or "evidence", $P(x|\mathcal{M}_i)$, which requires integrating over all possible parameters—a computationally prohibitive task.

Recent work has demonstrated the potential of unified inference models to tackle such challenges. Notably, Gloeckler et al. (2024) introduced a pioneering 'all-in-one' approach using a diffusion transformer to learn the joint distribution $p(\theta, x)$ and sample arbitrary conditionals. This provides remarkable flexibility. However, many critical scientific workflows, such as BMC, do not require the full flexibility of sampling arbitrary conditionals but instead rely on a structured interplay between posterior inference (to find plausible parameters) and likelihood estimation (to evaluate model

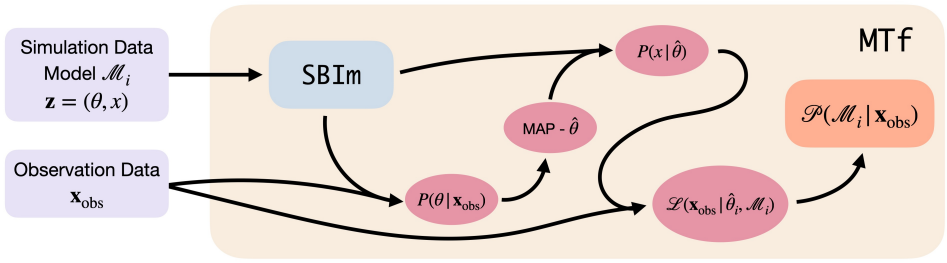

Figure 1: Flow chart of our model comparison workflow. Each candidate model ($\mathcal{M}_i$) is used to train a separate Score-Based Inference Model (SBIm). For a given observation $\mathbf{x}_{\text{obs}}$, the corresponding SBIm first infers the posterior $P(\theta|\mathbf{x}_{\text{obs}})$ in its NPE mode to find the MAP parameters $\hat{\theta}_i$. The same model is then switched to its NLE mode to estimate the maximized likelihood $L(\mathbf{x}_{\text{obs}}|\hat{\theta}_i, \mathcal{M}_i)$. The set of likelihoods from all models is used to derive the final model posterior probabilities $P(\mathcal{M}_i|\mathbf{x}_{\text{obs}})$, enabling a principled comparison.

evidence). This motivates our work: We build upon the foundation of the all-in-one SBI (Gloeckler et al., 2024) and ask whether we can design a framework that is optimized specifically for the common and crucial task of BMC, while also improving architectural scalability and interpretability.

Here, we introduce `compass` (Comparison Of Models using Probabilistic Assessment in Simulation-based Settings), a novel framework that specializes the all-in-one SBI concept for robust and scalable BMC. `compass` is built on a conditional score-based Diffusion Transformer (DiT Peebles & Xie, 2023), leveraging a state-of-the-art architecture for generative modelling. Our core contribution is the explicit and strategic use of attention masking to control the information flow within the transformer. This allows a single trained network to seamlessly switch between two operational modes: Neural Posterior Estimation (NPE) and Neural Likelihood Estimation (NLE). By masking either the parameters θ, the model infers the posterior $P(\theta|x)$ or by masking the observations x, the same model estimates the likelihood function $P(x|\theta)$. This unified approach unlocks our streamlined pipeline for BMC, shown in Figure 1.

Beyond performance, scientific adoption of complex models requires trust. `compass` provides this trust through interpretability. By inspecting the transformer's internal attention mechanism, we can verify that its reasoning is grounded in the physically meaningful relationships encoded by the simulator. Finally, to handle real-world scientific data, `compass` is equipped to deal with observational uncertainties through a statistically principled and efficient Monte Carlo approximation of the full parameter space. Thus, `compass` provides a fully open-source tool for principled BMC.

2 Preliminaries

Our framework builds upon a combination of ideas from BMC and SBI, leaning heavily on the capabilities offered by neural density estimation (NDE) and transformers. We therefore concisely review these topics to establish necessary context.

Bayesian Model Comparison. The objective of BMC is to select the most plausible model or hypothesis $\mathcal{M}_i$ from a set of competing models $\{\mathcal{M}_i\}$, given some observed data $\mathbf{x}_{\text{obs}}$. This selection is based on the model's posterior probability, which Bayes' theorem gives as

$$P(\mathcal{M}_i|x) \propto P(x|\mathcal{M}_i)P(\mathcal{M}_i) \tag{1}$$

where $P(\mathcal{M}_i)$ is the prior probability of the model and $P(x|\mathcal{M}_i)$ is the marginal likelihood or "evidence". The evidence requires integrating the likelihood $p(x|\theta, \mathcal{M}_i)$ over the entire parameter space θ of the model, $P(x|\mathcal{M}_i) = \int p(x|\theta, \mathcal{M}_i)p(\theta|\mathcal{M}_i)d\theta$. In simulator-based settings, this integral is almost always intractable.

Because computing the evidence directly is infeasible, model selection can be approximated using the maximized likelihood, $L_{\max}(x|\mathcal{M}_i) = \max_\theta \mathrm{p}(\mathrm{x}|\theta, \mathcal{M}_\mathrm{i})$. The Akaike Information Criterion (AIC) provides a principled framework for comparing models using this quantity (Akaike, 1978). For the common case of comparing models with the same number of parameters, this simplifies to a

direct ratio of their maximized likelihoods (Wagenmakers & Farrell, 2004), which can be converted into a posterior probability via a softmax function. This approach forms the basis of our BMC pipeline, as derived in Appendix A.

Simulation-Based Inference. SBI offers a powerful set of ML techniques to sidestep the intractability of the likelihood function (Cranmer et al., 2020). Instead of directly evaluating the likelihood, SBI leverages a simulator $\mathcal{M}$, a program that maps a set of parameters θ to a set of simulated observations x. Modern SBI methods train a neural network (NN) to approximate either the posterior (NPE) (Papamakarios & Murray, 2018) or likelihood distribution (NLE) (Papamakarios et al., 2021) based on samples from the simulator. By amortizing the inference task, these networks offer great potential to solve computationally intensive problems.

Score-based Diffusion Models. Score-based diffusion models (Song et al., 2020) are powerful generative models that learn a data distribution by training a NN to reverse a fixed noising process. This is defined by a forward-time stochastic differential equation (SDE) that gradually perturbs any data sample z_0 into Gaussian noise. A key result from Anderson (1982) shows that this process can be reversed in time by a corresponding reverse-time SDE, defined as

$$dz = [f(z,t) - g(t)^2 \nabla_z \log p_t(z)]dt + g(t)dw \tag{2}$$

where $f(z,t)$ and $g(t)$ are the drift and diffusion coefficients of the forward SDE, and dw is a reverse-time Wiener process. This equation shows that the generative process is entirely defined if one can provide the score function, $\nabla_z \log p_t(z)$. In diffusion models, a time-dependent NN $s_\phi(z_t, t)$ is trained to approximate this score function using a denoising score-matching objective.

Transformers and attention mechanisms. Transformers (Vaswani et al., 2023) have become the state-of-the-art architecture for processing sequential and set-structured data. Their power stems from the self-attention mechanism, which allows the model to dynamically weigh the importance of different input tokens when computing the representation for any given token. For a set of input tokens, attention computes an "attention map" quantifying the information flow between all pairs of tokens. This inherent mechanism provides a powerful tool for model interpretability, allowing us to inspect which inputs the model focuses on during inference. Recent work has successfully combined transformers with diffusion models, most notably the Diffusion Transformer (DiT) (Peebles & Xie, 2023), which has achieved state-of-the-art performance in generative modelling by replacing the traditional U-Net backbone with a scalable transformer architecture.

3 THE COMPASS FRAMEWORK

Our goal is to perform BMC for a set of competing simulators $\{\mathcal{M}_i\}$ and infer the parameters for the best performing model. This requires estimating the posterior model probability $P(\mathcal{M}_i|\mathbf{x}_{\text{obs}})$ given some observed data $\mathbf{x}_{\text{obs}}$. Since the model evidence $P(\mathbf{x}_{\text{obs}}|\mathcal{M}_i)$ is intractable, we follow an approach based on the maximized likelihood, which provides a robust approximation for model ranking (see Appendix A for derivation). This requires a framework that can both infer the parameters that maximize the likelihood and then evaluate that likelihood. `compass` achieves this with a single, unified model per hypothesis.

3.1 PROBLEM SETUP

Let a simulator $\mathcal{M}$ be a stochastic process that generates an observation $x \in R^{D_x}$ given a set of parameters $\theta \in R^{D_\theta}$. Our framework learns the joint distribution $p(\theta, x)$. We define a joint vector $z = (\theta, x)$ with total dimensionality $D_z = D_\theta + D_x$. The core of our method is a single generative model that can estimate any conditional distribution $p(z_A|z_B)$, where z_A and z_B are arbitrary disjoint subsets of the elements in z. For the BMC task, we are interested in two specific conditionals:

- **The Posterior Distribution**, $P(\theta|x)$: This is required to find the Maximum A Posteriori (MAP) parameters $\hat{\theta}$ that best explain an observation $\mathbf{x}_{\text{obs}}$.
- **The Likelihood Distribution**, $P(x|\theta)$: This is required to evaluate the likelihood of the observation $\mathbf{x}_{\text{obs}}$ at the MAP estimate.

`compass` is designed to estimate both of these conditionals with a single, flexibly conditioned NN.

3.2 The Conditional Diffusion Transformer

We model the joint distribution $p(z) = p(\theta, x)$ using a score-based diffusion model. This class of models learns to reverse a noising process that gradually transforms data into pure Gaussian noise.

Diffusion Process. We employ a Variance Exploding (VE) SDE (Song et al., 2020), which defines a forward process that perturbs an initial data point z_0 into a noise distribution over a continuous time variable $t \in [0, 1]$. The process is defined by the SDE $dz = \sigma^t dw$, where $\sigma > 1$ is a hyperparameter and w is a standard Wiener process. The perturbed data distribution $p_t(z_t|z_0)$ at any time t is a Gaussian with a tractable form.

Reverse Process. The generative process is defined by the corresponding reverse-time SDE (Anderson, 1982):

$$dz = [-\sigma^{2t} \nabla_{z_t} \log p_t(z_t)]dt + \sigma^t dw \tag{3}$$

where dw is a reverse-time Wiener process. Generating samples from $p_0(z)$ requires estimating the time-dependent score, $\nabla_{z_t} \log p_t(z_t)$, at each step of the reverse process.

Conditional Score Network. We approximate the score using a NN $s_\phi(z_t, M_c, t)$. The key to our unified framework is that we condition the score not just on time t, but also on a binary condition mask $M_c \in \{0, 1\}^D$. This mask specifies which elements of z_t are latent ($M_c^j = 0$) and which are observed or conditioned during the inference ($M_c^j = 1$).

