

Deep-Learning-Based Model Augmentation of Vehicle Dynamics

Bendegúz M. Györök¹, Jan H. Hoekstra², Tamás Péni¹, Maarten Schoukens², Roland Tóth^{1,2}

gyorokbende@sztaki.hu

¹HUN-REN Institute for Computer Science and Control (SZTAKI)

²Eindhoven University of Technology

Deep-learning-based model augmentation

Physics-based modeling

- Interpretable model
- Hard to model complex effects

Data-driven identification

- High accuracy
- Hard to interpret
- Inaccurate extrapolation

Data-driven model augmentation

- Utilizing the physics-based model
- Efficient augmentation with deep-learning-based techniques
- Goal: highly accurate, yet interpretable model

Simple model augmentation structures

DT NL-SS model

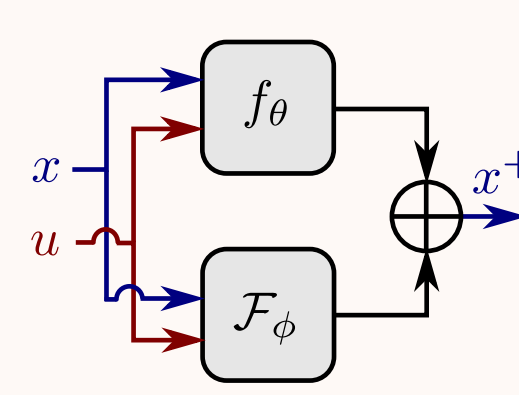
$$x_{k+1} = f_\theta(x_k, u_k)$$

$$\hat{y}_k = h_\theta(x_k, u_k)$$

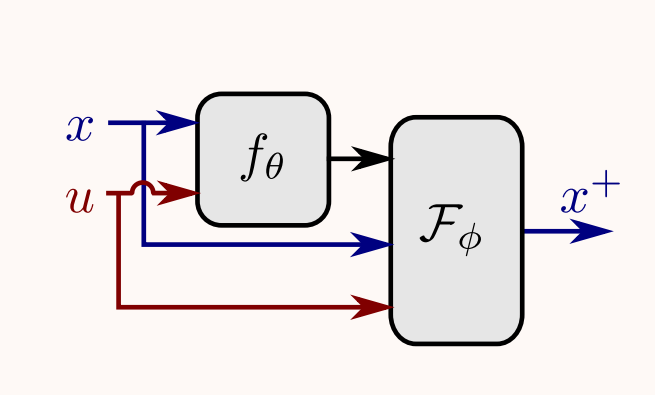
Neural networks

output = $\mathcal{F}_\phi$ (inputs)

Additive structure



Multiplicative structure



LFR-based augmentation structure

$$w = \mathcal{F}_\phi(z)$$

$$z = C_z \hat{x} + D_{zu} u + D_{zw} w$$

$$\hat{x}^+ = f_\theta(\hat{x}, u) + B_u w$$

$$\hat{y} = h_\theta(\hat{x}, u) + D_{yw} w$$

Currently not used to prevent algebraic loops (future possibility)

All LFR matrices are tuned during the optimization

Cost function

- Model augmentation embedded into SUBNET

$$\frac{1}{N_{\text{sec}}} \sum_{k \in \mathcal{I}_{N_{\text{sec}}}} \frac{1}{T} \sum_{i=0}^{T-1} \|y_{k+i} - \hat{y}_{k+i|k}\|_2^2$$

No. sections Section length

Training options

Optimized parameters

- Neural network parameters (ϕ)
- Option: tuning the physical parameters (θ)

