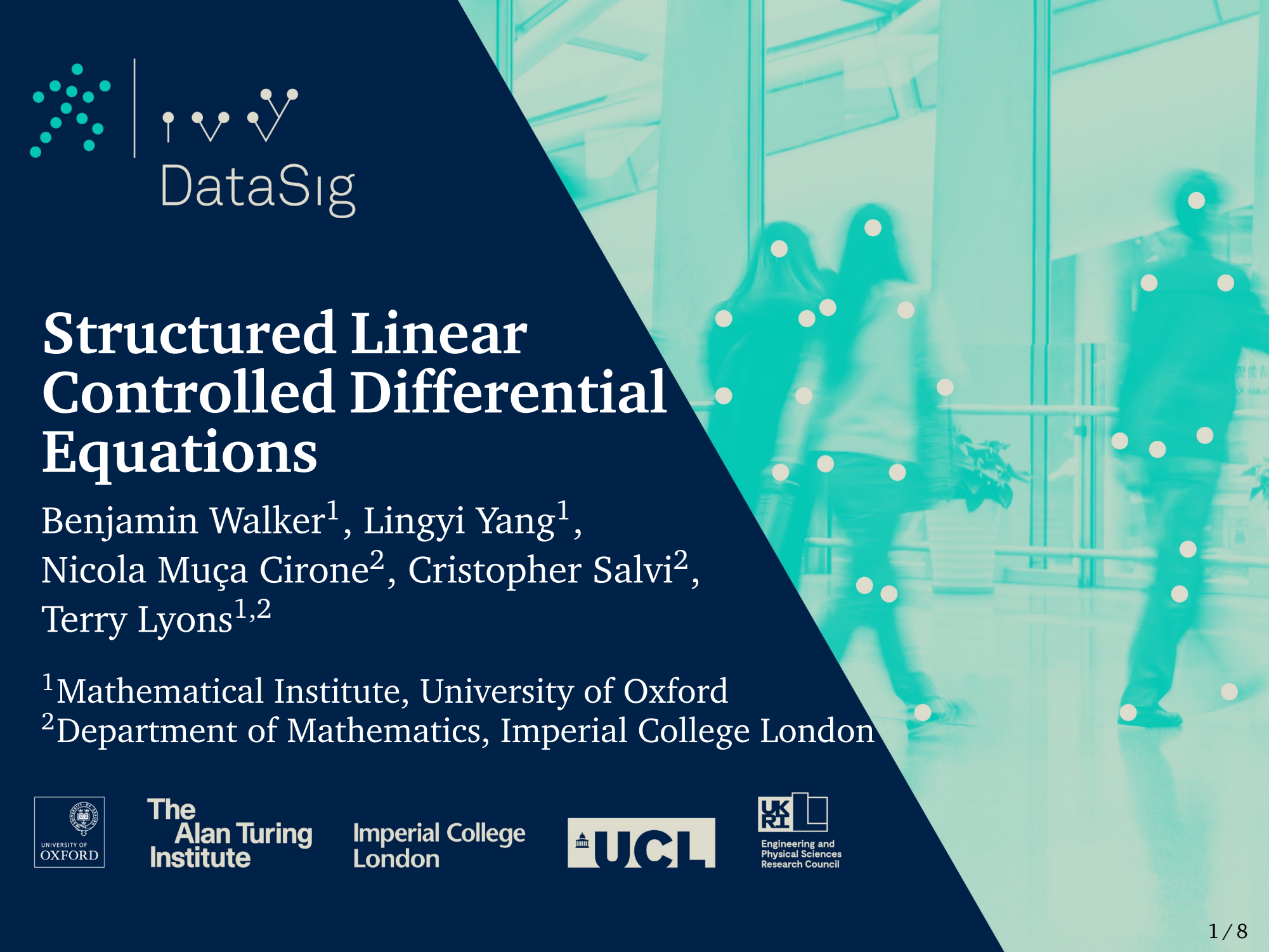


DataSig

Structured Linear Controlled Differential Equations

Benjamin Walker[1], Lingyi Yang[1],
Nicola Muça Cirone[2], Cristopher Salvi[2],
Terry Lyons[1,2]