Our score network is a Diffusion Transformer (DiT) based on the architecture of Peebles & Xie (2023), which has demonstrated state-of-the-art performance in generative modelling. A key architectural choice that distinguishes `compass` is our handling of the input data $z = (\theta, x)$. Instead of using fixed tokens and learned ID embeddings for each variable, we treat the input as a structured sequence where each position corresponds to a specific physical quantity. Each of these node inputs is then projected into the transformer's hidden dimension via its own learned linear embedding. This approach is particularly well-suited for the structured, continuous data prevalent in scientific simulators. Furthermore, this design makes the transformer's attention mechanism highly interpretable. The attention weights directly quantify the learned relationships between specific, named physical variables (e.g., between a model parameter and an observational feature), as we will demonstrate in Section 4.4. The diffusion time t is incorporated into the transformer blocks using an adaLN-Zero conditioning scheme (Perez et al., 2017; Peebles & Xie, 2023) after a Gaussian-Fourier embedding. Crucially, we modify the self-attention mechanism with a our custom attention mask, derived from the mask M_c, that prevents latent tokens from attending to other latent tokens. This enforces the conditional structure, ensuring that the denoising process for a latent variable is only informed by the observed variables (the exact architecture is described in Appendix B.1). Our custom conditioning mechanism allows the single network s_ϕ to operate in two distinct modes:

- **NPE Mode**: To estimate $p(\theta|x)$ we set the attention mask M_c to $(0, \ldots, 0, 1, \ldots, 1)$, where the 0's correspond to θ and 1's to x. The network then generates samples of θ conditioned on a fixed x.
- **NLE Mode**: To estimate $p(x|\theta)$ we set M_c to $(1, \ldots, 1, 0, \ldots, 0)$. The network then generates samples of x conditioned on a fixed θ.

3.3 The End-to-End BMC Pipeline

The dual-mode capability of our conditional Diffusion Transformer unlocks a streamlined and robust end-to-end pipeline for BMC, as illustrated in Figure 1. This process is designed to move from simulated data to a principled, data-driven ranking of competing scientific hypotheses, while also properly accounting for real-world observational uncertainties.

After a dedicated `compass` instance, s_{ϕ_i}, has been trained for each competing simulator, $\mathcal{M}_i$, the first step is to identify the most plausible parameters for each model ,given a set of real-world observations $\mathbf{x}_{\text{obs}}$. To achieve this, we operate each trained network s_{ϕ_i} in its NPE mode. Conditioned

on the observations, we generate an ensemble of samples from the posterior $P(\theta|\mathbf{x}_{\text{obs}}, \mathcal{M}_i)$ and identify the MAP estimate, $\hat{\theta}_i$, which represents the most probable set of parameters for that model.

With the MAP $\hat{\theta}_i$ identified for each hypothesis, the next step is to evaluate how well each model explains the data at its best-fit point. This requires estimating the maximized likelihood, $L(\mathbf{x}_{\text{obs}}|\hat{\theta}_i)$. We achieve this by switching the exact same trained network s_{ϕ_i} into its NLE mode. Conditioned on its respective $\hat{\theta}_i$, the model now generates a sample distribution of possible observations $x \sim p(x|\hat{\theta}_i)$. To obtain a continuous and differentiable estimate of the likelihood function from these samples, we fit a Kernel Density Estimator (KDE) (Parzen, 1962; Rosenblatt, 1956) to the generated distribution. The maximized likelihood is then calculated by evaluating this KDE at the actual observation, $\mathbf{x}_{\text{obs}}$.

A unique and critical feature of `compass` is its principled handling of observational uncertainties. Real scientific data is never a single point value $\mathbf{x}_{\text{obs}}$ but is more accurately described by a distribution, such as a Gaussian $\mathcal{N}(\mathbf{x}_{\text{obs}}, \mathbf{\Sigma}_{\text{obs}})$ centered on the measurement $\mathbf{x}_{\text{obs}}$ with a covariance $\mathbf{\Sigma}_{\text{obs}}$ representing the measurement error. To correctly propagate this uncertainty, `compass` employs a Monte Carlo approximation. Instead of conditioning on the single point $\mathbf{x}_{\text{obs}}$, we draw a large number of data realizations $x_j \sim \mathcal{N}(\mathbf{x}_{\text{obs}}, \mathbf{\Sigma}_{\text{obs}})$ from the uncertainty distribution. The entire inference process—from MAP estimation to likelihood evaluation—is performed for each realization x_j. This approach effectively marginalizes over the observational uncertainties, ensuring that the resulting model comparison is robust and the inferred parameter constraints are not over-confident.

Finally, with a robust estimate of the maximized log-likelihood for each competing model $\mathcal{M}_i$, the posterior probability for each model is computed via a softmax function (Appendix A), yielding a definitive, data-driven ranking. For analyses involving multiple independent observations, the total log-likelihood for each model is simply the sum of the individual marginalized log-likelihoods before the final softmax is applied, allowing evidence to be naturally aggregated.

4 Experiments

4.1 Toy Problem: Misspecified Gaussians

To first validate the core logic of our end-to-end BMC pipeline, we designed a simple toy problem with three competing simulators. The goal is to test whether `compass` can correctly identify the true data-generating process in a controlled environment where the ground truth is known and the models are subtly different.

Problem Setup. We define three hypotheses (simulators) that map a single latent parameter θ to a two-dimensional observation $\mathbf{x} = (x_1, x_2)$. For all hypotheses, the parameters θ are sampled from the same prior distribution $\theta \sim \mathcal{N}(0, 3^2)$. Hypothesis 1 is the ground truth, where the observation x_1 has a sinusoidal dependency on the parameter θ. In the alternative hypothesis 2, the dependency of x_1 to θ is phase-shifted to a co-sinusoidal relationship. This represents a plausible but incorrect model, posing a classical model selection challenge. The third hypothesis represents a baseline model, where the observations x_1 and x_2 are drawn from a distribution independent of θ. This acts as a crucial baseline to test whether the framework can reject a model with no explanatory power. The exact generative equations for all hypotheses are provided in Appendix D.1.

Task and Training. We trained a separate `compass` instance for each hypothesis, using 10^5 simulated data pairs $(\theta, \mathbf{x})$ each. We then generated a test set of 100 mock observations from Hypothesis 1 and tasked `compass` with computing the posterior model probabilities $P(\mathcal{M}_i|\mathbf{x}_{\text{obs}})$ for all three competing models.

Results. Figure 2 shows the result of the model comparison. The violin plot (left) shows the distribution of single-observation posterior probabilities. For the majority of individual observations, the framework correctly assigns the highest probability to the true model (Hypothesis 1), demonstrating strong discriminatory power at the single-sample level. The cumulative plot (right) shows how the joint model posterior evolves as evidence from multiple observations is aggregated. After incorpo-

rating all observations, `compass` reaches near-certainty ($P(\mathcal{M}_1|\mathbf{x}_{\text{obs}}) \approx 1.0$), while the posterior probabilities for the incorrect models correctly decay towards zero.

This result confirms that our BMC pipeline operates as intended, it effectively distinguishes between structurally similar models, robustly rejects a null hypothesis, and principledly aggregates evidence from multiple observations to reach a decisive conclusion.

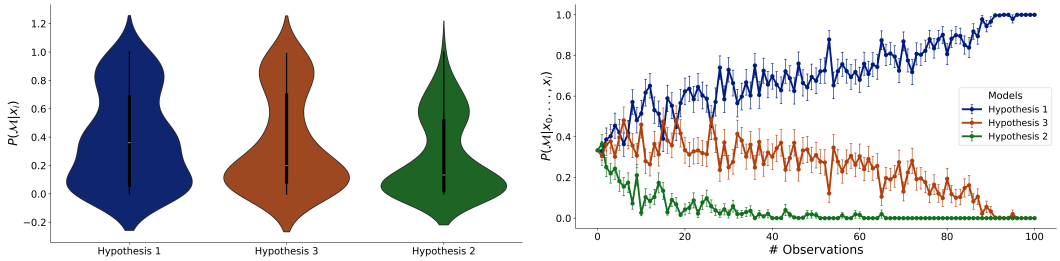

Figure 2: Validation of the `compass` BMC pipeline on a controlled toy problem. **Left**: Violin plot showing the distribution of single-observation posterior model probabilities. The framework correctly assigns the highest probability to the true model (Hypothesis 1) for most individual samples. **Right**: Cumulative model posterior probability as more observations are aggregated. After ~ 80 observations, `compass` reaches near-certainty in the correct model, demonstrating its ability to distinguish similar hypotheses and robustly aggregate evidence.

4.2 Toy Problem: Population Dynamics

To demonstrate the applicability of `compass` to time-series data and dynamical systems, we apply it to the classic ecological problem of predator-prey population dynamics. This setting is ideal for model comparison, as several competing mathematical models with different underlying assumptions (e.g., resource limits, predator saturation) exist to describe the same phenomenon.

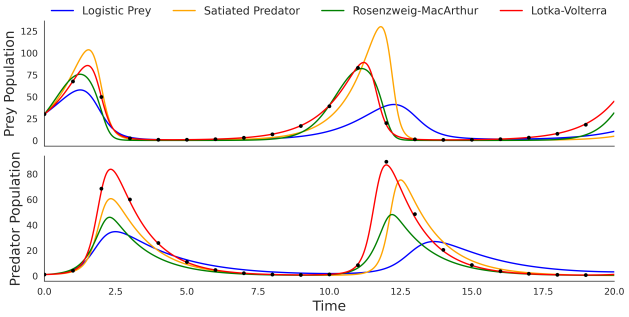

Figure 3: Posterior predictive check. Black dots represent the mock observed time-series data, generated from the Lotka-Volterra model. Coloured lines show the simulated trajectories for each of the four competing models using their respective MAP parameters, inferred by `compass`.

Problem Setup. We consider four competing models of population dynamics, each described by a system of ordinary differential equations (ODEs): (1) the classic Lotka-Volterra model, (2) a Logistic Prey model where the prey population has a carrying capacity, (3) a Satiated Predator model where the predator's consumption rate saturates (Holling Type II), and (4) the Rosenzweig-MacArthur model, which combines both logistic prey growth and predator satiation. For this test, we generate a single mock observation time-series from the Lotka-Volterra model (the ground truth). The task for `compass` is to correctly identify the data-generating model from the four candidates based on this time-series data. Each model has four free parameters $(\alpha, \beta, \gamma, \delta)$ controlling the interaction rates, which we infer. Further details on the model equations and simulation setup are provided in Appendix D.2.

Results. `compass` successfully recovers the correct data-generating process. With the single time-series observation as input, it chooses the Lotka-Volterra model, decisively rejecting the three incorrect hypotheses, and correctly recovers the parameters that describe the system (Fig. 3).

Figure 4 provides a deeper look into the likelihood evaluation itself. For each time point in the observation, we plot the full likelihood distribution $P(x|\hat{\theta}, \mathcal{M}_i)$ generated by `compass` for each of the four models, alongside the true measurement. This visualization confirms that for the incorrect models, the actual observations consistently fall in the low-probability tails of their predicted distri-

butions, leading to a vanishingly small total likelihood. This demonstrates that `compass`'s decision is grounded in a quantitative and robust assessment of how well each model predicts the data at its best-fit point.

