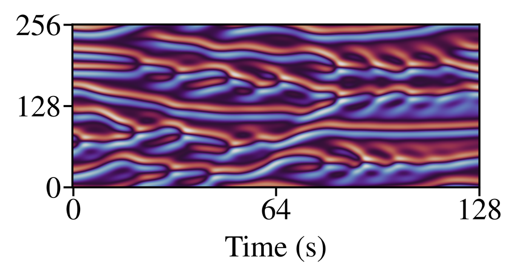
On conditional diffusion models for PDE simulations

Aliaksandra Shysheya[1*], Cristiana Diaconu[1*], Federico Bergamin[2*], Paris Perdikaris[3], José Miguel Hernández-Lobato[1], Richard E. Turner[1,3,4], Emile Mathieu[1]
[1]University of Cambridge, [2]Technical University of Denmark, [3]Microsoft AI4Science, [4]The Alan Turing Institute

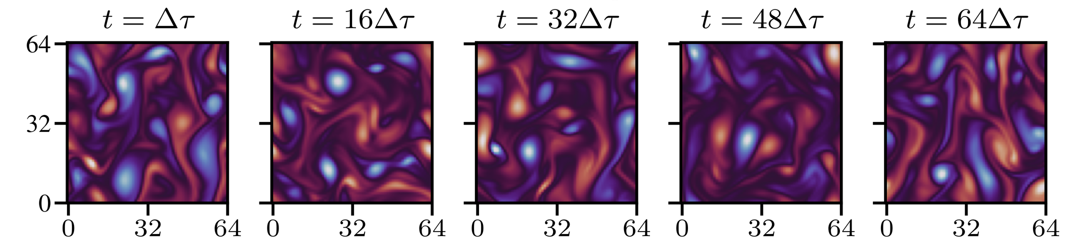
1 Problem Setup

We have access to some accurate numerical solutions from conventional solvers discretised in space and time $x_{1:L} = (x_1, \dots, x_L) \in \mathbb{R}^{D \times L}$

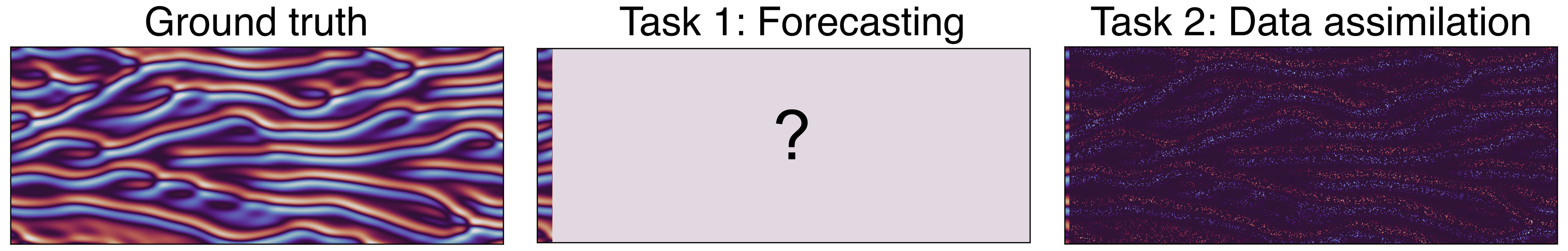
1D - Kuramoto-Sivashinsky (KS)



2D - Kolmogorov



PDE modelling tasks



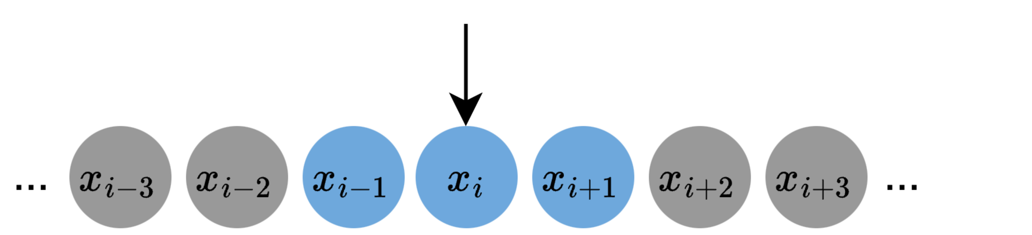
Research question

How can we **flexibly** condition diffusion models on observations in order to generate accurate PDE solutions?