[1]Mathematical Institute, University of Oxford
[2]Department of Mathematics, Imperial College London

University of Oxford · The Alan Turing Institute · Imperial College London · UCL · UKRI Engineering and Physical Sciences Research Council

Introduction

Structured linear controlled differential equations (SLiCEs) are a novel framework for understanding RNNs of the form:

$$h_{i+1} = A_\theta(x_i)h_i + B_\theta x_i,$$

where $x_i \in \mathbb{R}^{d_x}$, $h_i \in \mathbb{R}^{d_h}$, $A_\theta : \mathbb{R}^{d_x} \to \mathbb{R}^{d_h \times d_h}$, and $B_\theta \in \mathbb{R}^{d_h \times d_x}$.

Introduction

Structured linear controlled differential equations (SLiCEs) are a novel framework for understanding RNNs of the form:

$$h_{i+1} = A_\theta(x_i)h_i + B_\theta x_i,$$

where $x_i \in \mathbb{R}^{d_x}$, $h_i \in \mathbb{R}^{d_h}$, $A_\theta : \mathbb{R}^{d_x} \to \mathbb{R}^{d_h \times d_h}$, and $B_\theta \in \mathbb{R}^{d_h \times d_x}$. Examples include

Introduction

Structured linear controlled differential equations (SLiCEs) are a novel framework for understanding RNNs of the form:

$$h_{i+1} = A_\theta(x_i)h_i + B_\theta x_i,$$

where $x_i \in \mathbb{R}^{d_x}$, $h_i \in \mathbb{R}^{d_h}$, $A_\theta : \mathbb{R}^{d_x} \to \mathbb{R}^{d_h \times d_h}$, and $B_\theta \in \mathbb{R}^{d_h \times d_x}$. Examples include Mamba (Gu et al. 2023), Mamba-2 (Dao et al. 2024), DeltaNet (Yang et al. 2024b), input-dependent block-diagonal LRNN (Fan et al. 2024), DeltaProduct (Siems et al. 2025), Gated DeltaNet (Yang et al. 2025), RWKV-7 (Peng et al. 2025), HGRN-2 (Qin et al. 2024), mLSTM (Beck et al. 2024), Gated Linear Attention (Yang et al. 2024a), Gated Random Feature Attention (Peng et al. 2021), Gated Slot Attention (Zhang et al. 2024), TTT-Linear (Sun et al. 2025), etc.

Introduction

Structured linear controlled differential equations (SLiCEs) are a novel framework for understanding RNNs of the form:

$$h_{i+1} = A_\theta(x_i)h_i + B_\theta x_i,$$

where $x_i \in \mathbb{R}^{d_x}$, $h_i \in \mathbb{R}^{d_h}$, $A_\theta : \mathbb{R}^{d_x} \to \mathbb{R}^{d_h \times d_h}$, and $B_\theta \in \mathbb{R}^{d_h \times d_x}$. Examples include Mamba (Gu et al. 2023), Mamba-2 (Dao et al. 2024), DeltaNet (Yang et al. 2024b), ...

For more details see S. Yang et al. (2024b). "Parallelizing Linear Transformers with the Delta Rule over Sequence Length". In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*

Efficiency vs Expressivity

Why so many variants?

[1] N. M. Cirone et al. (2024). "Theoretical Foundations of Deep Selective State-Space Models". In: *Advances in Neural Information Processing Systems*

Efficiency vs Expressivity

Why so many variants?

- Dense matrices are maximally expressive but expensive[1].

[1] N. M. Cirone et al. (2024). "Theoretical Foundations of Deep Selective State-Space Models". In: *Advances in Neural Information Processing Systems*

Efficiency vs Expressivity

Why so many variants?