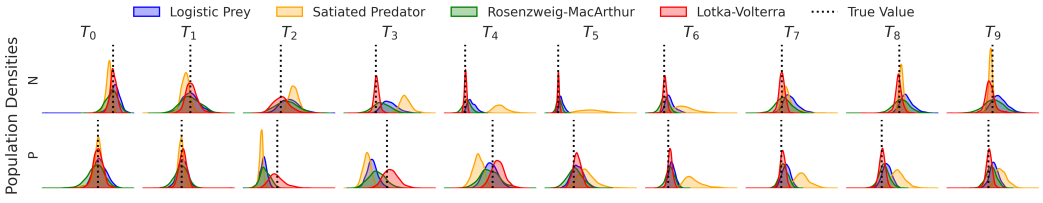

Figure 4: Likelihood evaluation for competing population models. For each time point T_i in the observation, the plot shows the likelihood distributions $P(x|\hat{\theta}, \mathcal{M}_i)$ generated by `compass` in NLE mode for all models. The vertical dashed lines indicate the true measured values which consistently falls in the high-probability region of the distributions predicted by the Lotka-Volterra model, leading to a high maximized likelihood and therefore the subsequent selection by the BMC pipeline.

4.3 Astrophysics Application: Galactic Chemical Enrichment

To demonstrate the power and scalability of `compass` on a real-world scientific problem, we apply it to a central challenge in astrophysics, modelling Galactic Chemical Evolution (GCE). GCE simulators aim to predict the chemical abundances of stars based on complex physical theories of star formation, nuclear fusion and stellar death. However, multiple competing physical models exist, and key parameters within these models are poorly constrained. This makes it an ideal testbed for joint model comparison and parameter inference.

Setup. Our goal is twofold: first, to select the most plausible physical model from a large set of candidates (BMC), and second, to infer key astrophysical parameters using that best model (SBI). As simulator we use the one-zone GCE model `chempy` (Rybizki et al., 2017), which models the chemical enrichment of stellar populations over time. A major uncertainty in GCE is the modelling of the nuclear fusion process inside the cores of stars. As a result, the theoretical "chemical yields" that specify which chemical elements are produced by different types of stars are highly uncertain. We construct a set of 40 competing GCE models, each corresponding to a unique combination of chemical yields for AGB stars and core-collapse supernovae (CC-SN) sourced from the astrophysics literature. The task is to identify which of these 40 physical models best explains the observational data given as the chemical composition of individual stars in the Milky Way derived from spectroscopic observations. For each model, we infer a set of six physical parameters. These include two global galactic parameters of primary scientific interest—the high-mass slope of the Initial Mass Function (α_{IMF}) and the normalization of Type Ia supernovae ($\log_{10}(N_{Ia})$)—as well as four local nuisance parameters describing the specific star-forming environment (see Appendix D.3 for details). As observational data $\mathbf{x}_{\text{obs}}$ we use a high-precision dataset of chemical abundances for 69 solar-type stars from Nissen et al. (2020). For each star, the data consists of an age estimate and the abundances of carbon (C), iron (Fe), magnesium (Mg), oxygen (O) and silicon (Si). Model calibration and robustness tests under model misspecification can be found in Appendix D.3.

Results. Figure 5 shows the results of the model comparison on real observational data. The left figure shows the relative probability distribution of a single observation for each of the 40 competing models. The cumulative posterior probability plot (right) reveals a decisive preference for one specific physical model: a combination of NuGrid AGB yields (Ritter et al., 2018) and TNG CC-SN yields (Pillepich et al., 2017). After aggregating evidence across all 69 stars, this model achieves a posterior probability approaching 100%, robustly ruling out the other 39 competing theories. This result demonstrates `compass`'s ability to adjudicate between a large number of complex scientific models in a principled, data-driven manner.

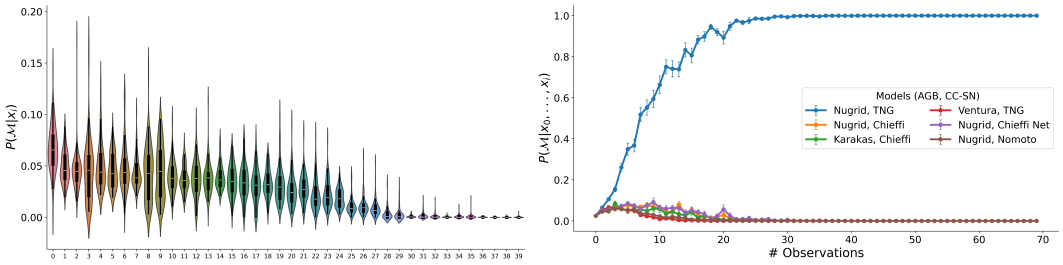

Figure 5: Model comparison results for 40 competing GCE models using real observational data. **Left**: Single-observation posteriors for all 40 models, sorted by median probability. **Right**: The cumulative model posterior probability. Evidence decisively favours a single physical model (NuGrid AGB + TNG CC-SN yields), which reaches near 100% relative posterior probability after aggregating data from all 69 stars.

Using this best-performing model, we then inferred the global galactic parameters. Figure 13 in the Appendix shows the final joint posterior for α_{IMF} and $\log_{10}(N_{\text{Ia}})$.

4.4 Interpretability with Attention Maps

A critical requirement for deploying machine learning models in scientific discovery is trust. To ensure that `compass` is not merely a "black box" exploiting simulator artifacts, but is instead learning scientifically meaningful relationships, we can investigate its internal reasoning process. The transformer architecture at the core of `compass` provides a natural mechanism for this through its attention weights. By analysing these weights, we can visualize the flow of information and understand which observational features the model deems important for inferring specific parameters.

Methodology. We visualize the attention patterns from the GCE model identified as the most plausible in Section 4.3. Following recent best practices (Helbling et al., 2025; Yeh et al., 2023), we analyse the attention weights at a representative midpoint of the diffusion process ($t = 0.5$), where the model's internal representations are most structured. The weights are averaged across all observations to reveal the model's systematic inference strategy. An attention weight from a parameter (query) to an observation (key) indicates the relative importance of that observation for inferring the parameter.

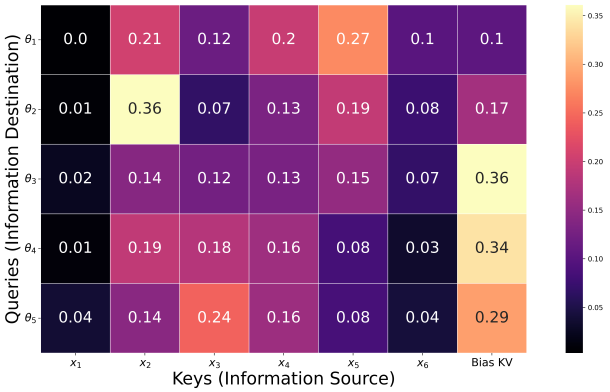

Figure 6: Layer-averaged attention map for the best-performing GCE model, revealing the learned inference strategy. The rows (queries) are the parameters being inferred, and the columns (keys) are the observational data and a learned bias term.

Results. Figure 6 shows the layer-averaged attention map, revealing the dominant information pathways learned by the model. The analysis demonstrates a sophisticated and physically-grounded inference strategy. For instance, to infer θ_1 (α_{IMF} - controls the rate of massive star explosions in the simulator), the model correctly learns to focus on the abundances of Oxygen and Magnesium (x_4, x_5), the primary chemical products of massive stars. More remarkably, to infer $\log_{10}(N_{\text{Ia}})$ (θ_2), the model discovers a non-obvious strategy. It largely ignores the canonical tracer element (Iron, x_3) in favour of a less ambiguous one (Carbon, x_2), effectively identifying a cleaner signal within the complex simulation hence allowing for deeper physical insights as well. Finally, for parameters that are poorly constrained by the data, the model learns to rely on a learned bias term (Bias KV), indicating a robust mechanism for handling uncertainty. This confirms that `compass` can autonomously discover valid and efficient inference strategies directly from simulated data.

The layer-by-layer evolution of these attention patterns (see Figure 14 in the Appendix) further reveals a hierarchical reasoning process, where the model's focus sharpens from broad features to specific, multi-element relationships in deeper layers. This deep interpretability analysis confirms that `compass`'s inferences are not arbitrary but are derived from a learned strategy that aligns with—and in the case of Carbon, provides new insights into—the physical relationships encoded in the simulator. This capability is crucial for building trust and validating the scientific conclusions drawn from complex, simulation-based inference frameworks.

5 Discussion

In this work, we addressed the critical challenge of BMC in simulation-based settings, where intractable likelihoods render traditional methods infeasible. We introduced `compass`, a novel framework that specializes the recent "all-in-one" SBI paradigm to create a robust and scalable end-to-end pipeline specifically for comparing competing scientific hypotheses. By leveraging a conditional Diffusion Transformer with a strategic masking mechanism, `compass` unifies NPE and NLE within a single model per hypothesis. This design provides a streamlined workflow for what is often a cumbersome, multi-stage process.

Our experimental results demonstrate that this specialized approach is both effective in principle and powerful in practice. On two controlled toy problems, `compass` flawlessly identified the ground-truth data-generating process, showcasing the core logic of our BMC pipeline. When applied to a complex grand challenge in astrophysics, it successfully adjudicated between 40 competing physical models of Galactic Chemical Evolution, identifying a single, best-fitting model with near-certainty from real observational data.

Beyond performance, we established that `compass` is not a black box. The transformer architecture's inherent attention mechanism provides a window into the model's internal reasoning. Our analysis of these attention patterns on the GCE problem confirmed that `compass` learns physically-grounded inference strategies, such as correctly identifying the most informative chemical elements for constraining specific astrophysical parameters. In one instance, it even discovered a non-obvious but more robust inference strategy than the one typically used by domain experts. This interpretability is crucial for building trust and enabling the scientific validation of a model's outputs.

Limitations. While `compass` provides a powerful and robust framework, it is important to acknowledge its inherent limitations, many of which are shared by methods built on similar underlying technologies. Like other diffusion-based inference methods, sample generation in `compass` is an iterative process. We must numerically solve the reverse SDE, which is computationally more intensive than the single-pass generation of methods based on normalizing flows (Greenberg et al., 2019). This results in a trade-off: while `compass` avoids the slow convergence of traditional MCMC-based approaches and can achieve accurate inference with a practical number of evaluation steps (as shown in our SDE solver analysis in Appendix B), it does not match the millisecond-level inference speed of flow-based NPEs. Furthermore, like many deep learning models, the performance of `compass` is sensitive to the choice of the transformer architecture and key hyperparameters. Factors such as network depth, hidden size, and the diffusion noise schedule significantly influence posterior accuracy. While we provide a well-performing default configuration, applying `compass` to new problems may require dedicated tuning using standard procedure such as `optuna` (Akiba et al., 2019).

6 Conclusion

We have introduced `compass`, a novel framework that makes robust BMC practical for complex, simulation-based models. By specializing the 'all-in-one' inference paradigm with a conditional Diffusion Transformer, `compass` provides a unified, scalable, and end-to-end pipeline for both model selection and parameter inference. Our experiments demonstrated its ability to correctly identify ground-truth models, provide robust parameter estimates and offer crucial interpretability into its internal reasoning. By providing an open-source tool [1] that is both powerful and trustworthy, `compass` helps bridge the gap between complex simulations and principled scientific discovery, paving the way for data-driven validation of scientific theories in the likelihood-free era.

[1] `https://anonymous.4open.science/r/COMPASS-6CC6/`

REFERENCES

H. Akaike. On newer statistical approaches to parameter estimation and structure determination. *IFAC Proceedings Volumes*, 11(1):1877–1884, 1978. ISSN 1474-6670. doi: https://doi.org/10.1016/S1474-6670(17)66162-7. URL https://www.sciencedirect.com/science/article/pii/S1474667017661627. 7th Triennial World Congress of the IFAC on A Link Between Science and Applications of Automatic Control, Helsinki, Finland, 12-16 June.