2 Methodology

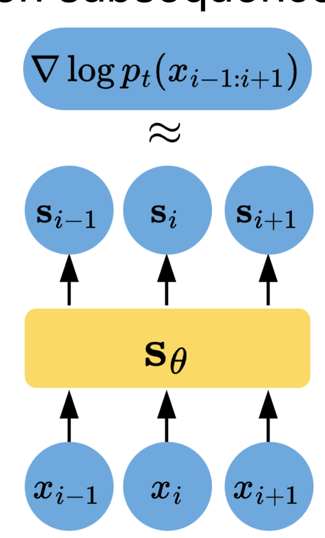
a) Tackling variable-length trajectories [1]

Assume k-order Markov structure on $x_{1:L} \in \mathbb{R}^{D \times L}$.



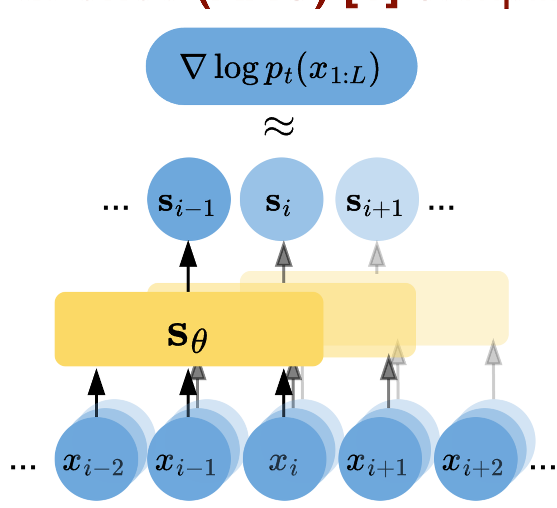
$$\nabla_{x_i(t)} \log p(x_{1:L}(t)) \approx \nabla_{x_i(t)} \log p(x_{i-k:i+k}(t))$$

Train score network on subsequences



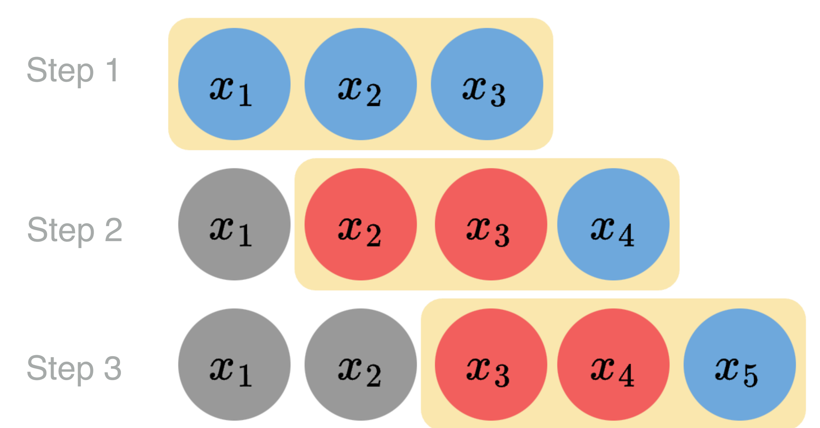
b) Sampling full trajectories

All-at-once (AAO) [1] sampling



At each denoising step, compose local scores to sample the full trajectory

Autoregressive (AR) sampling



Generate P states at a time conditioned on the last previously-generated C states

2 Methodology (cont'd)

Design space for diffusion-based models

MODEL	SCORE	ROLLOUT	CONDITIONING
Joint AAO [1]	$s_\theta(t, x_{1:L}(t))$	AAO	Guidance
Joint AR (ours)	$s_\theta(t, x_{1:L}(t))$	AR	Guidance
Amortised [2]	$s_\theta(t, x_{1:L}(t), y)$	AR	Architecture
Universal amortised (ours)	$s_\theta(t, x_{1:L}(t), y)$	AR	Architecture/Guidance