- Dense matrices are maximally expressive but expensive[1].
- Diagonal matrices are cheap but not maximally expressive[1].

[1] N. M. Cirone et al. (2024). "Theoretical Foundations of Deep Selective State-Space Models". In: *Advances in Neural Information Processing Systems*

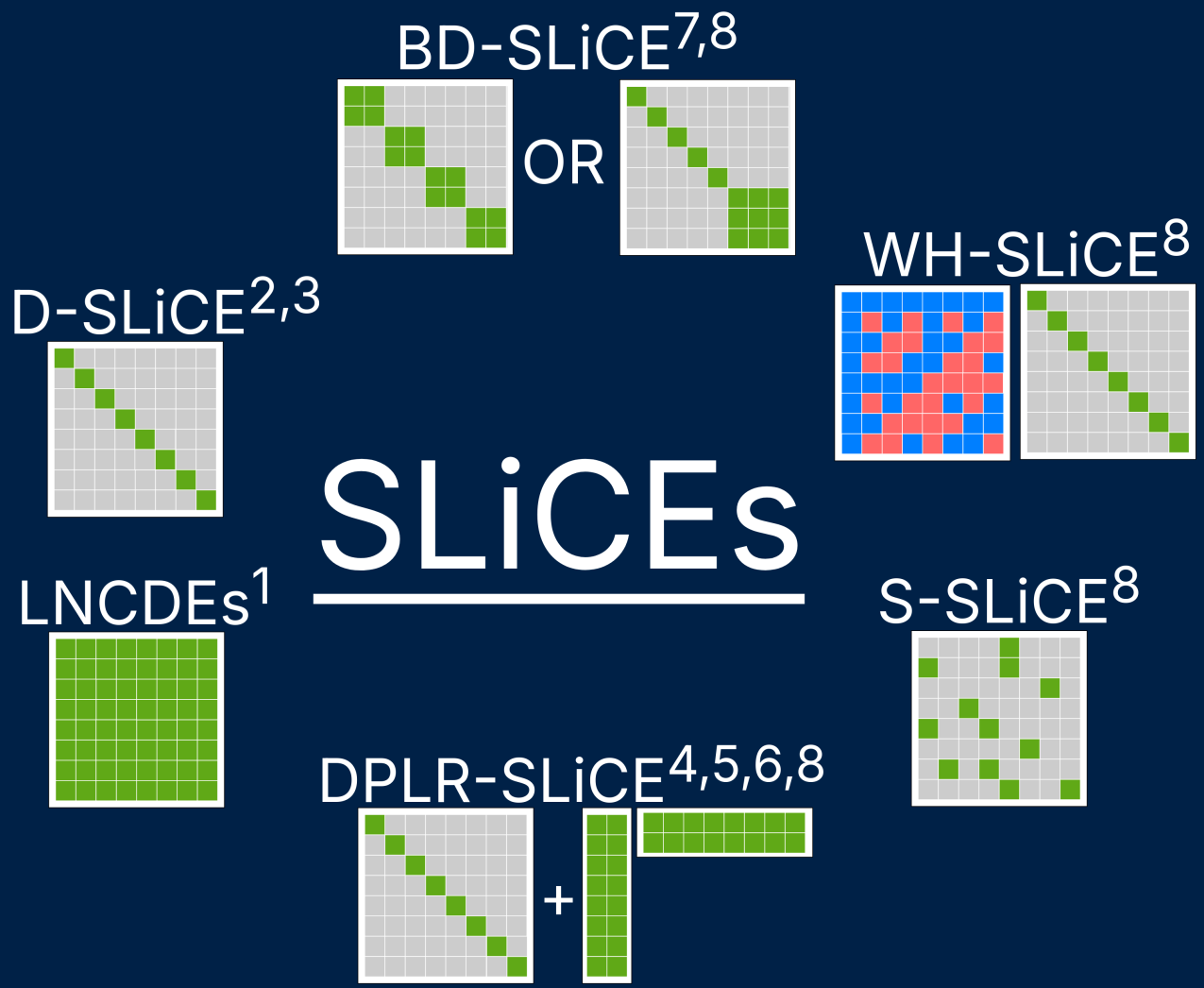
Efficiency vs Expressivity

Why so many variants?