Hirotugu Akaike. A bayesian extension of the minimum aic procedure of autoregressive model fitting. *Biometrika*, 66(2):237–242, 08 1979. ISSN 0006-3444. doi: 10.1093/biomet/66.2.237. URL https://doi.org/10.1093/biomet/66.2.237.

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework, 2019. URL https://arxiv.org/abs/1907.10902.

Brian D.O. Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982. ISSN 0304-4149. doi: https://doi.org/10.1016/0304-4149(82)90051-5. URL https://www.sciencedirect.com/science/article/pii/0304414982900515.

F. Bigiel, A. Leroy, F. Walter, E. Brinks, W. J. G. de Blok, B. Madore, and M. D. Thornley. The Star Formation Law in Nearby Galaxies on Sub-Kpc Scales. , 136:2846–2871, December 2008. doi: 10.1088/0004-6256/136/6/2846.

Hamparsum Bozdogan. Model selection and akaike's information criterion (aic): The general theory and its analytical extensions. *Psychometrika*, 52(3):353–356, 1987. doi: 10.1007/BF02294361.

Tobias Buck, Jan Rybizki, Sven Buder, Aura Obreja, Andrea V Macciò, Christoph Pfrommer, Matthias Steinmetz, and Melissa Ness. The challenge of simultaneously matching the observed diversity of chemical abundance patterns in cosmological hydrodynamical simulations. *Monthly Notices of the Royal Astronomical Society*, 508(3):3365–3387, September 2021. ISSN 1365-2966. doi: 10.1093/mnras/stab2736. URL http://dx.doi.org/10.1093/mnras/stab2736.

Tobias Buck, Berkay Günes, Giuseppe Viterbo, William H. Oliver, and Sven Buder. Inferring galactic parameters from chemical abundances with simulation-based inference, 2025. URL https://arxiv.org/abs/2503.02456.

K.P. Burnham and D.R. Anderson. *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Verlag, 2002.

G. Chabrier. Galactic Stellar and Substellar Initial Mass Function. , 115:763–795, July 2003. doi: 10.1086/376392.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, May 2020. ISSN 1091-6490. doi: 10.1073/pnas.1912789117. URL http://dx.doi.org/10.1073/pnas.1912789117.

Aaron Defazio, Xingyu Alice Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, and Ashok Cutkosky. The road less scheduled, 2024. URL https://arxiv.org/abs/2405.15682.

Manuel Gloeckler, Michael Deistler, Christian Weilbach, Frank Wood, and Jakob H. Macke. All-in-one simulation-based inference, 2024. URL https://arxiv.org/abs/2404.09636.

David S. Greenberg, Marcel Nonnenmacher, and Jakob H. Macke. Automatic posterior transformation for likelihood-free inference, 2019. URL https://arxiv.org/abs/1905.07488.

Berkay Günes, Sven Buder, and Tobias Buck. A compass to model comparison and simulation-based inference in galactic chemical evolution, 2025. URL https://arxiv.org/abs/2507.05060.

Alec Helbling, Tuna Han Salih Meral, Ben Hoover, Pinar Yanardag, and Duen Horng Chau. Conceptattention: Diffusion transformers learn highly interpretable features, 2025. URL https://arxiv.org/abs/2502.04320.

Pablo Lemos, Adam Coogan, Yashar Hezaveh, and Laurence Perreault-Levasseur. Sampling-based accuracy testing of posterior estimators for general inference, 2023. URL https://arxiv.org/abs/2302.03026.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps, 2022. URL https://arxiv.org/abs/2206.00927.

Jan-Matthis Lueckmann, Jan Boelts, David S. Greenberg, Pedro J. Gonçalves, and Jakob H. Macke. Benchmarking simulation-based inference, 2021. URL https://arxiv.org/abs/2101.04653.

D. Maoz and F. Mannucci. Type-Ia Supernova Rates and the Progenitor Problem: A Review. , 29: 447–465, January 2012. doi: 10.1071/AS11052.

P. E. Nissen, J. Christensen-Dalsgaard, J. R. Mosumgaard, V. Silva Aguirre, E. Spitoni, and K. Verma. High-precision abundances of elements in solar-type stars: Evidence of two distinct sequences in abundance-age relations. *Astronomy amp; Astrophysics*, 640:A81, August 2020. ISSN 1432-0746. doi: 10.1051/0004-6361/202038300. URL http://dx.doi.org/10.1051/0004-6361/202038300.

George Papamakarios and Iain Murray. Fast ϵ-free inference of simulation models with bayesian conditional density estimation, 2018. URL https://arxiv.org/abs/1605.06376.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference, 2021. URL https://arxiv.org/abs/1912.02762.

Emanuel Parzen. On Estimation of a Probability Density Function and Mode. *The Annals of Mathematical Statistics*, 33(3):1065 – 1076, 1962. doi: 10.1214/aoms/1177704472. URL https://doi.org/10.1214/aoms/1177704472.

William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023. URL https://arxiv.org/abs/2212.09748.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer, 2017. URL https://arxiv.org/abs/1709.07871.

Annalisa Pillepich, Volker Springel, Dylan Nelson, Shy Genel, Jill Naiman, Rüdiger Pakmor, Lars Hernquist, Paul Torrey, Mark Vogelsberger, Rainer Weinberger, and Federico Marinacci. Simulating galaxy formation with the illustristng model. *Monthly Notices of the Royal Astronomical Society*, 473(3):4077–4106, 10 2017. ISSN 0035-8711. doi: 10.1093/mnras/stx2656. URL https://doi.org/10.1093/mnras/stx2656.

C Ritter, F Herwig, S Jones, M Pignatari, C Fryer, and R Hirschi. Nugrid stellar data set – ii. stellar yields from h to bi for stellar models with mzams = 1–25
$rmM_{\odot}$ and z = 0.0001–0.02. *Monthly Notices of the Royal Astronomical Society*, 480(1): 538–571, June 2018. ISSN 1365-2966. doi: 10.1093/mnras/sty1729. URL http://dx.doi.org/10.1093/mnras/sty1729.

Murray Rosenblatt. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, 27(3):832 – 837, 1956. doi: 10.1214/aoms/1177728190. URL https://doi.org/10.1214/aoms/1177728190.

Jan Rybizki, Andreas Just, and Hans-Walter Rix. Chempy: A flexible chemical evolution model for abundance fitting: Do the sun's abundances alone constrain chemical evolution models? *Astronomy amp; Astrophysics*, 605:A59, September 2017. ISSN 1432-0746. doi: 10.1051/0004-6361/201730522. URL http://dx.doi.org/10.1051/0004-6361/201730522.

Ivo R. Seitenzahl, Franco Ciaraldi-Schoolmann, Friedrich K. Röpke, Michael Fink, Wolfgang Hillebrandt, Markus Kromer, Rüdiger Pakmor, Ashley J. Ruiter, Stuart A. Sim, and Stefan Taubenberger. Three-dimensional delayed-detonation models with nucleosynthesis for type ia supernovae. *Monthly Notices of the Royal Astronomical Society*, 429(2):1156–1172, 12 2012. ISSN 0035-8711. doi: 10.1093/mnras/sts402. URL https://doi.org/10.1093/mnras/sts402.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-Based Generative Modeling through Stochastic Differential Equations. *arXiv e-prints*, art. arXiv:2011.13456, November 2020. doi: 10.48550/arXiv.2011.13456.

P. G. van Dokkum, J. Leja, E. J. Nelson, S. Patel, R. E. Skelton, I. Momcheva, G. Brammer, K. E. Whitaker, B. Lundgren, M. Fumagalli, C. Conroy, N. Förster Schreiber, M. Franx, M. Kriek, I. Labbé, D. Marchesini, H.-W. Rix, A. van der Wel, and S. Wuyts. The Assembly of Milky-Way-like Galaxies Since z ˜ 2.5. , 771:L35, July 2013. doi: 10.1088/2041-8205/771/2/L35.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Eric-Jan Wagenmakers and Simon Farrell. Aic model selection using akaike weights. *Psychonomic Bulletin & Review*, 11(1):192–196, 2004. doi: 10.3758/BF03206482. URL https://doi.org/10.3758/BF03206482.

Catherine Yeh, Yida Chen, Aoyu Wu, Cynthia Chen, Fernanda Viégas, and Martin Wattenberg. Attentionviz: A global view of transformer attention, 2023. URL https://arxiv.org/abs/2305.03210.

A Derivation of Model Posterior

Model comparison typically relies on model evidence. However, in simulator-based settings where the likelihood function is often intractable, calculating model evidence directly is not feasible. An alternative approach for model comparison, particularly suitable for non-nested models, is to use the Akaike Information Criterion (AIC) (Akaike, 1978). AIC estimates the prediction error and thereby the relative quality of statistical models for a given set of data.

Given the maximized log-likelihood for model $\mathcal{M}_i$, denoted as $\ln \mathcal{L}(\mathbf{x}|\hat{\theta}_i, \mathcal{M}_i)$ (where $\hat{\theta}_i$ are the parameter values that maximize the likelihood for model $\mathcal{M}_i$), the AIC is defined as:

$$\text{AIC}_i = -2 \ln \mathcal{L}(\mathbf{x}|\hat{\theta}_i, \mathcal{M}_i) + 2k_i \tag{4}$$

where k_i is the number of estimable parameters in model $\mathcal{M}_i$. The model with the lowest AIC is generally preferred.

To compare a set of N models, we first calculate the AIC difference for each model $\mathcal{M}_i$ relative to the model with the minimum AIC in the set ($\text{AIC}_{\min}$): (Wagenmakers & Farrell, 2004; Akaike, 1978; 1979; Bozdogan, 1987):

$$\Delta_i(\text{AIC}) = \text{AIC}_i - \text{AIC}_{\min} \tag{5}$$

The likelihood of model $\mathcal{M}_i$ being the best model (in the Kullback-Leibler information sense), given the data $\mathbf{x}$, can be estimated relative to the other models using these AIC differences. The relative likelihood of model $\mathcal{M}_i$, sometimes called an "Akaike weight", is given by (Wagenmakers & Farrell, 2004; Burnham & Anderson, 2002):

$$\mathcal{L}_{\text{rel}}(\mathcal{M}_i|\mathbf{x}) \propto \exp\left(-\frac{1}{2}\Delta_i(\text{AIC})\right) \tag{6}$$

To obtain posterior probabilities for each model, $\mathcal{P}(\mathcal{M}_i|\mathbf{x})$, these relative likelihoods are normalized by summing over all models in the candidate set:

$$\mathcal{P}(\mathcal{M}_i|\mathbf{x}) = \frac{\exp\left(-\frac{1}{2}\Delta_i(\text{AIC})\right)}{\sum_{j=1}^{N} \exp\left(-\frac{1}{2}\Delta_j(\text{AIC})\right)} \tag{7}$$

Substituting equation 5 into equation 7:

$$\mathcal{P}(\mathcal{M}_i|\mathbf{x}) = \frac{\exp\left(-\frac{1}{2}(\text{AIC}_i - \text{AIC}_{\min})\right)}{\sum_{j=1}^{N} \exp\left(-\frac{1}{2}(\text{AIC}_j - \text{AIC}_{\min})\right)} \tag{8}$$

$$= \frac{\exp\left(-\frac{1}{2}\text{AIC}_i\right)}{\sum_{j=1}^{N} \exp\left(-\frac{1}{2}\text{AIC}_j\right)} \tag{9}$$

Now, substituting the definition of AIC_i from equation 4 and under consideration that all models in `compass` under comparison have the same number of parameters, i.e., $k_i = k_j = k$ for all i, j, then the additional parameter term is common to the numerator and all terms in the sum in the denominator, and thus cancels out. In this specific scenario, the posterior model probability simplifies to a direct ratio of the maximized likelihoods of the data given each model:

$$\mathcal{P}(\mathcal{M}_i|\mathbf{x}) = \frac{\mathcal{L}(\mathbf{x}|\hat{\theta}_i, \mathcal{M}_i)}{\sum_{j=1}^{N} \mathcal{L}(\mathbf{x}|\hat{\theta}_j, \mathcal{M}_j)} \quad (\text{if } k_i = k_j \text{ for all } i, j) \tag{10}$$

The model with the highest posterior probability $\mathcal{P}(\mathcal{M}_i|\mathbf{x})$ is then considered the best-supported model by the data, according to this criterion.