Regularization of physical parameters

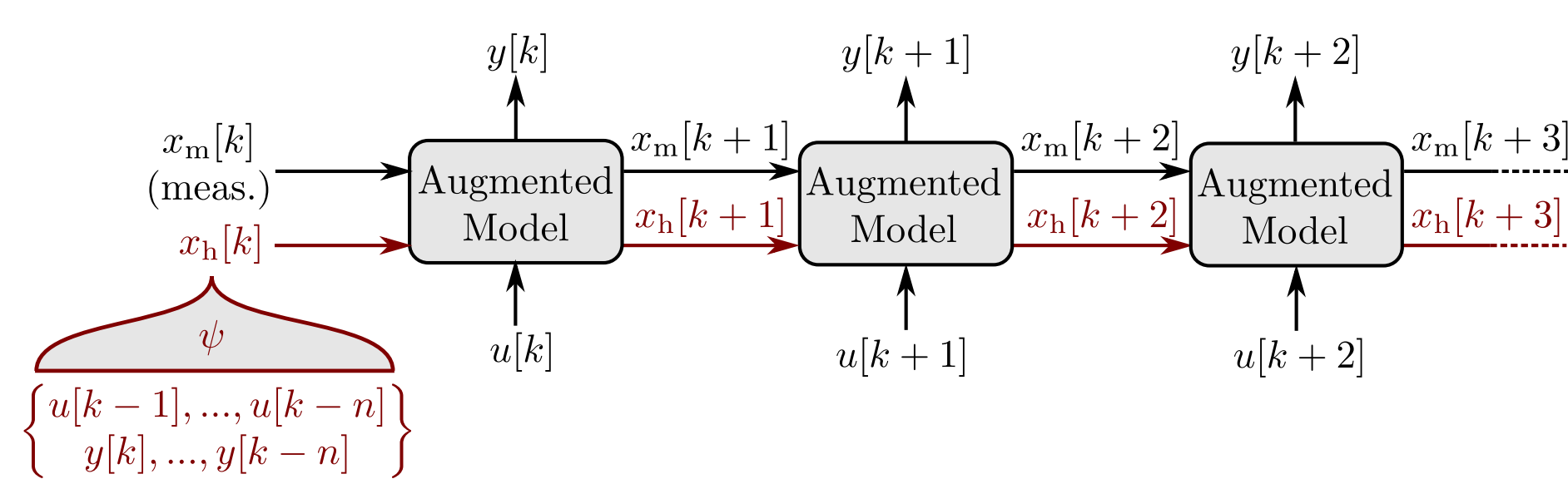
- Physical parameters should not reach unrealistic values
- $$V_{\mathcal{D}_N}^{(\text{reg})}(\phi, \theta) = V_{\mathcal{D}_N}^{(\text{sec})}(\phi, \theta) + \alpha \|\theta - \theta_0\|_{\Lambda_0}$$
- Original cost fun. Regularization parameter Normalization

Orthogonalization-based regularization

- ANN prioritizes identification of unknown dynamics
- $$\|\mathcal{F}_\phi\|_{\Pi_{x,u}} = \beta \|\Pi_{x,u} \mathcal{F}_\phi(x, u)\|_2^2$$
- Regul. coeff. Projection matrix

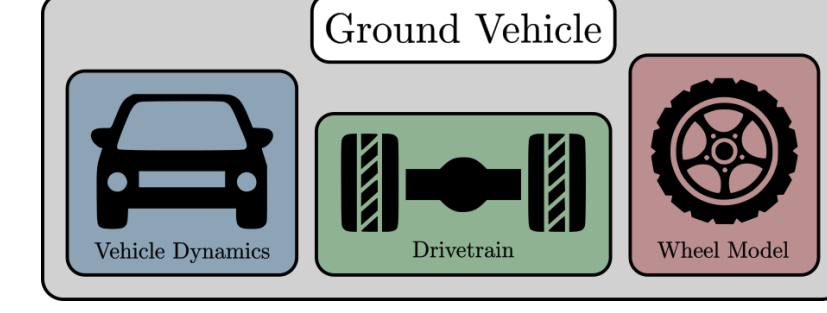
Dynamic model augmentation

- Introducing hidden states $\rightarrow$ memory effect of the unmodeled terms can be considered
- Hidden states are estimated by an encoder network similarly to the black-box case