- Dense matrices are maximally expressive but expensive[1].
- Diagonal matrices are cheap but not maximally expressive[1].

Is there a balance?

[1] N. M. Cirone et al. (2024). "Theoretical Foundations of Deep Selective State-Space Models". In: *Advances in Neural Information Processing Systems*



1.Cirone, N. M. et al. (2024). Theoretical foundations of deep selective state-space models. NeurIPS.

2. Gu, A. et al. (2022). Efficiently modeling long sequences with structured state spaces. ICLR.

3. Gu, A. et al. (2024). *Mamba: Linear-time sequence modeling with selective state spaces. COLM.*

4. Schlag, I. et al. (2021). Linear transformers are secretly fast weight programmers. ICML.

5. Yang, S. et al. (2024). Parallelizing linear transformers with the delta rule over sequence length. NeurIPS.

6. Siems, J. et al. (2025). DeltaProduct: Improving state-tracking in linear RNNs via Householder products. ICLR.

7. Fan, T. et al. (2024). Advancing regular language reasoning in linear recurrent neural networks. NAACL.

8. Walker, B. et al. (2025). Structured Linear CDEs: Maximally Expressive and Parallel-in-Time Models. NeurIPS.

SLiCE Comparison

Model	Parameters	Recurrent Cost	Parallel Cost	Maximally Expressive
DE-LNCDEs	$\mathcal{O}(d_h^3)$	$\mathcal{O}(nd_h^3)$	$\mathcal{O}(\log(n), d_h^3)$	Yes

SLiCE Comparison

Model	Parameters	Recurrent Cost	Parallel Cost	Maximally Expressive
DE-LNCDEs	$\mathcal{O}(d_h^3)$	$\mathcal{O}(nd_h^3)$	$\mathcal{O}(\log(n), d_h^3)$	Yes
D-SLiCEs	$\mathcal{O}(d_h^2)$	$\mathcal{O}(nd_h^2)$	$\mathcal{O}(\log(n), d_h^2)$	No

SLiCE Comparison

Model	Parameters	Recurrent Cost	Parallel Cost	Maximally Expressive
DE-LNCDEs	$\mathcal{O}(d_h^3)$	$\mathcal{O}(nd_h^3)$	$\mathcal{O}(\log(n), d_h^3)$	Yes
D-SLiCEs	$\mathcal{O}(d_h^2)$	$\mathcal{O}(nd_h^2)$	$\mathcal{O}(\log(n), d_h^2)$	No
DPLR-SLiCEs	$\mathcal{O}(rd_h^2)$	$\mathcal{O}(nrd_h^2)$	$\mathcal{O}(\log(n), d_h^3)$	Yes
S–SLiCEs	$\mathcal{O}(d_h^{2+\epsilon})$	$\mathcal{O}(nd_h^{2+\epsilon})$	$\mathcal{O}(\log(n), d_h^3)$	Yes
WH–SLiCEs	$\mathcal{O}(d_h^2)$	$\mathcal{O}(nd_h^2)$	$\mathcal{O}(\log(n), d_h^3)$	Yes
BD–SLiCEs	$\mathcal{O}\left(d_h \sum_j b_j^2\right)$	$\mathcal{O}\left(nd_h \sum_j b_j^2\right)$	$\mathcal{O}\left(\log(n), d_h \sum_j b_j^2\right)$	Yes

Table 1: Comparison of SLiCEs on parameter count, computational cost, and expressivity. Parallel cost is measured as $\mathcal{O}$(scan depth, cost per composition) when applying a parallel associative scan.