B Network Architecture & Calibration

B.1 Conditional Transformer Architecture

`Compass` employs a time-dependent transformer — `ConditionTransformer` — to approximate the conditional score

$$s_\phi(\mathbf{z}_t, \mathcal{M}_C, t) \approx \nabla_{\mathbf{z}_t} \log p_t(\mathbf{z}_t), \qquad \mathbf{z} = (\theta, x). \quad (11)$$

The network is built on the DiT-style transformer backbone (Peebles & Xie, 2023) and includes two explicit conditioning mechanisms. The timestep conditioning via adaLN, where the scalar diffusion timestep t is embedded (Gaussian-Fourier embedding) and injected through adaptive layer normalization (scale & shift) parameters, which are initialized at zero to stabilize early training. And the structural conditioning via attention masking, where a binary condition mask $\mathcal{M}_C \in \{0,1\}^D$ marks observed dimensions (1) and latent dimensions (0). From $\mathcal{M}_C$ we build a custom attention mask for every self-attention layer so that latent tokens may only attend to observed tokens (preventing direct latent–latent leakage). By switching $\mathcal{M}_C$ at inference we obtain two modes:

- **NPE mode:** $\mathcal{M}_C = (0, 1)$
  Infer θ conditioned on observed x.
- **NLE mode:** $\mathcal{M}_C = (1, 0)$
  Generate x conditional on θ.

The input is processed by projecting each scalar/value in $\mathbf{z}$ to a high-dimensional embedding space using a MLP. These embeddings, along with the condition mask $\mathcal{M}_C$, are processed by a stack of N transformer blocks, each with Multi-Head Self-Attention, a MLP, residual connections and adaLN, as illustrated in Figure 7. The attention heads use the custom mask derived from $\mathcal{M}_C$ so that the allowed attention pattern is enforced at every layer. The final token embeddings are projected to a vector-valued score of the same dimension as $\mathbf{z}_t$, producing $s_\phi(\mathbf{z}_t, \mathcal{M}_C, t)$. The network is trained with the usual denoising/score matching objective appropriate for the chosen SDE.

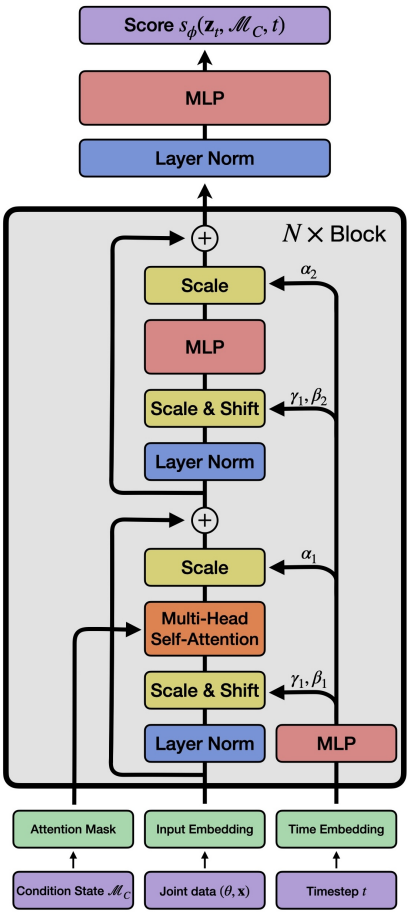

Figure 7: ConditionTransformer: a time-dependent, conditional transformer that predicts the diffusion score. Inputs are tokenized joint vectors (θ, x); conditioning is implemented via (i) adaptive layer normalization (adaLN) with timestep embeddings and (ii) a condition mask $\mathcal{M}_C$ that constructs a custom self-attention mask controlling information flow between tokens.

The dimension of the embedding space, the expansion rate of the multi-layer perceptron (MLP), number of heads in the self-attention layer and the depth of the transformer (N) are adjustable hyperparameters of the network.

B.2 Hyperparameter Selection

The performance of the score-based inference model is highly sensitive to the architectural choices that define its inductive bias and capacity. To identify a robust and high-performing configuration, we conducted a comprehensive hyperparameter optimization study using the `optuna` framework (Akiba et al., 2019), for all three examples from Section 4. The optimization trials were targeting the predictive accuracy and the posterior calibration. The accuracy is measured by the negative log-posterior ($-\log P(\theta|x)$), evaluated at the ground-truth parameters for a set of mock observations. Lower values indicate that the true parameters fall in regions of higher posterior probability, signifying a more accurate model. The calibration of the posterior is measured by the maximum deviation of the TARP diagnostic ($\Delta_{\max}$TARP) from the ideal diagonal line (Lemos et al., 2023). Lower

values indicate better statistical coverage and more reliable uncertainty estimates. The key architectural hyperparameters explored in this study were the number of transformer blocks (`depth`), the dimensionality of the token embeddings (`hidden size`), the expansion factor for the hidden layers in the MLP blocks (`mlp ratio`), the number of attention heads in the Multi-Head Self-Attention mechanism (`num heads`), the batch size (`batch size`) and the maximum noise level in the VESDE schedule (`sigma`).

Table 1 summarizes the top-performing configurations for each of the two objectives on the GCE example from Section 4.3. Trial 275 emerged as a uniquely strong candidate, as it was the only hyperparameter combination to appear in the top-10 list for both predictive accuracy and posterior calibration. This configuration strikes an excellent balance between generating sharp, accurate posteriors and ensuring those posteriors are statistically well-calibrated. We therefore selected the architecture from Trial 275 as the default for the GCE experiments.

Trial	$-\log P(\theta\|x)$	$\Delta_{\max}$TARP	Batch Size	Sigma	Depth	Heads	Hidden Size	MLP Ratio
113	-1.76	0.08	862	2.5	5	1	36	3
174	-1.60	0.09	862	2.5	5	1	36	3
215	-1.90	0.10	862	2.5	11	1	65	3
223	-1.97	0.10	375	2.5	5	1	36	1
260	-1.94	0.10	125	2.5	5	1	34	1
242	-1.76	0.10	842	2.5	5	1	28	3
249	-1.87	0.10	862	2.5	5	1	34	3
275	-2.20	0.10	125	2.5	5	1	65	3
287	-1.92	0.10	608	2.5	5	1	65	3
277	-1.74	0.11	842	2.5	6	1	28	3
193	-2.26	0.15	125	22.2	11	1	65	3
233	-2.22	0.16	125	23.0	3	1	65	10
213	-2.22	0.12	485	2.5	5	1	167	1
275	-2.20	0.10	125	2.5	5	1	65	3
272	-2.17	0.12	485	2.5	3	1	167	2
263	-2.16	0.16	203	27.0	11	1	34	3
256	-2.15	0.11	125	2.5	7	1	65	1
290	-2.15	0.16	161	23.0	4	1	65	9
284	-2.14	0.13	125	3.8	3	1	106	10
195	-2.10	0.16	203	20.1	11	1	27	8

Table 1: Top-performing hyperparameter configurations from a 1000-trial `optuna` study on the GCE task. The study targeted minimizing both the negative predictive accuracy ($-\log P(\theta|x)$) and the maximal deviation of the TARP diagnostic ($\Delta_{\max}$TARP). The upper section lists the top 10 trials ranked by the $\Delta_{\max}$TARP objective, while the lower section lists the top 10 trials ranked by the $-\log P(\theta|x)$ objective. Trial 275 is highlighted as it is the only configuration to appear in both lists and was selected as the default architecture.

A crucial insight from the `optuna` study was the overwhelming importance of the diffusion noise schedule parameter, `sigma` (σ). The choice of σ defines the variance of the terminal noise distribution, $p_T(z) = \mathcal{N}(0, \sigma^2)$, into which all data points are perturbed during the forward process. For the reverse generative process to be effective, this terminal distribution must fully contain the support of the initial data distribution $p_0(z)$. If the initial data contains values that fall outside the typical range of the noise distribution (i.e., if the data variance is larger than the noise variance), the model will fail to generate them accurately. `optuna`'s internal feature importance analysis confirmed this theoretical sensitivity, identifying `sigma` as the most influential hyperparameter. This highlights that ensuring the validity of the diffusion process's core assumption is more critical to performance than the specific architectural details of the denoiser network.

B.3 Diffusion Time

Inference quality depends on the reverse SDE discretization. We evaluated Euler-Maruyama and 1st–3rd order DPM-Solvers (Lu et al., 2022) over 15 timestep schedules, measuring $-\log P(\theta|x)$ vs. runtime on 1000 test samples (see Fig. 8) on the GCE example (4.3). The 1st-order DPM-Solver offers the best accuracy-efficiency trade-off, achieving reliable posteriors ($-\log P(\theta|x) < 0.693$)

with 500 steps and 1s/sample inference time on 8 RTX 2080 Ti GPUs. This configuration is therefore adopted as default.

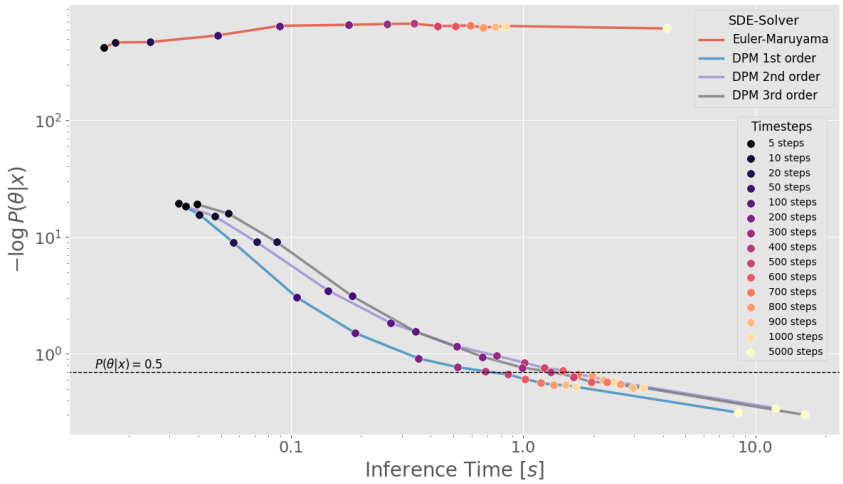

Figure 8: Accuracy of SDE-Solvers comparison of SDE solver performance in terms of predictive accuracy ($-\log P(\theta|x)$) versus inference time per sample. Euler-Maruyama and DPM-Solvers (1st, 2nd, and 3rd order) were tested with varying numbers of diffusion steps (indicated by colored points, ranging from 5 to 5000 steps). Accuracy is averaged over 1000 mock observations. The dashed horizontal line at $-\log P(\theta|x) \approx 0.693$ represents a posterior probability of $P(\theta|x) = 0.5$ for the true parameter. Lower $-\log P(\theta|x)$ values indicate higher accuracy. The DPM-Solver (1st order) with 500 steps offers a good balance of accuracy and computational efficiency.

