Using ARGs[1] to study barriers to gene flow

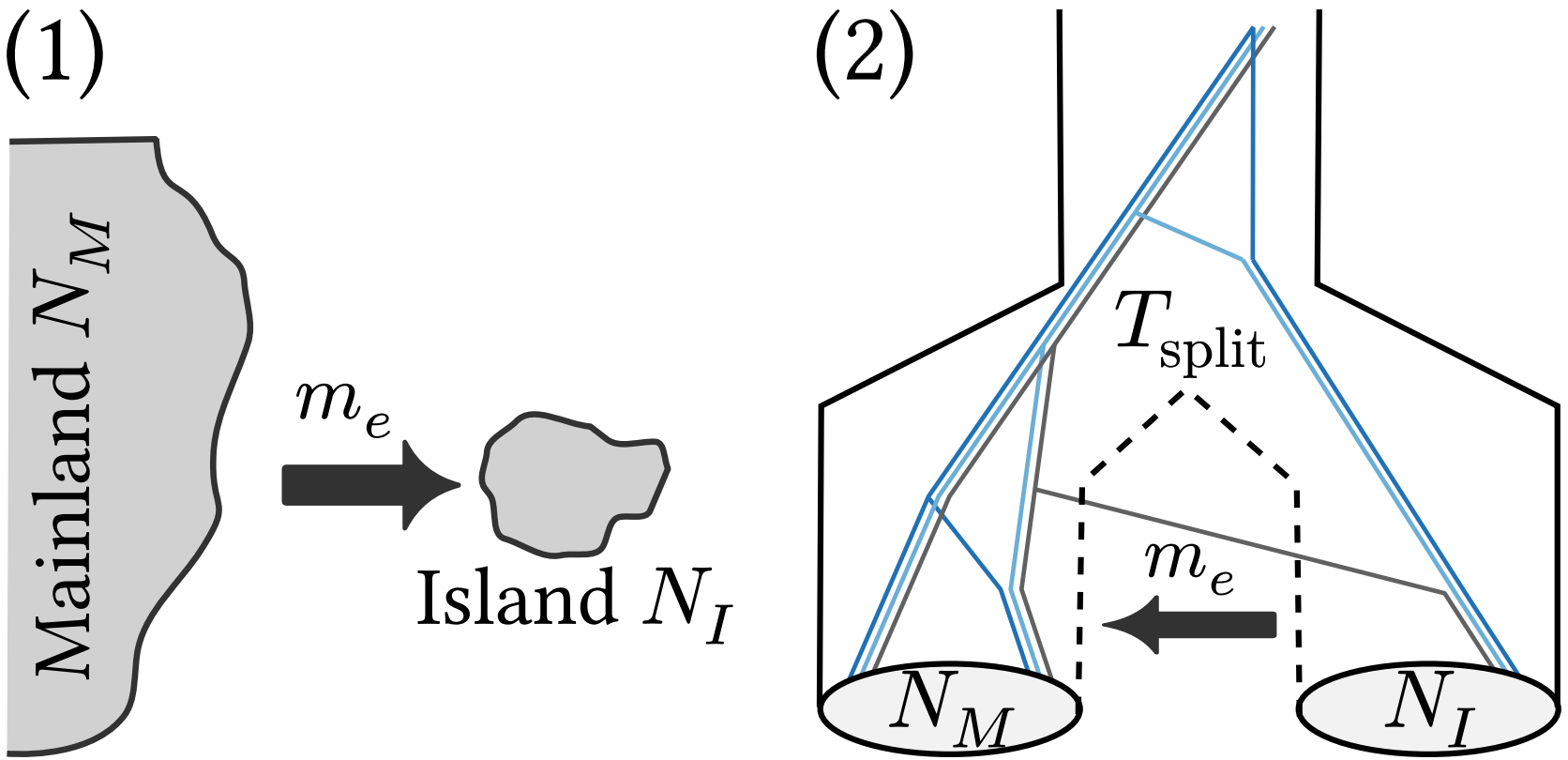
Curro Campuzano Jiménez

University of Antwerp

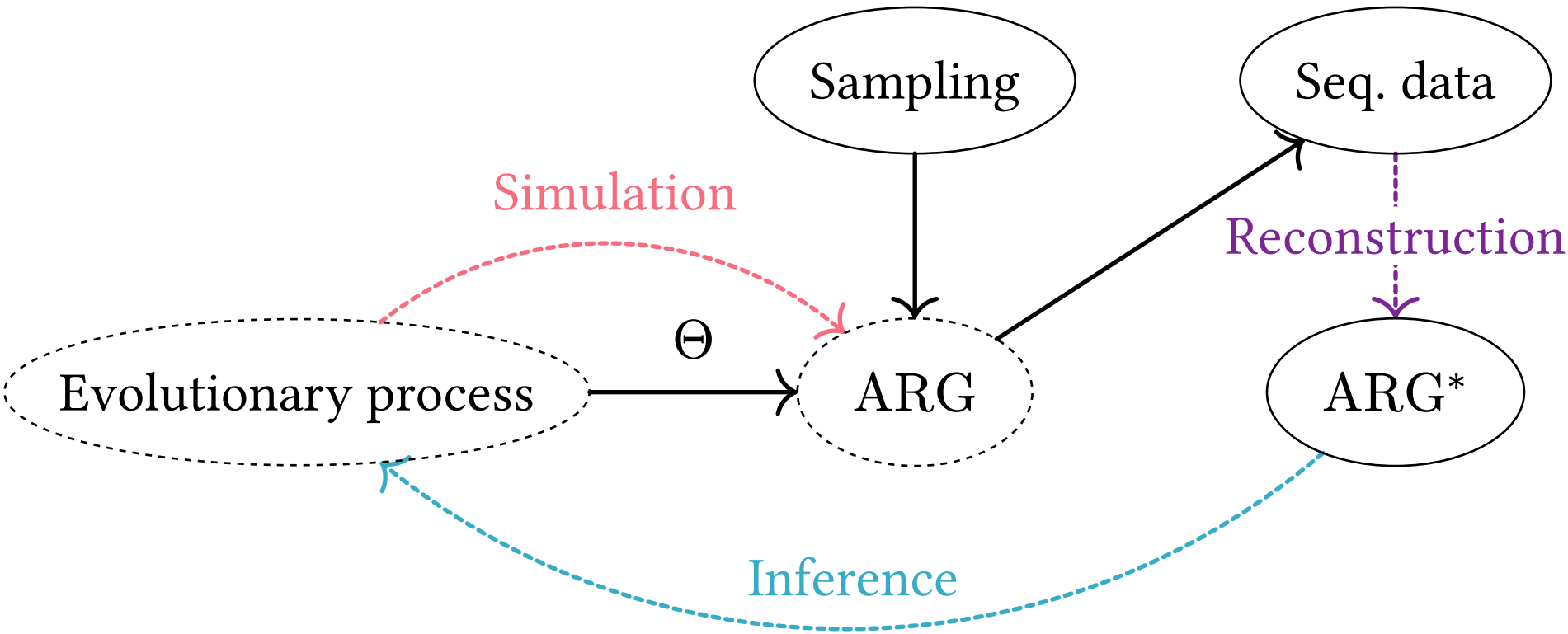
BridgeBarrier meeting

[1] Ancestral recombination graphs

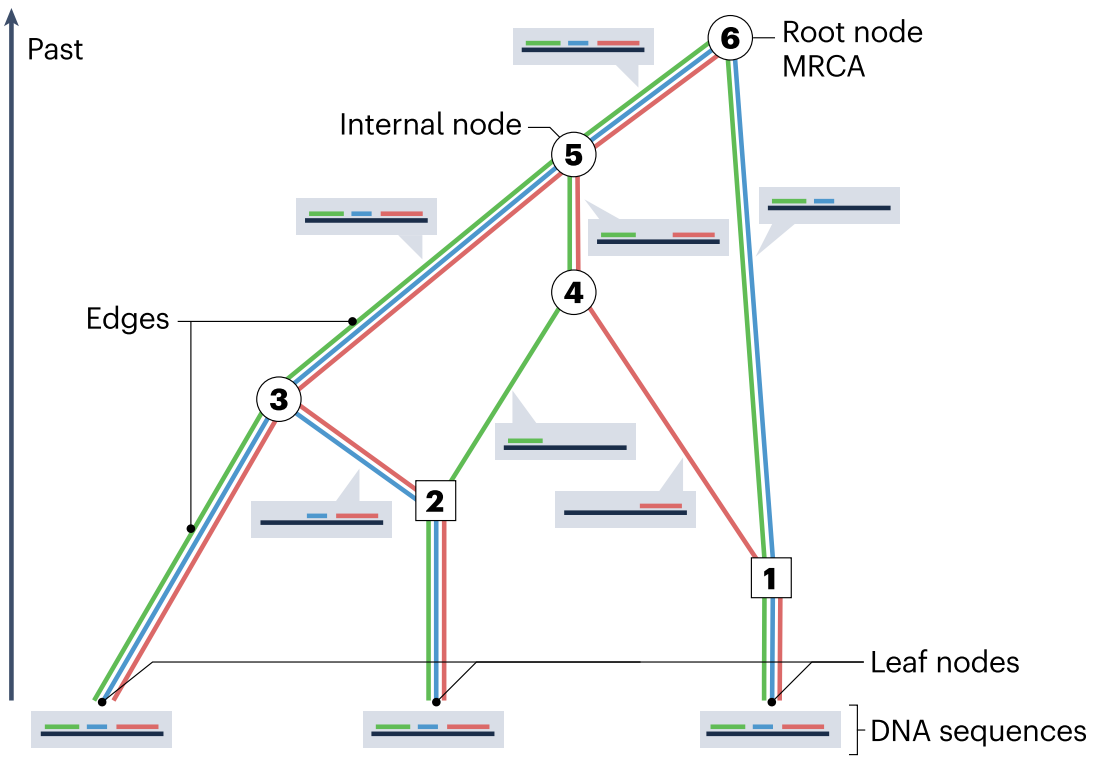
The scenario



Generative model