Length Generalisation

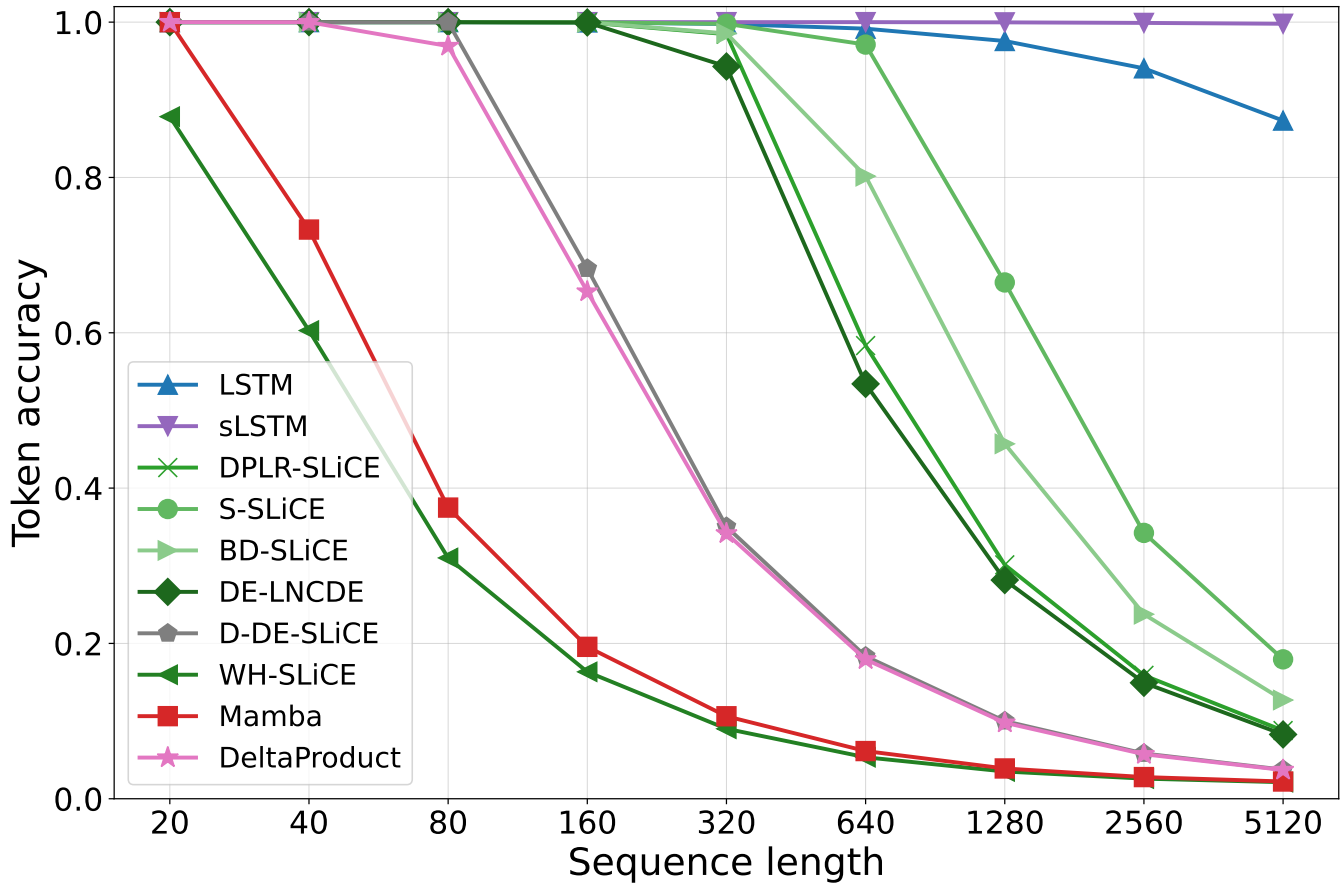


Figure 1: Length generalisation on the A_5 benchmark.