B.4 Posterior Calibration

To ensure reliable uncertainty quantification, we evaluate posterior calibration using the TARP diagnostic (Lemos et al., 2023) and true-vs-predicted plots. Figure 9 presents calibration results for the six free parameters in the `chempy` simulator from the GCE application (Section 4.3). TARP plots (top row) show credibility intervals match empirical coverage well—curves align with the diagonal, confirming well-calibrated uncertainty estimates. Posterior mean vs. true parameter plots (bottom row) show strong agreement for global parameters (α_{IMF}, $\log_{10}(N_{Ia})$), which benefit from strong data constraints. Local parameters ($\log_{10}(\text{SFE})$, $\log_{10}(\text{SFR}_{\text{peak}})$, x_{out}, and T) show greater spread, reflecting increased uncertainty and weaker constraints—expected given their spatial and temporal variability. On the right of figure 9 is the aggregated TARP across all parameters and test samples. The curve closely tracks $y = x$, confirming that the overall posterior coverage is statistically sound. For example, 90% intervals contain the true values approximately 90% of the time.

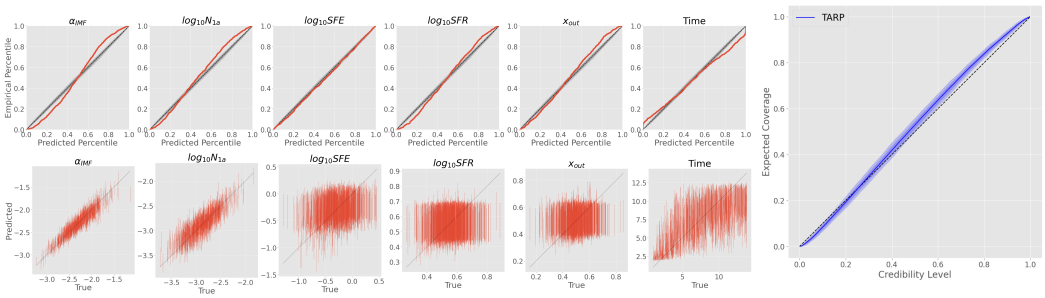

Figure 9: Posterior calibration diagnostics showing the TARP plots at the top and the true vs. predicted parameter plots for each of the six parameters at the bottom and the TARP over all parameters on the right

In summary, the selected architecture and solver yield accurate, calibrated posteriors—especially for global parameters—supporting robust Bayesian inference and model comparison. The broader posteriors for local parameters reflect inherent data limitations.

C Training & Sampling

Training The `ScoreBasedInferenceModel` (`SBIm`) $s_\phi(\mathbf{z}_t, \mathcal{M}_C, t)$ is trained by minimizing a weighted sum of denoising score matching objectives. Specifically, the objective is to learn the score of the data distribution conditioned on the observed values specified by $\mathcal{M}_C$. During the training process, the condition mask $\mathcal{M}_C$ is sampled randomly at every training batch from a Bernoulli distribution, as proposed by Gloeckler et al. (2024), to learn the correlation between all tokens.

`compass` is implemented in PyTorch and designed for efficient training on multiple GPUs. The `AdamWScheduleFree` optimizer from the `schedulefree` package (Defazio et al., 2024) is employed, which replaces the momentum of an underlying optimizer (AdamW) with a combination of interpolation and averaging. This approach eliminates the need for manual learning rate schedule tuning while maintaining state-of-the-art performance. The training routine of a batch is detailed in Algorithm 1.

Sampling Once the score model `SBIm` s_ϕ is trained, samples can be generated from the target conditional distribution $p_0(\mathbf{z}_{\text{latent}}|\mathbf{z}_{\text{observed}})$ by numerically solving the reverse SDE.

`compass` primarily utilizes the DPM-Solver proposed in Lu et al. (2022), a family of high-order solvers that generally offers better accuracy and efficiency compared to the simpler Euler-Maruyama method for a given number of function evaluations. The DPM-Solver leverages the semi-linearity of diffusion ODEs and it directly approximates a simplified formulation of the exact solution, which consists of an exponentially weighted integral of the noise prediction model (Lu et al., 2022). The deterministic part of the reverse SDE for the employed VESDE is $-\sigma^{2t} s_\phi(\mathbf{z}, \mathcal{M}_C, t)$. The DPM-Solver integrates this term, often by leveraging the corresponding probability flow ODE (Song et al., 2020). For a time step form t to t' (where $dt = t - t' > 0$) with the condition mask $\mathcal{M}_C$, the first order DPM-Solver update in the `SBIm` is:

$$\mathbf{z}_{t'} = \mathbf{z}_t - (1 - \mathcal{M}_C)\sigma_t s_\phi(\mathbf{z}_t, \mathcal{M}_C, t)dt \tag{12}$$

Higher-order versions (2nd and 3rd), also implemented in `compass`, utilize intermediate evaluations of the score function and weighted averages to achieve greater accuracy per step.

To further enhance sample quality,`compass`'s implementation incorporates optional Langevin corrector steps. Following a predictor step a few corrector steps can be applied:

$$\mathbf{z}_{t'} = \mathbf{z}_{t'} + (1 - \mathcal{M}_C) \cdot \left(\delta_L \sigma_{t'}^2 s_\phi(\mathbf{z}_{t'}, \mathcal{M}_C, t') + \sqrt{2\delta_L \sigma_{t'}^2} \cdot \mathbf{n}\right) \tag{13}$$

where $\mathbf{n} \sim \mathcal{N}(0, \mathbf{I})$, t' is the next timestep after the last predictor step, and δ_L is the signal-to-noise ratio. These corrector steps inject appropriately scaled noise and re-apply a score-based update, helping to refine samples, improve adherence to the data manifold, and prevent potential mode collapse. In`compass`, these Langevin corrector steps are applied periodically (e.g., every 5 main diffusion timesteps by default) and for a fixed number of iterations (e.g., 5 corrector steps) each time they are triggered.

The combination of the DPM-Solver (predictor) with periodic Langevin correction allows for a more accurate and robust evolution of the diffusion process. The ODE solver efficiently traverses the probability space, while the stochastic Langevin refinement helps to explore it more thoroughly. Algorithm 2 details the sampling procedure for generating conditional samples using a first order DPM-Solver coupled with Langevin corrector steps.

Algorithm 1 Training COMPASS

Require: $z_0 = (\theta, x)$

Sample: $\epsilon \sim \mathcal{N}(0, \mathbf{I})$
Sample: $t \sim \text{Uniform}(0, 1)$
Sample: $\mathcal{M}_C \sim \text{Bernoulli}(1/3)$

$\sigma_t \leftarrow \sqrt{\frac{\sigma^{2t}-1}{2 \ln \sigma}}$
$z_t \leftarrow \mathcal{M}_C \cdot z_0 + (1 - \mathcal{M}_C) \cdot \sigma_t \epsilon$
$\hat{s} \leftarrow s_\phi(\mathbf{z}_t, \mathcal{M}_C, t)$
$\mathcal{L}(\phi) \leftarrow \sigma_t^2 \|(1 - \mathcal{M}_C) \cdot (\epsilon - \sigma_t \hat{s})\|^2$

Update: $\phi \leftarrow \phi - \eta \nabla_\phi \mathcal{L}_{avg}$

Algorithm 2 Sampling COMPASS

Require: $z_{0;\theta} = \theta$ **or** $z_{0;x} = x$
if $z_{0;\theta}$ **then** $\mathcal{M}_C \leftarrow (\{1\}^{D_\theta}, \{0\}^{D_x})$
if $z_{0;x}$ **then** $\mathcal{M}_C \leftarrow (\{0\}^{D_\theta}, \{1\}^{D_x})$

Sample: $\epsilon \sim \mathcal{N}(0, \mathbf{I})$
$z_1 \leftarrow (1 - \mathcal{M}_C) \cdot \epsilon + \mathcal{M}_C \cdot z_{0;\{\theta \text{ or } x\}}$

for $t = 1$ **to** 0
$z_{t'} \leftarrow z_t + (1 - \mathcal{M}_C) \sigma_t s_\phi(\mathbf{z}_t, \mathcal{M}_C, t) dt$
if Corrector Step **then**
$n \sim \mathcal{N}(0, \mathbf{I})$
$z_{t'} \leftarrow z_{t'} + (1 - \mathcal{M}_C) \cdot (\delta_L \sigma_{t'}^2 s_\phi + \sqrt{2\delta_L \sigma_{t'}^2} \cdot n)$

D EXPERIMENTS

D.1 MISSPECIFIED GAUSSIANS

This experiment was designed to validate the core logic of the `compass` BMC pipeline in a controlled setting. The goal was to test its ability to correctly identify a known data-generating process, distinguish it from a subtly incorrect but plausible alternative, and reject a null hypothesis with no explanatory power.

D.1.1 GENERATIVE MODELS

We defined three competing hypotheses, each mapping a single latent parameter θ to a two-dimensional observation $\mathbf{x} = (x_2, x_2)$. For all models, the latent parameter θ was drawn from a standard normal prior distribution, $\theta \sim \mathcal{N}(0, 3^2)$. The specific generative processes for each hypothesis are detailed in Table 2. Hypothesis 1 represents the ground truth. Hypothesis 2 is a phase-shifted version, representing model misspecification. Hypothesis 3 is a null model where the observation is independent of the parameter θ. The data distributions for samples drawn from these three models are visualized in Figure 10.

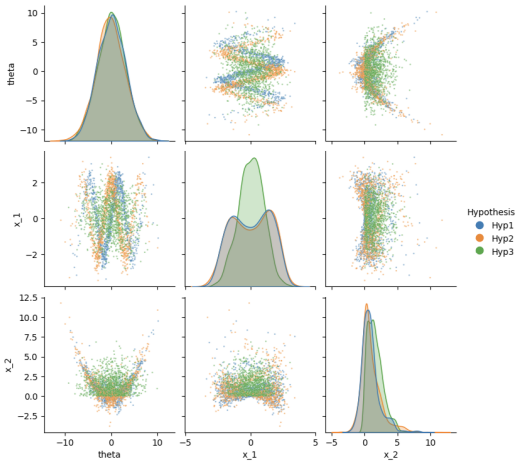

Figure 10: Data distributions for the misspecified Gaussians toy problem. The plots show pairwise relationships between the latent parameter θ and the two observational dimensions (x_1, x_2). The distinct structures generated by the sinusoidal (Hypothesis 1, blue), co-sinusoidal (Hypothesis 2, orange), and null (Hypothesis 3, green) models are clearly visible.