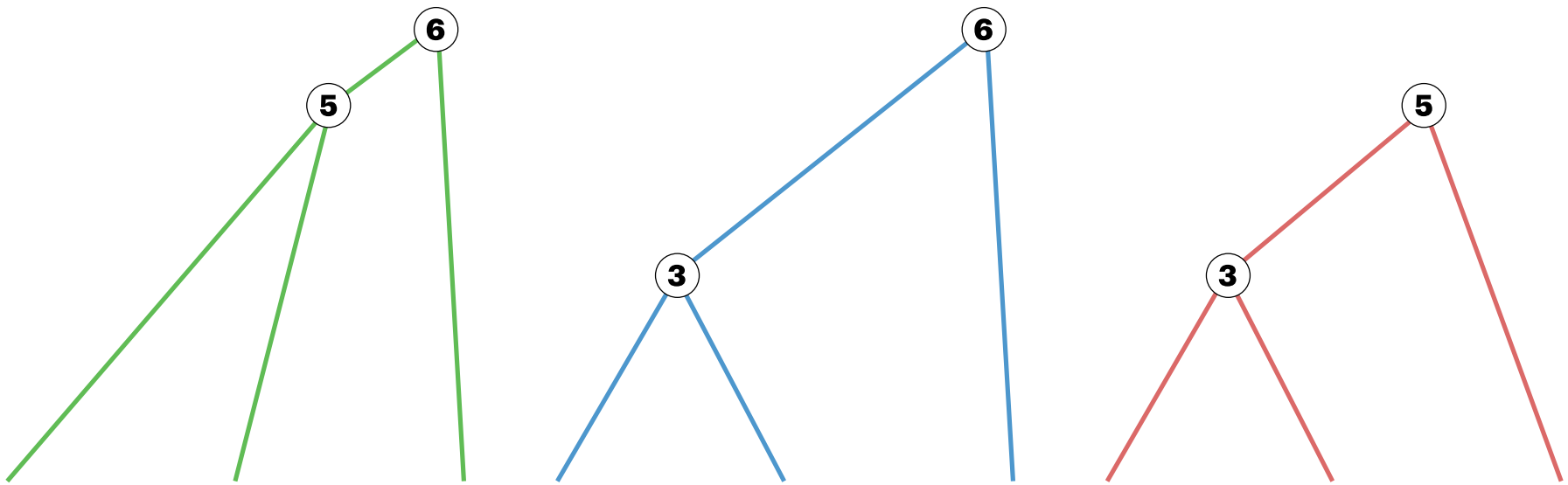


ARGs in a nutshell



Nielsen *et al.*, Nat. Rev. Genet (2024)

ARGs in a nutshell

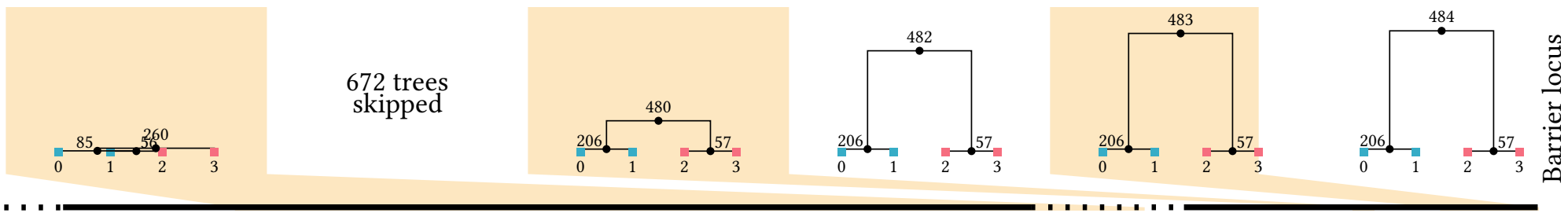


Alternative representation as a sequence of trees

Nielsen *et al.*, Nat. Rev. Genet (2024)

ARGs application

If we look from the left of a barrier locus:



Simulating ARGs

Forward-in-time

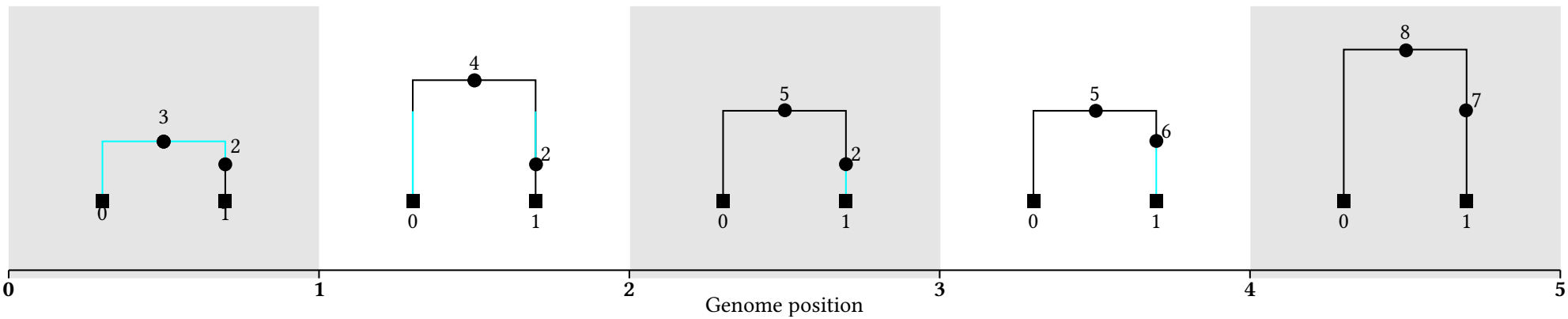
- Complex genetic architectures of reproductive isolation
- Tree-sequence recording

Backwards-in-time

- Effective parameters from single-locus theory
- But what next?

An “extended” SMC’ model

Consider a sequence of trees with (unary) migration nodes



The idea: what if we allow parameters to vary “spatially”?

An SMC' approximation (when $n = 2$)

Initialize with

$$x \leftarrow 0$$

$$T_{\text{mig}}^{(0)} \sim \text{Exp}\left(m_e^{(x)}\right)$$

$$T_{\text{coal}}^{(0)} \sim T_{\text{mig}}^{(0)} + \text{Exp}\left(\lambda_m^{(x)}\right)$$

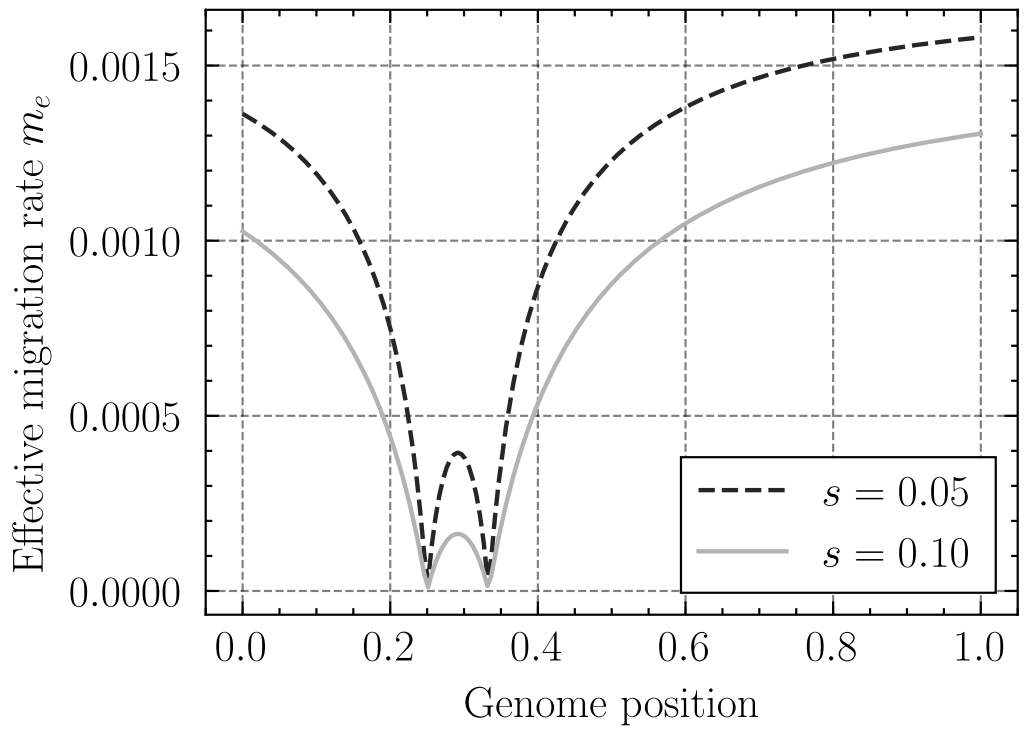
Wait until the next tree,

$$x \leftarrow x + \text{Exp}\left(2T_{\text{coal}}^{(i+1)} \cdot r\right)$$

and draw the next tree

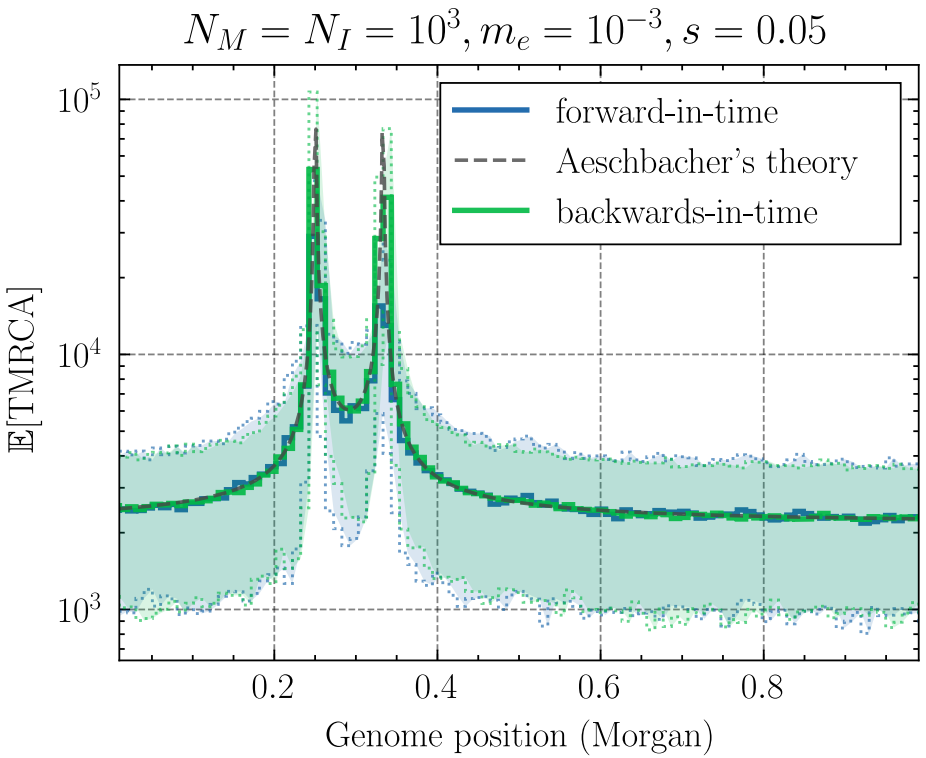
$$\left(T_{\text{mig}}^{(i+1)}, T_{\text{coal}}^{(i+1)}\right) \sim \text{SMC}'\left(T_{\text{mig}}^{(i)}, T_{\text{coal}}^{(i)}, x\right)$$

Plug-in m_e predictions into an extended SMC'



Predicted m_e under Aeschbacher's theory with two barrier loci.