First-principle model for augmentation

- Dynamic single track model
- Linearized Pacejka Formula
- Empirical drivetrain model



$$\dot{x} = v_\xi \cos \varphi - v_\eta \sin \varphi$$

$$\dot{y} = v_\xi \sin \varphi + v_\eta \cos \varphi$$

$$\dot{\varphi} = \omega$$

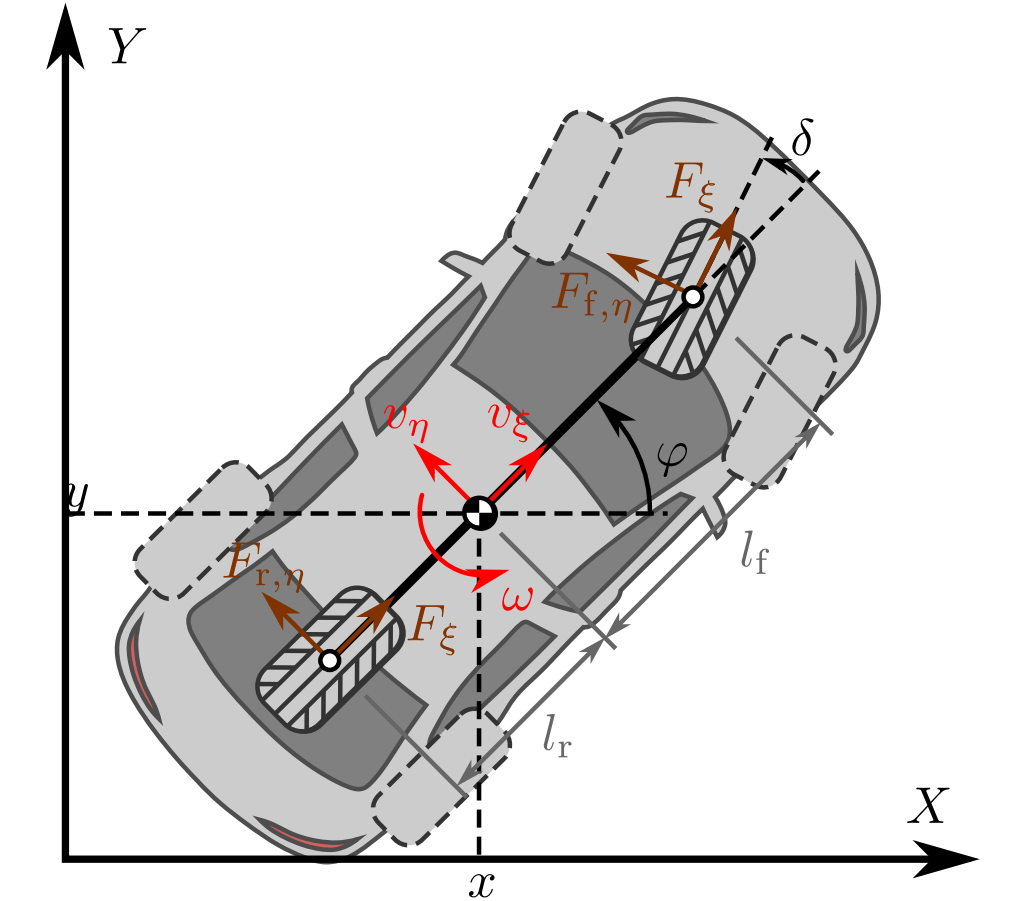
Kinematic relations (known dynamics)

$$\dot{v}_\xi = \frac{1}{m} (F_\xi + F_\xi \cos \delta - F_{f,\eta} \sin \delta + m v_\eta \omega)$$