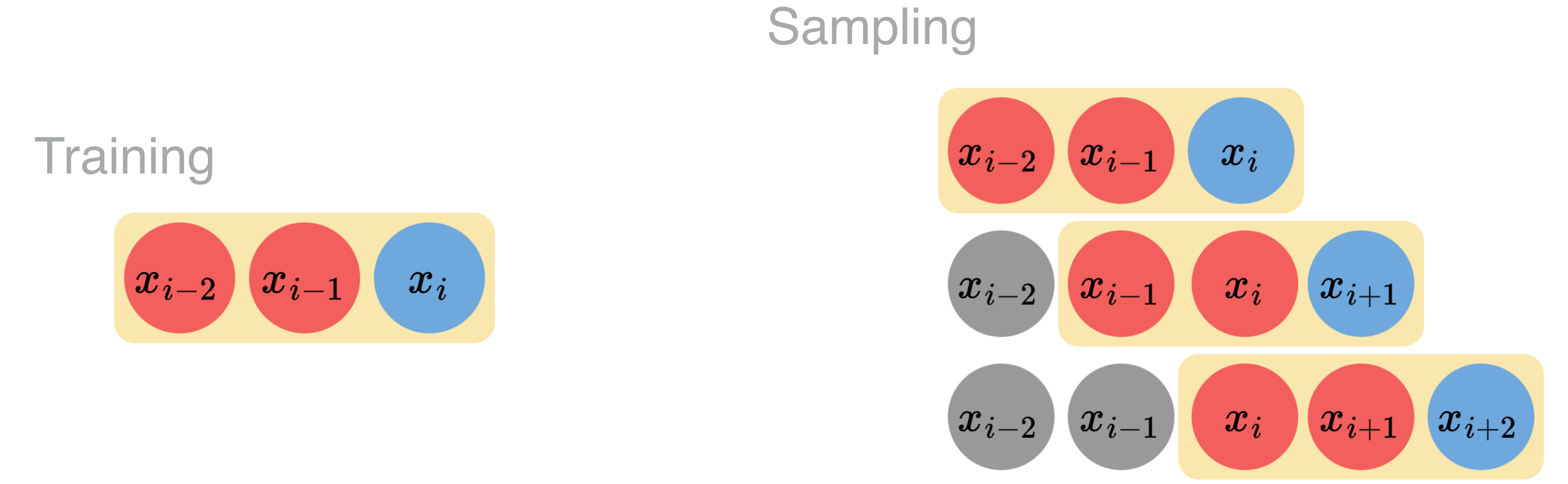
c) Conditioning

Joint model $s_\theta(t, x_{i-k:i+k}(t))$

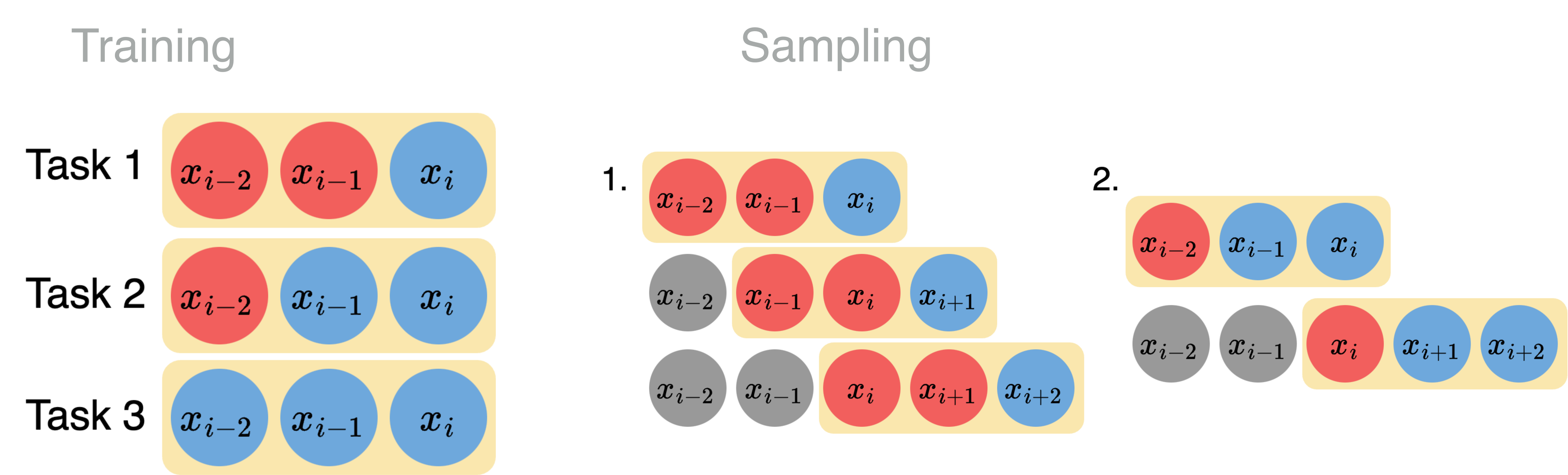
$$\nabla \log p(x_{i-k:i+k}(t)|y) \approx \underbrace{s_\theta(t, x_{i-k:i+k}(t))}_{\text{Unconditional model}} + \underbrace{\nabla \log p(y|x_{i-k:i+k}(t))}_{\text{Reconstruction guidance [3]}}$$