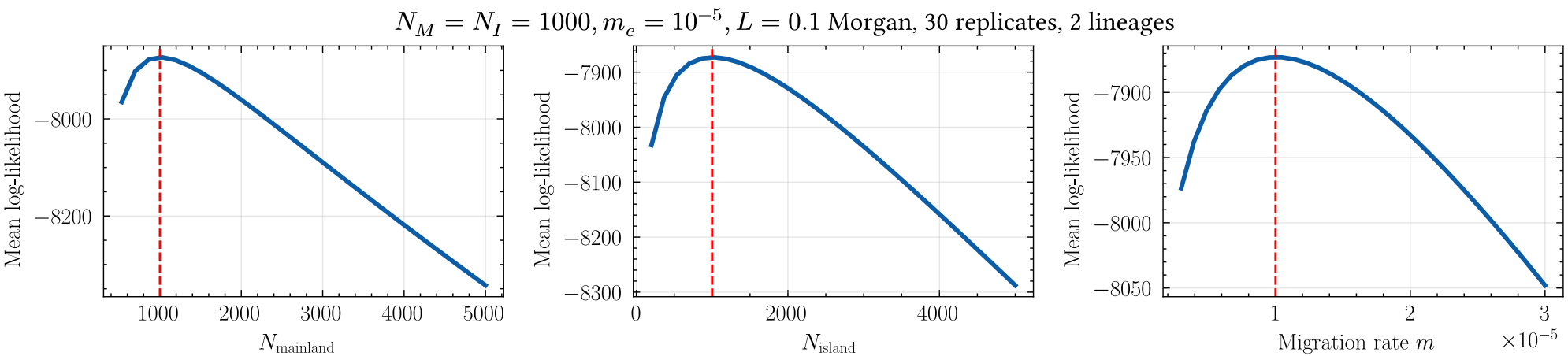
Plug-in m_e predictions into an extended SMC'



Distribution of tree heights.

Likelihood under the extended SMC' for $n = 2$

Right now I'm working on deriving exact likelihoods under the SMC' model and use them to infer genetic architectures.



Problem: I'm finding marginalizing T_{mig} tricky

Inference with ARGs

- Increasing number of inference schemes that treat ARGs as data
- Yet, the focus is on inferring complex (but neutral) demographic processes.

What's the (if any) benefit of using ARGs to study barriers to gene flow?

Guo *et al.*, PLoS Computational Biology (2022)
Pope *et al.*, PNAS (2023)
DeHaas *et al.*, bioRxiv (2025)
Fan *et al.*, Nat Genet (2025)

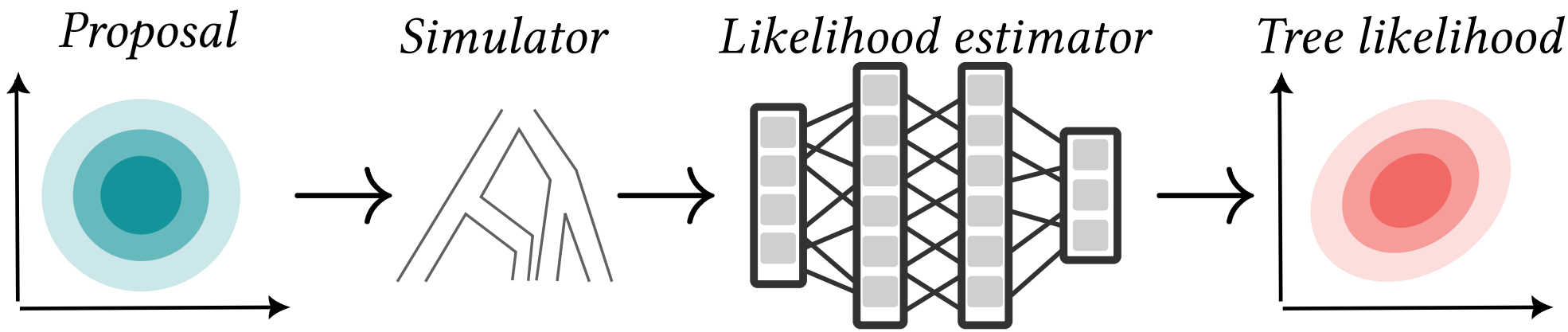
Approximate inference scheme

Idea: replace the complicated joint-likelihood with a power-scaled composite likelihood across marginal trees

Problem: computing exact likelihoods of marginal trees is expensive (in $\mathcal{O}(n)^4$) and we have many trees!

Approximate inference scheme

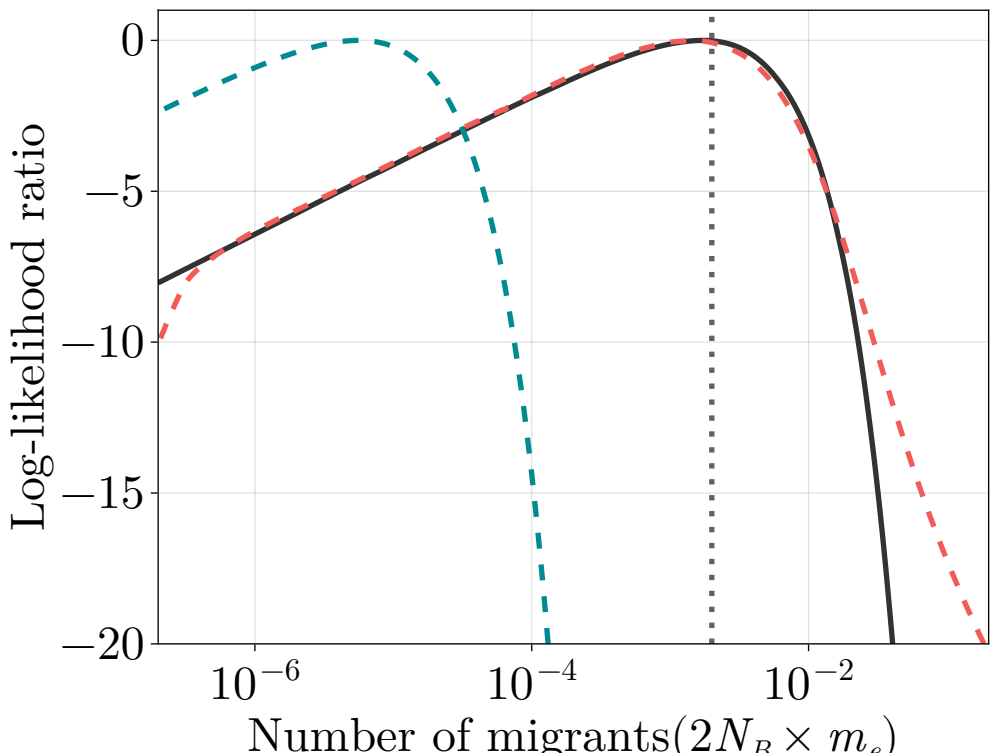
(1) Training amortized tree likelihood estimator $\mathcal{L}_{\text{nn}}(\theta \mid t)$