D.1.2 TRAINING & INFERENCE SETUP

For each of the three hypotheses, a dedicated `compass` instance was trained. The training dataset for each model consisted of 10^5 simulated data pairs $(\theta, \mathbf{x})$. The models were trained using the best performing architecture from an `optuna` study (as detailed in B.2) with 5 attention heads, a hidden size of 20, a MLP expansion ratio of 4, 4 transformer blocks, batch size of 256 and a noise level of $\sigma = 3$.

For the model comparison task, a test set of 100 mock observations were generated from the ground-truth model (Hypothesis 1). The BMC pipeline was then run on each of these 100 observations, to produce the results shown in Figure 2 of the main part. Inference was performed using the first order DPM-Solver with 500 diffusion steps.

Hypothesis	x_1 Distribution	x_2 Distribution	Description
Hypothesis 1	$\mathcal{N}(2 \cdot \sin(\theta), 0.5^2)$	$\mathcal{N}(0.1 \cdot \theta^2, (0.5 \cdot \lvert x_1 \rvert)^2)$	Ground Truth
Hypothesis 2	$\mathcal{N}(2 \cdot \cos(\theta), 0.5^2)$	$\mathcal{N}(0.1 \cdot \theta^2, (0.5 \cdot \lvert x_1 \rvert)^2)$	Misspecified Model
Hypothesis 3	$\mathcal{N}(0, 1^2)$	$\lvert \mathcal{N}(0, 2^2) \rvert$	Null Model

Table 2: Generative models for the misspecified Gaussians toy problem. $\mathcal{N}(\mu, \sigma^2)$ denotes a normal distribution with mean μ and variance σ^2.

D.2 Population Dynamics

This experiment demonstrates the application of `compass` to dynamical systems described by ordinary differential equations (ODEs) (Lueckmann et al., 2021). The goal is to choose between competing ecological models of predator-prey dynamics based on time-series data, a common challenge in many scientific fields.

D.2.1 Generative Models

We consider four competing models, each describing the interaction between a prey population $N(t)$ and a predator population $P(t)$. Each model is defined by four free parameters, $\theta = \{\alpha, \beta, \gamma, \delta\}$, the prey growth rate α, the predation rate β and the predator death rate γ. The physical interpretation of the δ parameter and the use of fixed constants differ between the models, representing distinct physical assumptions. The specific ODEs for each model are listed below.

1. **Lotka-Volterra:** The classic model with exponential prey growth and linear predator response.
$$dN/dt = \alpha N - \beta NP \tag{14}$$
$$dP/dt = \delta NP - \gamma P \tag{15}$$

2. **Logistic Prey:** The prey population exhibits logistic growth with a carrying capacity δ', preventing unbounded growth and a fixed conversion efficiency $\epsilon = 0.5$. The carrying capacity is defined as $\delta' = 1000 * \delta$.
$$dN/dt = \alpha N(1 - N/\delta') - \beta NP \tag{16}$$
$$dP/dt = \epsilon \beta NP - \gamma P \tag{17}$$

3. **Satiated Predator:** The predator's consumption follows a Holling Type II functional response, where the predation rate per prey saturates as the prey population N increases. The conversion efficiency is fixed to $\epsilon = 0.5$.
$$\text{consumption} = \frac{\beta N}{1 + \beta \delta N} \tag{18}$$
$$dN/dt = \alpha N - \text{consumption} * \text{P} \tag{19}$$
$$dP/dt = \text{consumption} * \epsilon\text{P} - \gamma\text{P} \tag{20}$$

4. **Rosenzweig-MacArthur:** A combination of the Logistic Prey and Satiated Predator models, incorporating a prey carrying capacity ($\delta' = 1000 * \delta$), a fixed conversion efficiency $\epsilon = 0.5$ and a fixed handling time ($h = 0.1$).
$$\text{consumption} = \frac{\beta N}{1 + \beta h N} \tag{21}$$
$$dN/dt = \alpha N(1 - N/\delta') - \text{consumption} * \text{P} \tag{22}$$
$$dP/dt = \text{consumption} * \epsilon\text{P} - \gamma\text{P} \tag{23}$$

D.2.2 Simulation & Data Generation

To generate the training and validation datasets, we followed the setup of Lueckmann et al. (2021), sampling the four parameters $\theta = \{\alpha, \beta, \gamma, \delta\}$ from a shared prior distribution. Specifically, the

logarithm of each parameter is drawn from a Normal distribution:

$$\log(\alpha) \sim \text{LogNormal}(-0.125, 0.5) \tag{24}$$
$$\log(\beta) \sim \text{LogNormal}(-3, 0.5) \tag{25}$$
$$\log(\gamma) \sim \text{LogNormal}(-0.125, 0.5) \tag{26}$$
$$\log(\delta) \sim \text{LogNormal}(-3, 0.5) \tag{27}$$

For each parameter set, the system was evolved over a time of $T = 20$ with initial conditions $(N(0), P(0)) = (30, 1)$. The final simulated time-series data x_{sim} consists of the population counts at 20 integer time steps. To replicate observational uncertainties, Gaussian noise was added to the population counts $n \sim \mathcal{N}(0, 0.1^2)$.

D.2.3 Training & Inference Setup

For each of the four population models, a dedicated `compass` instance was trained. The training dataset for each model consisted of $5 * 10^5$ simulated data pairs $(\theta, \mathbf{x})$ and $5 * 10^4$ validation pairs, generated as described in Section D.2.2. The raw population counts were scaled down by a factor of 100 before being passed to the network. This normalization step is crucial for ensuring the data distribution falls within the support of the initial noise distribution of the diffusion model (see Appendix B.2). The models were trained using the an architecture optimized for this task via an additional `optuna` study targeting this problem (as detailed in B.2). The final configuration uses 5 attention heads, a hidden size of 50, a MLP expansion ratio of 4, 8 transformer blocks, batch size of 1000 and a noise level of $\sigma = 2$.

For the model comparison task shown in 4.2, a single ground-truth time-series was generated using the Lotka-Volterra model with log-parameters $(\log(\alpha), \log(\beta), \log(\gamma), \log(\delta)) = (-0.1, -3.0, -0.1, -3.0)$. The resulting 40-dimensional observation vector (20 time points for prey, 20 for predator) was fed to `compass`. The BMC pipeline was run using the second-order DPM-Solver with 50 diffusion steps. This reduced step count was chosen to demonstrate that `compass` retains sufficient accuracy for decisive model selection even at significantly accelerated inference speeds.

D.3 Galactic Chemical Enrichment

This section provides the detailed experimental setup for the main scientific application of `compass`, performing model comparison and parameter inference in the context of Galactic Chemical Evolution (GCE).

D.3.1 The `chempy` Simulator

All simulations were performed using the one-zone GCE model `chempy` (Rybizki et al., 2017). This model simulates the chemical enrichment of a stellar population over cosmic time. Our inference focuses on a set of six free parameters that govern the simulation's outcome, which can be grouped into global and local parameters. The parameters and their Gaussian prior distributions, used for parameter inference benchmarks and as a reference point for the real data analysis, are detailed in Table 3.

D.3.2 Nucleosynthetic Yield Sets for Model Comparison

A central goal of this work is to determine which set of theoretical nucleosynthetic yields best reproduces observational data. We constructed a suite of 40 competing GCE models, where each model is a unique combination of yield tables for Asymptotic Giant Branch (AGB) stars and core-collapse supernovae (CC-SN). The SN-Ia yields were held fixed to the tables from Seitenzahl et al. (2012). All yield table configurations can be found in Table 4. The results match the inference results in Figure 5.

D.3.3 Observational Data

For the primary scientific analysis, we used the high-precision stellar abundance dataset from Nissen et al. (2020). This dataset contains observations for 69 solar-type stars, providing measurements for

Parameter	Description	$\overline{\theta}_{\text{prior}} \pm \sigma_{\text{prior}}$	Prior from:
	$\vec{\Lambda}$: *Global stellar (SSP) parameters*		
α_{IMF}	High-mass slope of the (Chabrier, 2003) IMF	-2.3 ± 0.3	(Chabrier, 2003, Tab. 1)
$\log_{10}(N_{\text{Ia}})$	Number of SN Ia per $M_\odot$ over 15 Gyr	-2.89 ± 0.3	(Maoz & Mannucci, 2012, Tab.1)
	$\vec{\Theta}_i$: *Local ISM parameters*		
$\log_{10}(\text{SFE})$	Star formation efficiency governing gas infall	-0.3 ± 0.3	(Bigiel et al., 2008)
$\log_{10}(\text{SFR}_{\text{peak}})$	SFR peak in Gyr (scale of $k = 2$ Γ-distribution)	0.55 ± 0.1	(van Dokkum et al., 2013, fig. 4b)
x_{out}	Stellar feedback fraction	0.5 ± 0.1	(Rybizki et al., 2017, Tab. 1)
	T_i: *Timescale*		
T_i	Time of stellar birth in Gyr	[1,13.8]	Observations

Table 3: Free parameters of the `chempy` simulator. The global parameters $(\alpha_{\text{IMF}}, \log_{10}(N_{\text{Ia}}))$ are assumed to be shared across all stars in a given dataset. The local parameters and the stellar birth time T are specific to each individual star's formation environment.

AGB Yields	**Core-Collapse Supernovae (CC-SN) Yields**									
	Chieffi	Nomoto	Portinari	Chieffi Net	Nomoto Net	NuGrid	West	TNG	CL18	Frischknecht
NuGrid	**1**	6	29	4	5	22	19	**0**	30	36
Karakas	**2**	18	31	14	21	26	23	9	35	39
Ventura	10	16	34	13	17	27	20	3	32	38
TNG	7	15	28	11	12	25	24	8	33	37

Table 4: Ranking of nucleosynthetic yield set combinations based on Bayesian Model Comparison with Nissen et al. (2020) data. Lower ranks indicate higher posterior probability and correspond to the numbers in Figure 5.

stellar age and the abundances of carbon (C), iron (Fe), magnesium (Mg), oxygen (O) and silicon (Si).

D.3.4 Training & Inference Setup

For each of the 40 competing nucleosynthetic yield set models, a dedicated `compass` instance was trained. The training dataset for each model consisted of 10^6 simulated data pairs $(\theta, \mathbf{x})$, with an additional 10^5 pairs reserved for validation. To ensure the model learned robustly across a wide parameter space, the training parameters (Λ, Θ_i) were drawn from a uniform prior distribution spanning ± 5 times the Gaussian prior widths (σ_{Prior} from Tab. 3) around their respective prior means (μ_{Prior}). This wide uniform sampling helps mitigate potential biases towards the prior mean during training. All simulated abundances are then perturbed with a 5% observational uncertainty to mimic the statistical uncertainties of real observational data.

The models were trained using the an architecture optimized for this task via an `optuna` study targeting this problem (Trial 275 in Tab. 1). The inference settings for the reverse SDE solver were adapted to the specific task to balance accuracy and computational cost. For the task of ranking the 40 yield set models, the number of diffusion steps was reduced to 50. As demonstrated in Figure 8, this provides sufficient accuracy for model discrimination at a fraction of the computational cost ($\sim$ 0.1 seconds per sample), making large-scale model comparison feasible. For the final parameter inference on the Nissen et al. (2020) dataset using the best-performing models, the number of diffusion steps was increased to 1000. Given the relatively small sample size (69 stars), this maximized the accuracy of the final parameter constraints. For all benchmark tests on mock data shown in Figures 11 and 12, the default configuration of 500 diffusion steps was used to ensure a consistent and fair comparison with the default settings of the baseline methods. All inference was performed using the 1st-order DPM-Solver unless otherwise noted.