Amortised model $s_\theta(t, x_{i-k:i+k}(t), y)$

Plain amortised - **fix** number of conditioning states at training time [2]



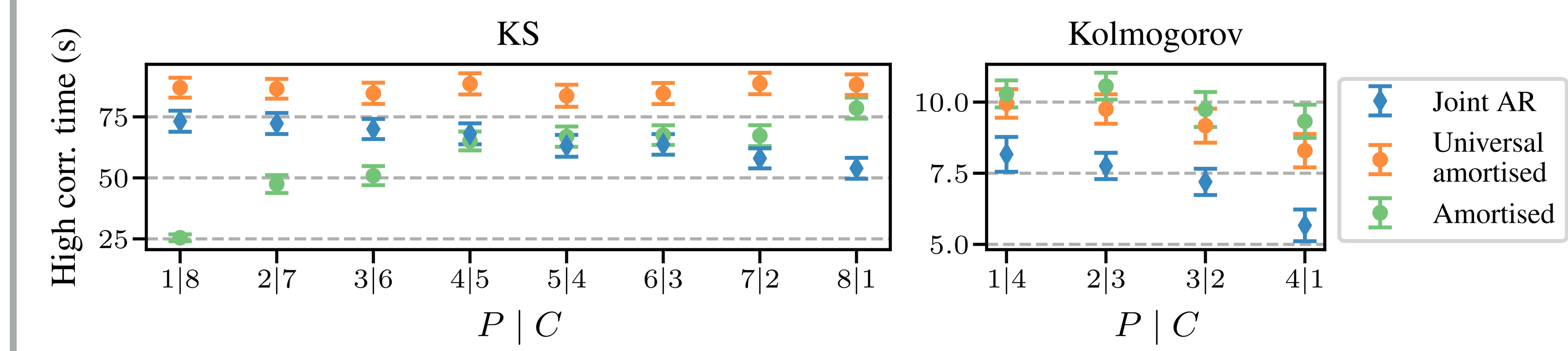
Universal amortised - train over a variety of tasks $\Longrightarrow$ flexible sampling



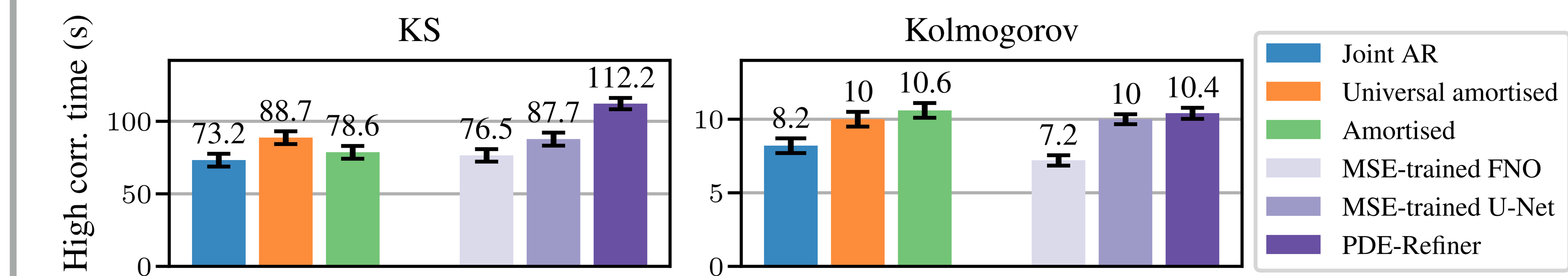
Condition amortised models through both the architecture (on previously-generated states) and through reconstruction guidance (on sparse observations) to tackle DA.