(2) Approximate inference with a composite likelihood

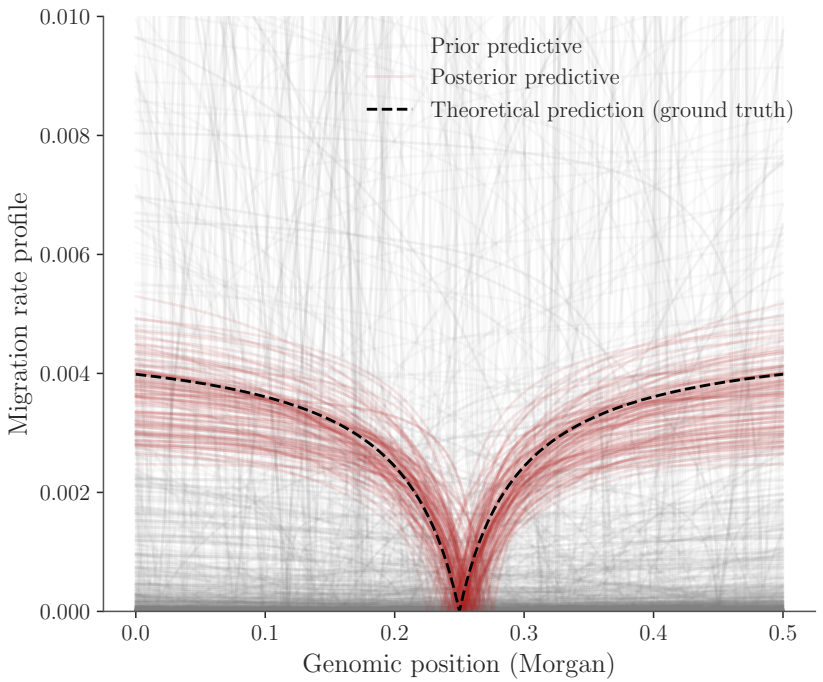
$$\mathcal{L}(\theta|\text{ARG}) \propto \prod_{\text{every } t \text{ tree}} \mathcal{L}(\theta|t)^{\alpha} \cdot \text{span}(t) \approx \mathcal{L}_{\text{nn}}(\theta|t)^{\alpha} \cdot \text{span}(t)$$

The surrogate neural likelihood seems to work!



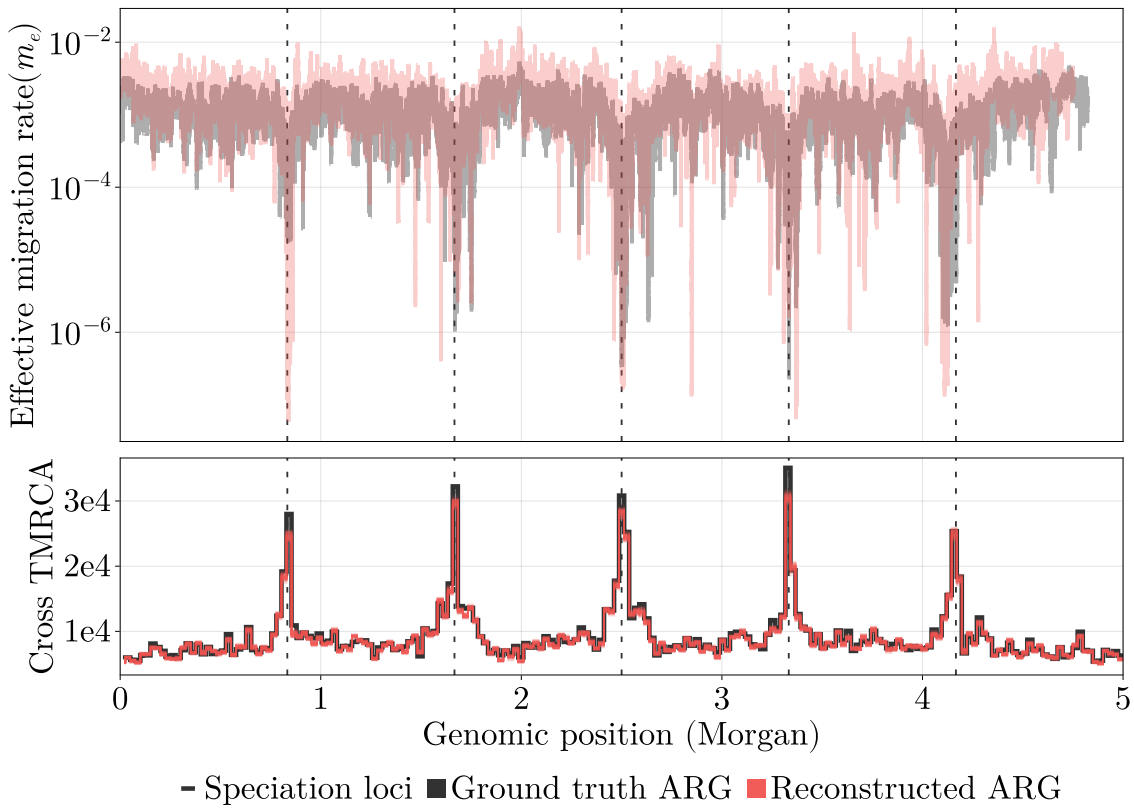
- Ground truth
- ODE solver (exact)
- Neural likelihood (approximation)
- Volz (2012) analytical eq. (approximation)