Time Series Classification

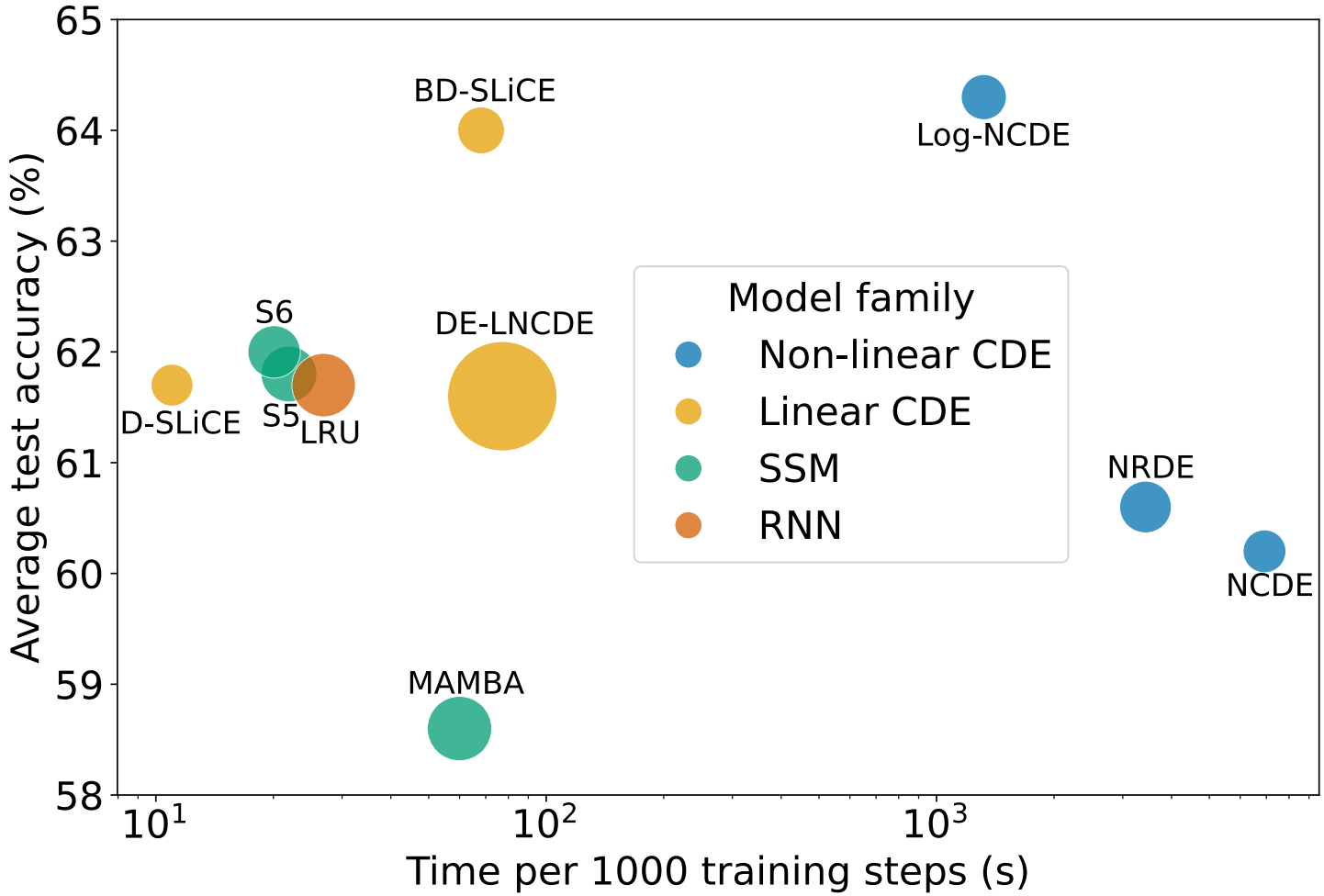


Figure 2: Accuracy, speed, and memory footprint on the UEA-MTSCA.

Thank you!

walkerb1@maths.ox.ac.uk