3 Experiments

Forecasting - predict future states given C past states

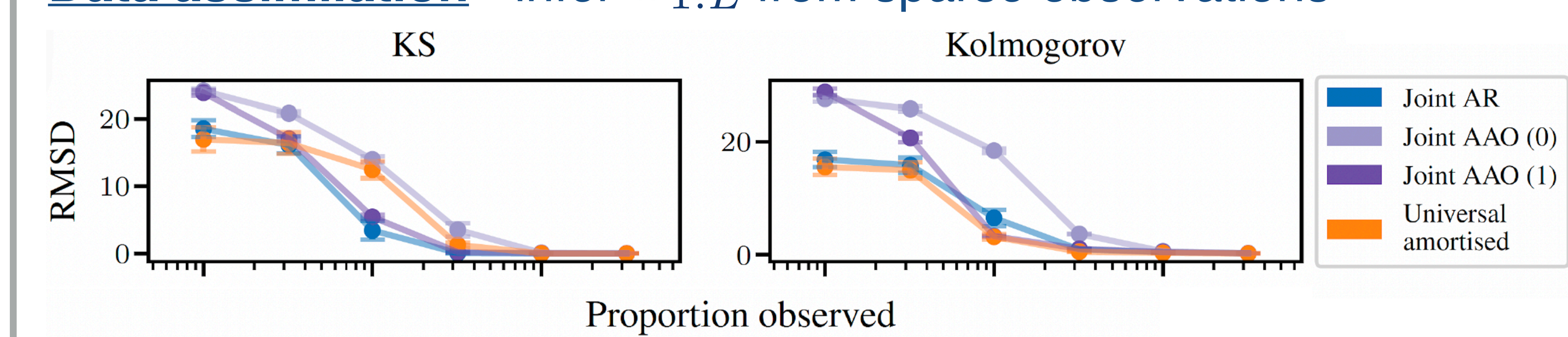


- **Universal amortised** outperforms **Joint AR** on both datasets for any $P \mid C$
- On KS - **Universal amortised** is stable across all $P \mid C$
- On Kolmogorov - **Amortised** perform a bit better than **Universal amortised**



- Amortised diffusion-based models perform on par with MSE-trained baseline, while offering more **flexibility**

Data assimilation - infer $x_{1:L}$ from sparse observations



- High sparsity: **Joint AR** and **Universal amortised** outperform **Joint AAO**
- Low-Moderate sparsity: comparable performance for all methods, but **Joint AAO** requires expensive corrector steps

Conclusions

1. Diffusion models perform **on par** with deterministic baselines, but benefit from more **flexibility -** the same trained model can tackle forecasting and DA
2. Yet, they underperform compared to SOTA task-specific models [4]
3. Unclear how findings depend on PDE characteristics (time discretisation, data volume, frequency spectrum, etc.)

References

[1] Rozet, F. and Louppe, G. Score-based Data Assimilation. In Advances in Neural Information Processing Systems (NeurIPS), 2023.

[2] Kohl, G., Chen, L.-W., and Thuerey, N. Turbulent flow simulation using autoregressive conditional diffusion models, 2023.

[3] Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models. In Deep Generative Models for Highly Structured Data Workshop, ICLR, volume 10, 2022.

[4] Phillip Lippe, Bastiaan S. Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. PDE-Refiner: Achieving Accurate Long Rollouts with Neural PDE Solvers. In Advances in Neural Information Processing Systems (NeurIPS), 2023.

Check the paper