Putting all together

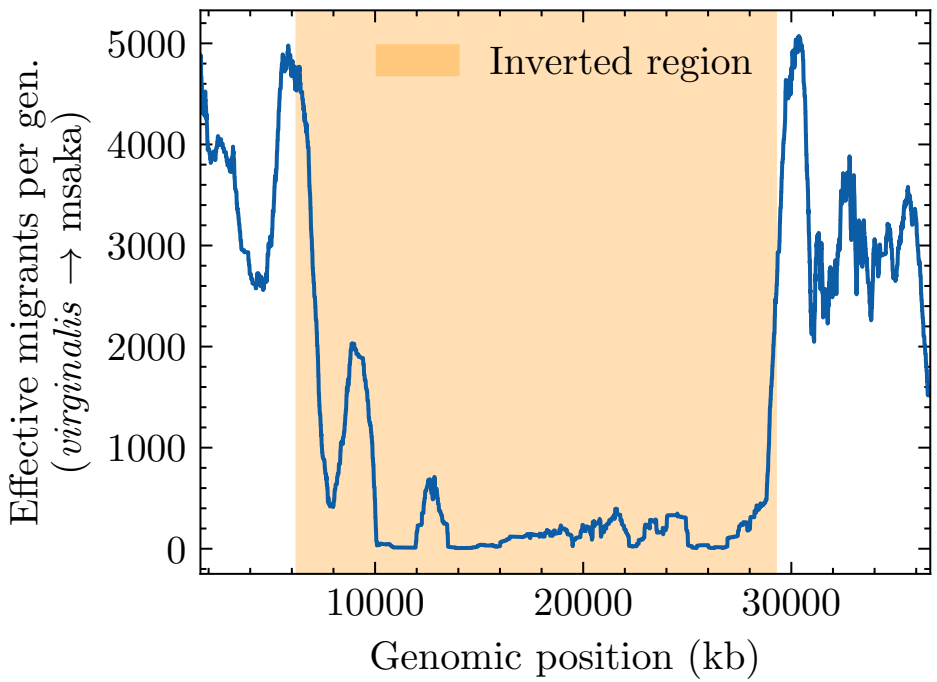


Selection coefficient estimated under weak migration as in Petry (1983)

A la gIMble / *a la* DILS



Preliminary analysis with empirical data



MLE estimate from 40 cichlid chromosomes

Acknowledgements

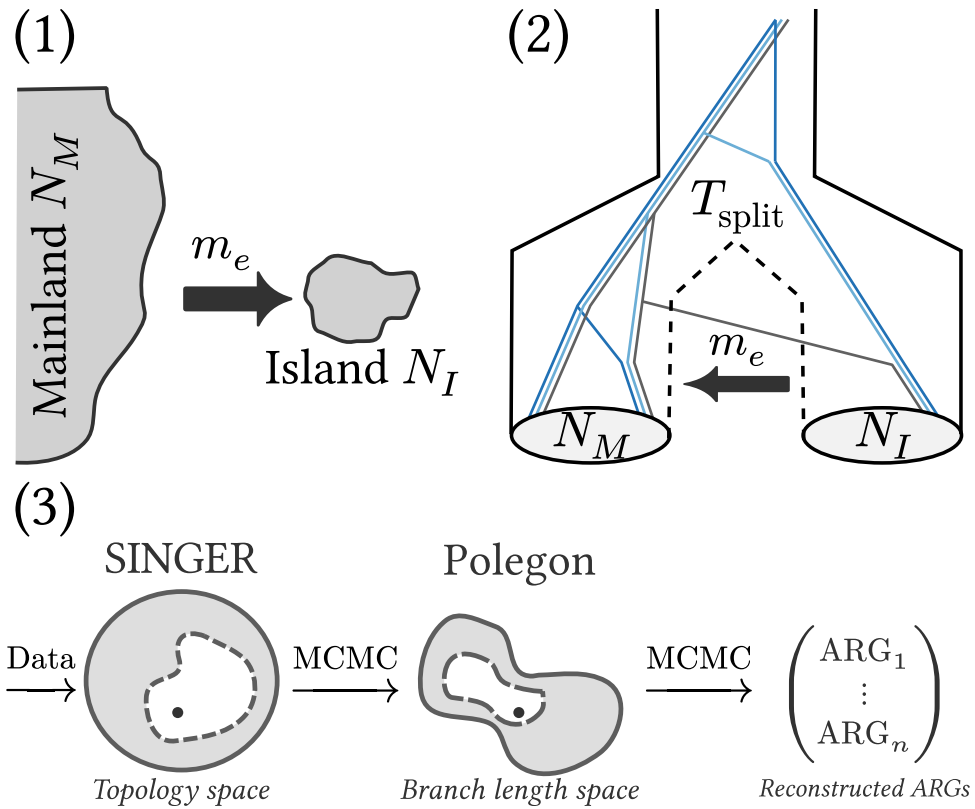
fwo

University of Antwerp

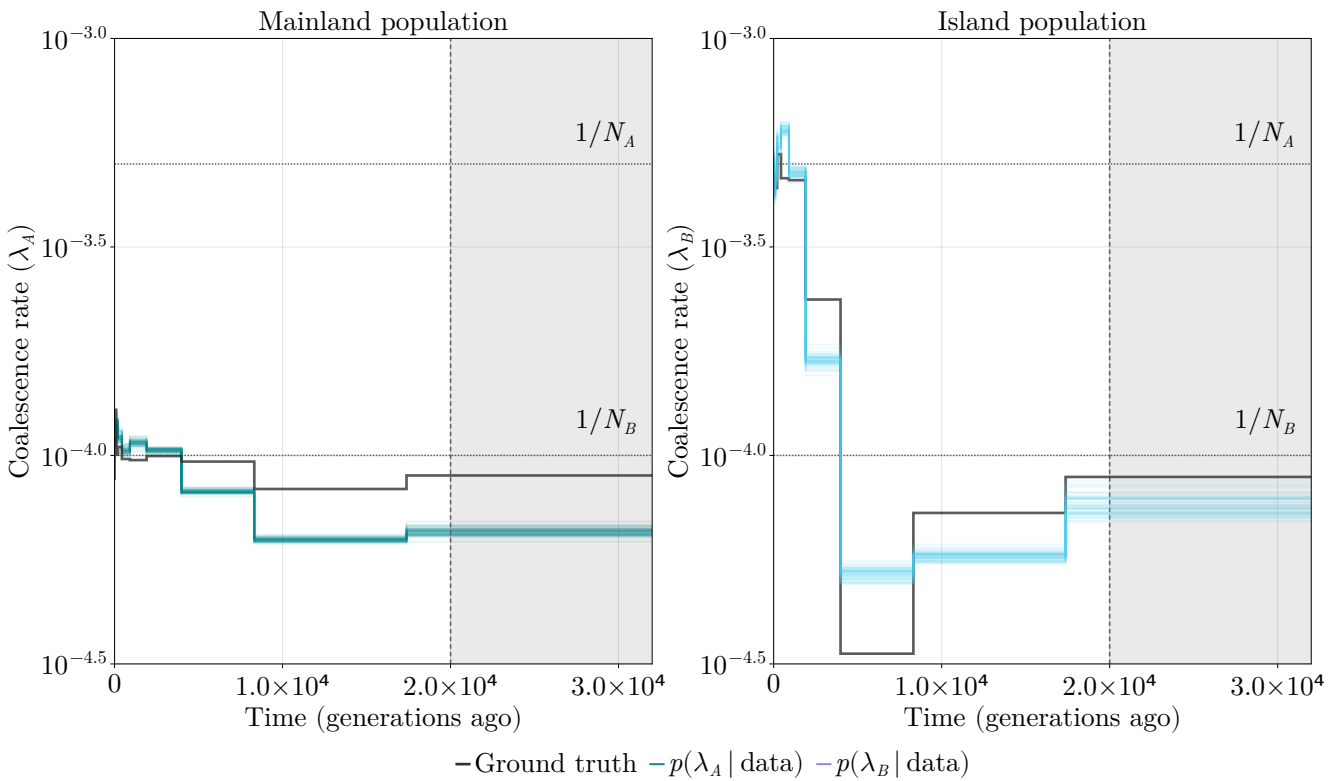
Svardal lab members

Suggestions?