D.3.5 Validation on Mock Data

To rigorously validate `compass`'s parameter inference capabilities, three distinct mock datasets were used, each designed to test a different aspect of robustness.

Accuracy in Matched Conditions. First, we tested parameter inference in an idealized setting where the training and test data are generated from the same model (the "matched" TNG yield set). Figure 11 compares `compass` to a strong NPE-based SBI baseline (Buck et al., 2025) and traditional Hamiltonian Monte Carlo (HMC). The results show that `compass` achieves comparable accuracy to the specialized SBI method and correctly recovers the ground-truth parameters. Notably, as an amortized method, its inference time is orders of magnitude faster than HMC, making it scalable to large datasets.

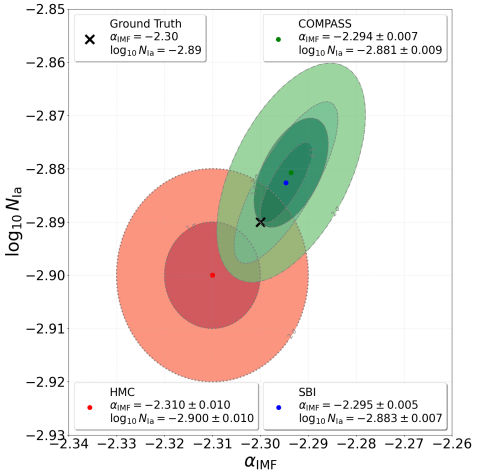

Figure 11: Parameter inference on mock GCE data under matched conditions. The joint posterior (1σ and 2σ contours) for the two global parameters $(\alpha_{\rm IMF}, \log_{10}(N_{Ia}))$ is shown for `compass` (green), an NPE-based SBI baseline (blue), and HMC (red). The ground truth is marked with a black 'X'. `compass` achieves comparable accuracy to the specialized SBI method and is significantly more computationally efficient than HMC.

Robustness to Model Misspecification. A critical test for any inference tool is its performance when the underlying model assumptions are wrong. We test this in two ways. First, we generate data using an "alternative" yield set not used during training. Figure 12 shows on the left that while all methods exhibit a bias (as expected), `compass` remains more robust, with its inferred posterior staying closer to the ground truth than the baselines. Second, we perform a more extreme test by applying our relatively simple `chempy`-trained model to infer parameters from a full, complex cosmological hydrodynamical simulation (Pillepich et al., 2017). As shown on the right in Figure 12, `compass` still provides reasonable parameter estimates, demonstrating remarkable resilience to structural model error.

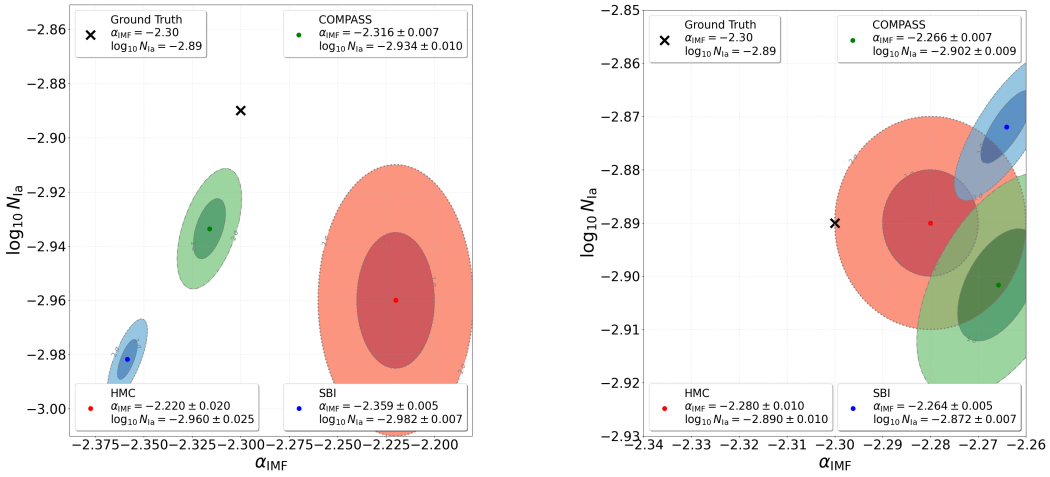

Figure 12: `compass` demonstrates robustness to model misspecification. **Left**: Inference results when the model is trained on one set of simulator physics ("yields") but tested on data generated from another. `compass` (green) remains closer to the ground truth than baselines. **Right**: Inference results when a simple `chempy` model is used to infer parameters from a complex hydrodynamical simulation. `compass` still provides reasonable parameter estimates, highlighting its resilience to structural model error.

D.3.6 Parameter Inference

While the primary focus of the GCE analysis was model selection, we also performed parameter inference using the top-performing yield set models identified in the BMC pipeline. This step

demonstrates the framework's end-to-end utility and provides scientifically relevant constraints on key galactic parameters.

Figure 13 shows the joint posterior distributions for the global parameters α_{IMF} and $\log_{10}(N_{Ia})$, inferred from the Nissen et al. (2020) observational data. Each coloured contour represents the posterior derived using one of the top six yield set combinations from the model comparison analysis (see Figure 5). The posteriors are tightly constrained and consistent across the best models, indicating that the inference is data-driven and robust to minor variations in the underlying model physics. All inferred posteriors are clearly distinct from the prior distribution (marked by 'X'), highlighting the strong constraining power of the data. The final mean values and 1σ uncertainties for each of these models are summarized in Table 5.

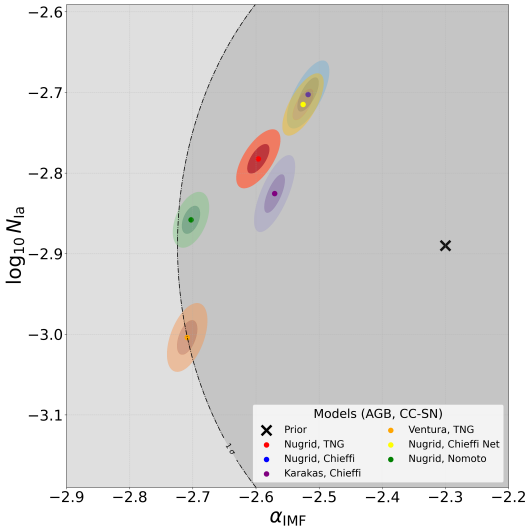

Figure 13: Inferred galactic parameters from observational data.

AGB **CC-SN**	Nugrid TNG	Nugrid Chieffi	Karakas Chieffi	Ventura TNG	Nugrid Chieffi Net	Nugrid Nomoto
α_{IMF}	-2.60 ± 0.02	-2.52 ± 0.02	-2.57 ± 0.01	-2.71 ± 0.01	-2.53 ± 0.02	-2.70 ± 0.01
$\log_{10} N_{\mathrm{Ia}}$	-2.78 ± 0.02	-2.70 ± 0.02	-2.83 ± 0.02	-3.00 ± 0.02	-2.71 ± 0.02	-2.86 ± 0.02

Table 5: Inferred galactic parameters from observational data. Mean values and $\pm 1\sigma$ uncertainties for α_{IMF} and $\log_{10}(N_{\mathrm{Ia}})$ for the top six yield set combinations.

D.3.7 Layer-wise Evolution of the Attention Mechanism

To supplement the layer-averaged attention analysis presented in the main paper (Figure 6), Figure 14 provides a detailed view of how the attention patterns evolve layer-by-layer within the transformer. This visualization offers deeper insight into the model's hierarchical reasoning process. In the early layers (1-2), attention is broadly distributed across multiple observational features. A significant shift occurs in the intermediate layers (e.g., Layer 3), where the model's inference for α_{IMF} begins to sharply focus on key elemental tracers like Oxygen (x_5). In the final layers (4-5), this strategy evolves into a more sophisticated, multi-element pattern. This confirms that the final parameter estimates are derived from learned physical relationships between different abundances, rather than relying on a single tracer, showcasing the interpretability of the model's internal strategy.

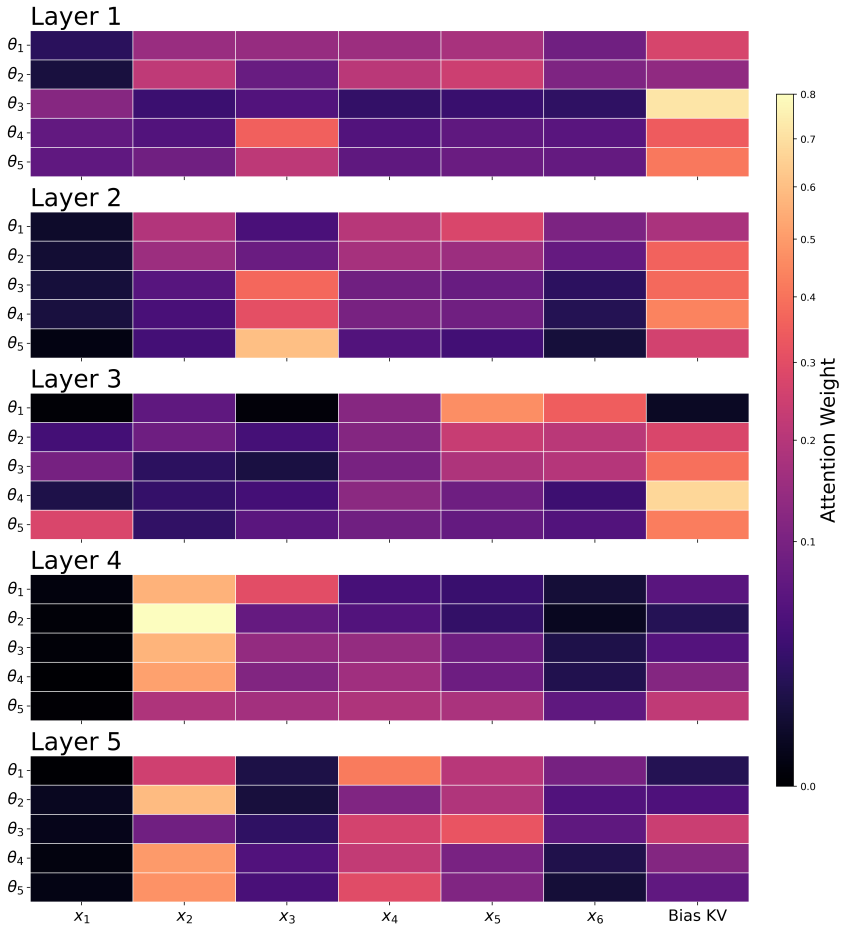

Figure 14: Layer-wise evolution of mean attention in the GCE model. The heatmaps show the mean attention weights for each of the transformer's five layers, averaged over all 69 stellar observations from the Nissen et al. (2020) dataset. The y-axis (queries) represents the parameters being inferred, while the x-axis (keys) represents the observational features and a learned bias term. The plot reveals a hierarchical reasoning process, with the model's focus sharpening from diffuse patterns in early layers to specific elemental tracers in later layers.