Reconstruction



Reconstruction



Training neural likelihood

Sufficient summary statistic of a marginal tree, $S(\text{tree})$, as a two-dimensional matrix:

$$S(\text{tree}) = \begin{pmatrix} t_1 & e_{1,1} & e_{1,2} & e_{1,3} \\ t_2 & e_{2,1} & e_{2,2} & e_{2,3} \\ \vdots & \vdots & \vdots & \vdots \\ t_{n-1} & e_{n,1} & e_{n,2} & e_{n,3} \end{pmatrix}$$

Training neural likelihood

Simulate a training dataset

$$\theta_i \sim \text{some prior / proposal}$$

$$S(\mathcal{T}_i) \sim \theta_i$$

Training neural likelihood

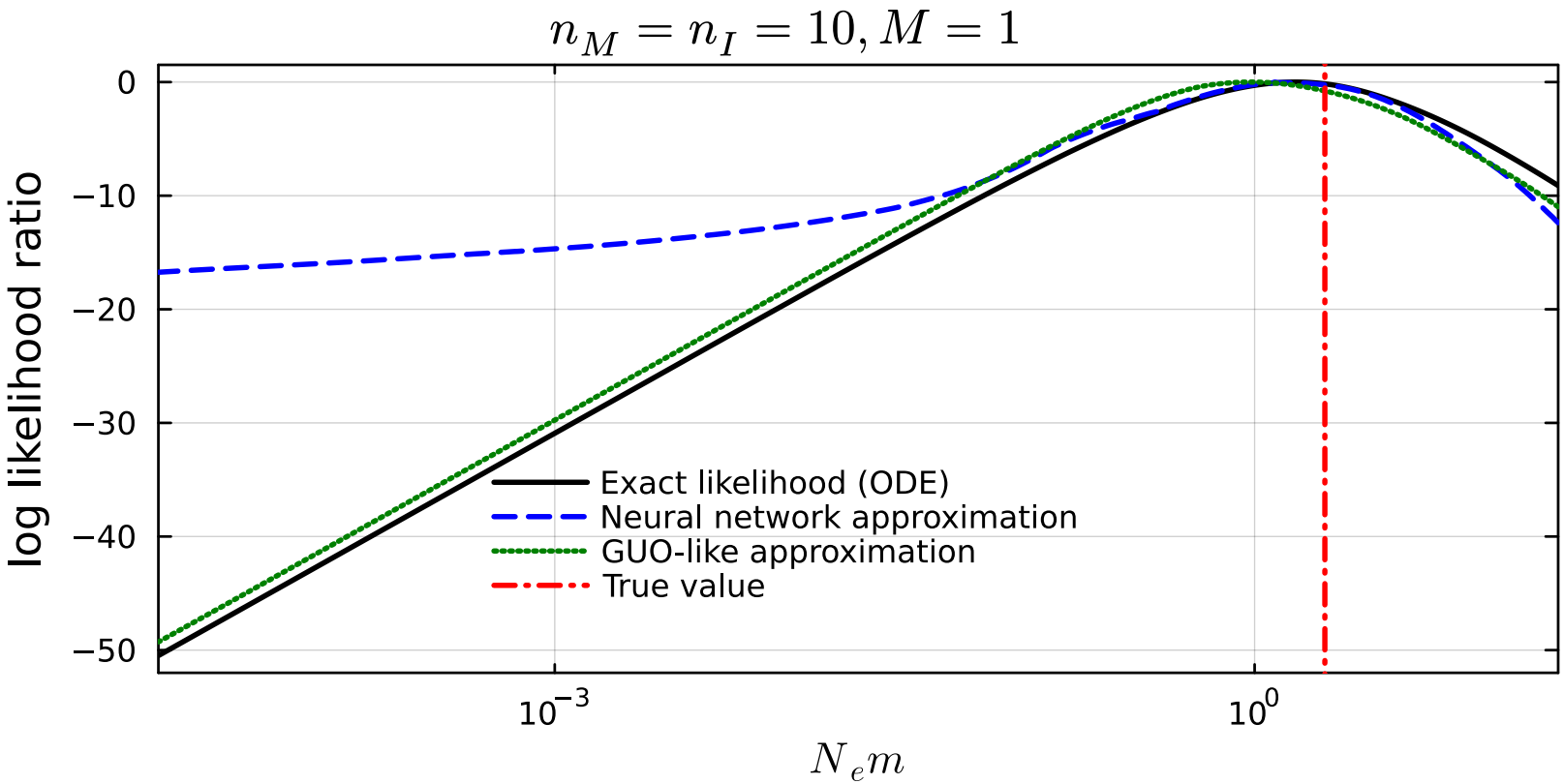
For a neural network φ architecture that emulates the likelihood, $q_{\varphi(\text{tree} \mid \theta)}$, we want to minimize

$$\mathbb{E}_\theta \Big[\mathrm{KL} \Big(\mathcal{L}(\text{tree} \mid \theta) \parallel q_{\varphi(\text{tree} \mid \theta)} \Big) \Big]$$

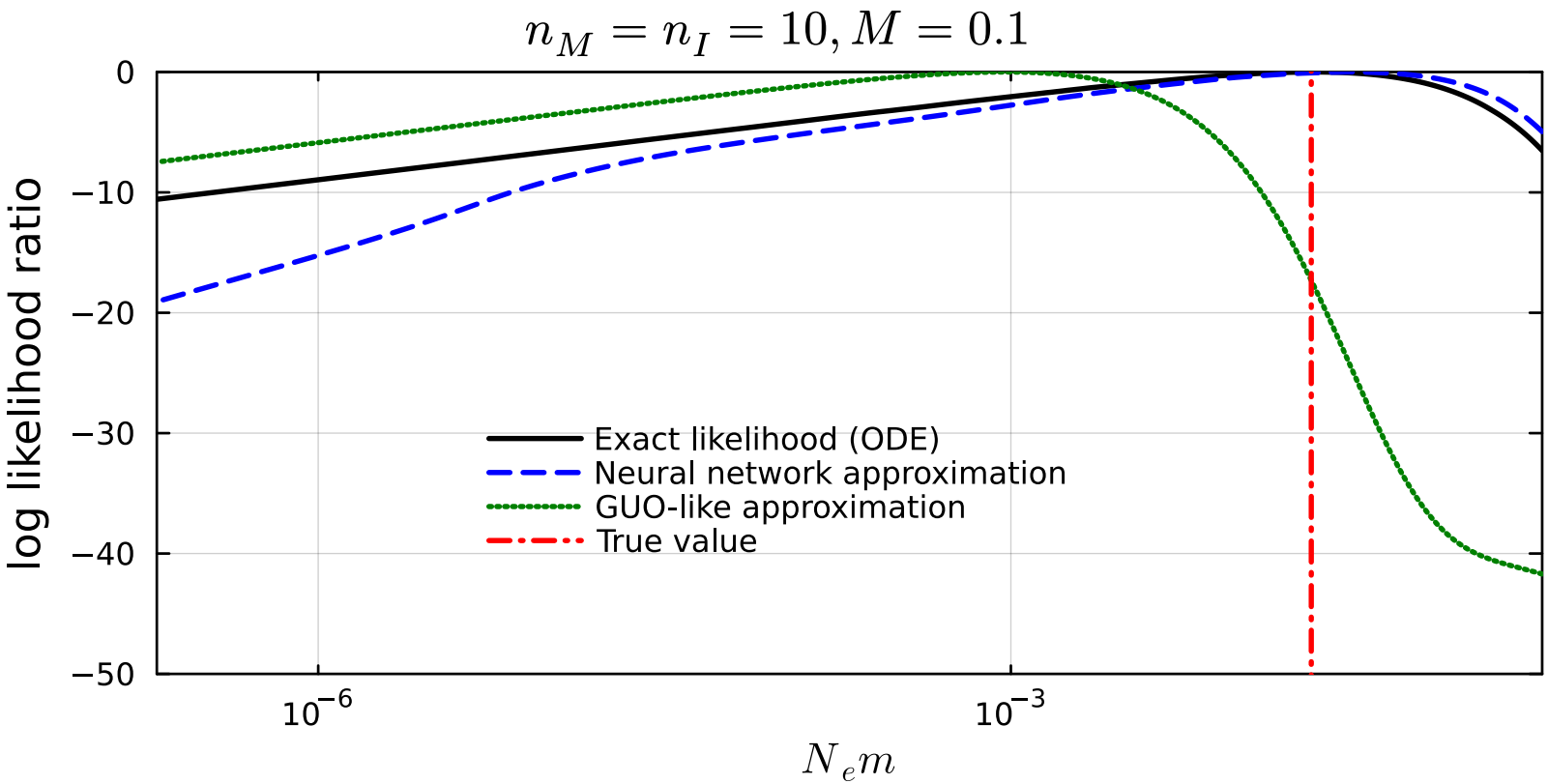
which turns out to be equivalent to:

$$\arg\min_{\varphi} \sum_{i=1}^{N} - \log q_{\varphi(\text{tree}_i \mid \theta_i)}$$

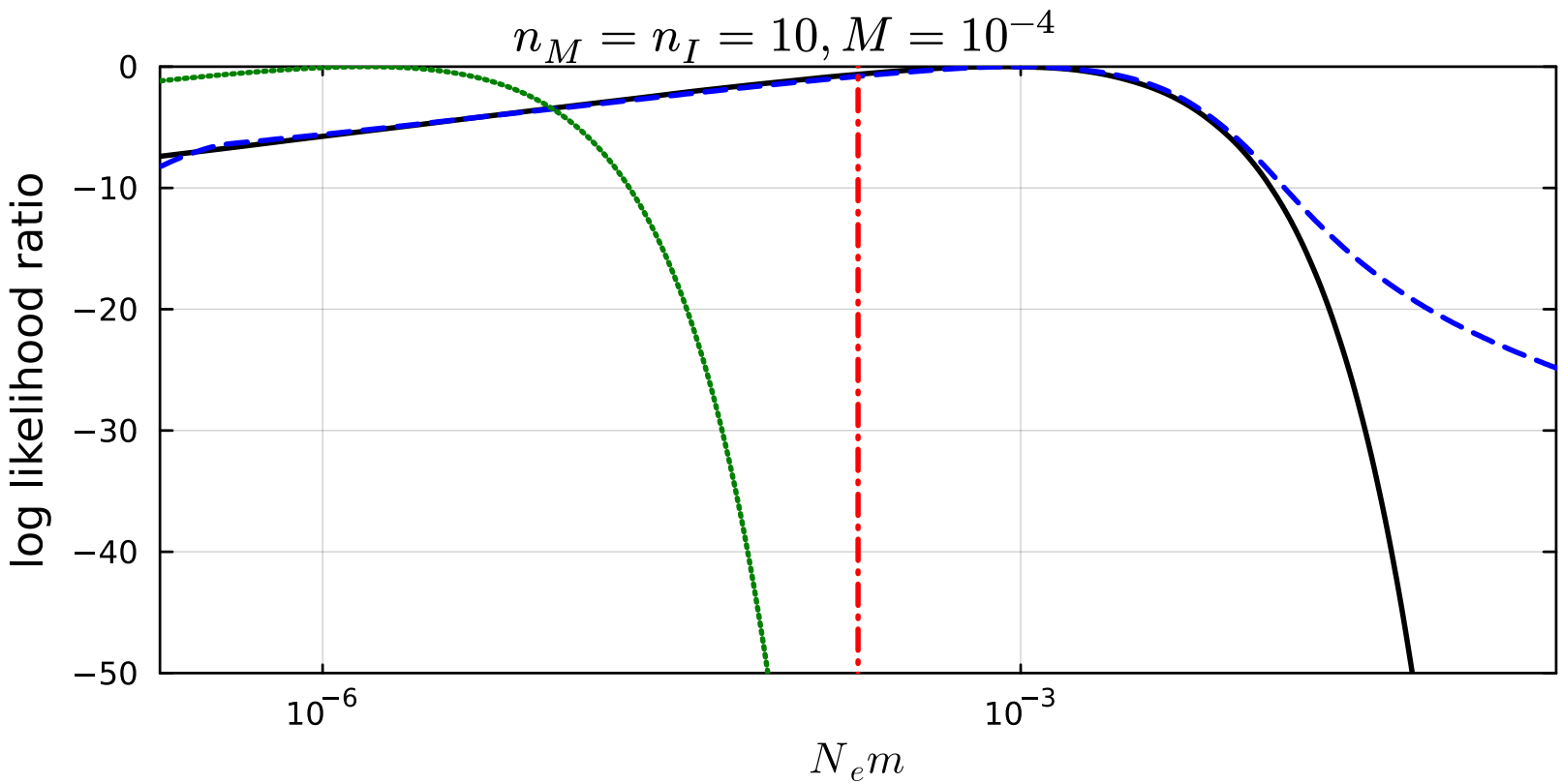
Neural likelihood



Neural likelihood



Neural likelihood



Calibration

$$\theta^* \sim P(\theta \mid y_{\text{obs}})$$
$$y_{\text{sim}} \sim P(y \mid \theta^*)$$

We fit to an augmented dataset

$$\theta^{**} \sim P(\theta \mid y_{\text{obs}}, y_{\text{sim}})$$

The original posterior, θ^* and θ^{**} should be indistinguishable if the model is properly calibrated.

Säilynoja *et al.*, Statistics and Computing (2026)

Calibration

The power-composite log-likelihood is written as

$$\alpha \cdot \log \mathcal{L}(\theta, y)$$

I treat α as a hyperparameter that controls the number of "effective independent observations" and do binary search over $\alpha \in (0, 1]$.