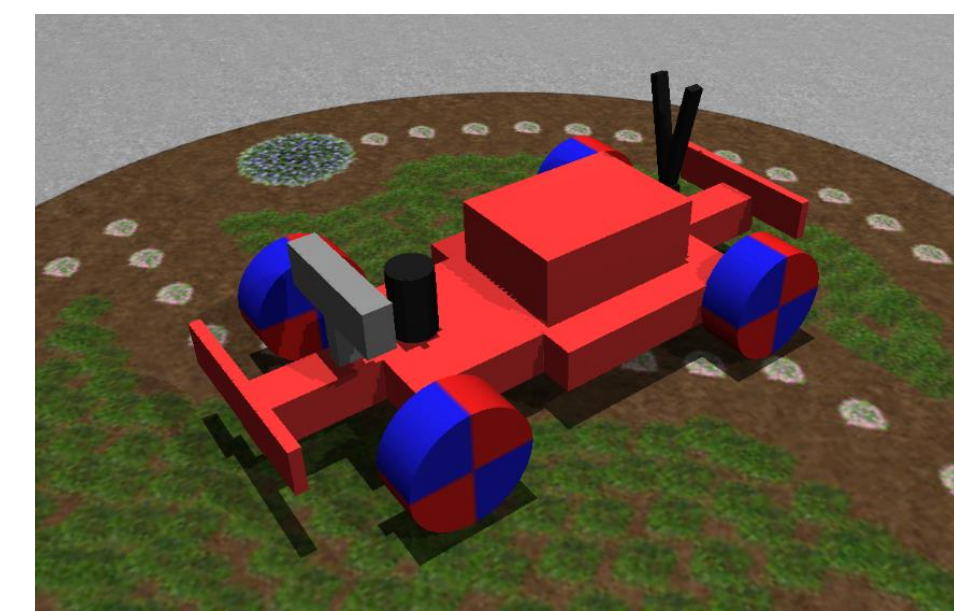
$$\dot{v}_\eta = \frac{1}{m} (F_{f,\eta} + F_\xi \sin \delta - F_{f,\eta} \cos \delta - m v_\xi \omega)$$

$$\dot{\omega} = \frac{1}{J_\xi} (F_{f,\eta} l_f \cos \delta + F_\xi l_f \sin \delta - F_\xi l_r)$$

Velocity relations (to be augmented / learned)



Data acquisition with a digital twin model



- Data gathered in MuJoCo high-fidelity simulator
- Two trajectories (circle, lemniscate), with multiple reference velocities
- Sampling frequency of 40 Hz
- 6701 data points for training, 1675 for validation, and 8283 for testing

The SUBNET structure

- Initial states are estimated by encoder network
- Multi-shooting-based loss function with truncated, overlapping sections
- CT SUBNET implements an ODE solver step with derivative normalization factor
- Initialization: randomly (e.g. Xavier-method)

Deep subspace encoders for nonlinear system identification



Deep learning of vehicle dynamics

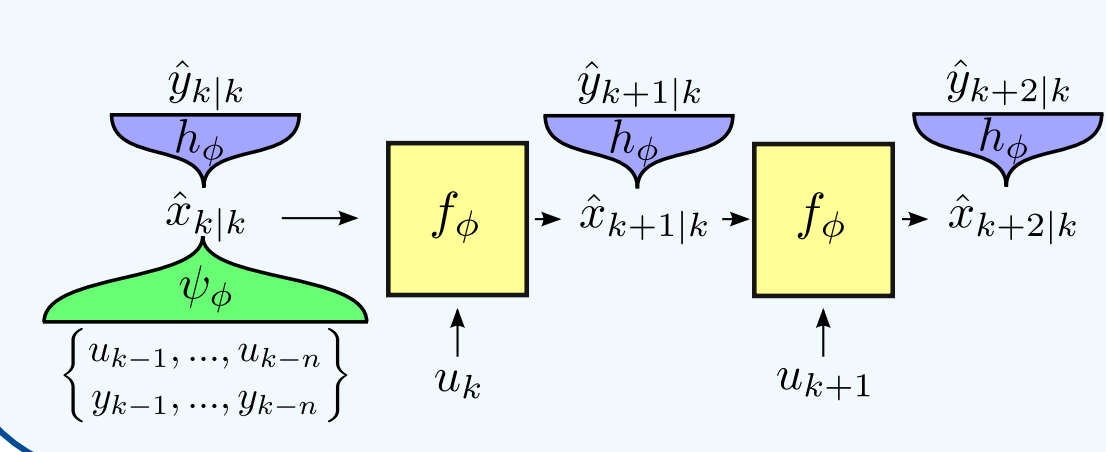


DT SUBNET model equations

$$\hat{x}_{k|k} = \psi_\phi(u_{k-n}^{k-1}, y_{k-n}^{k-1})$$

$$\hat{x}_{k+1|k} = f_\phi(\hat{x}_{k|k}, u_k)$$

$$\hat{y}_{k|k} = h_\phi(\hat{x}_{k|k}, u_k)$$

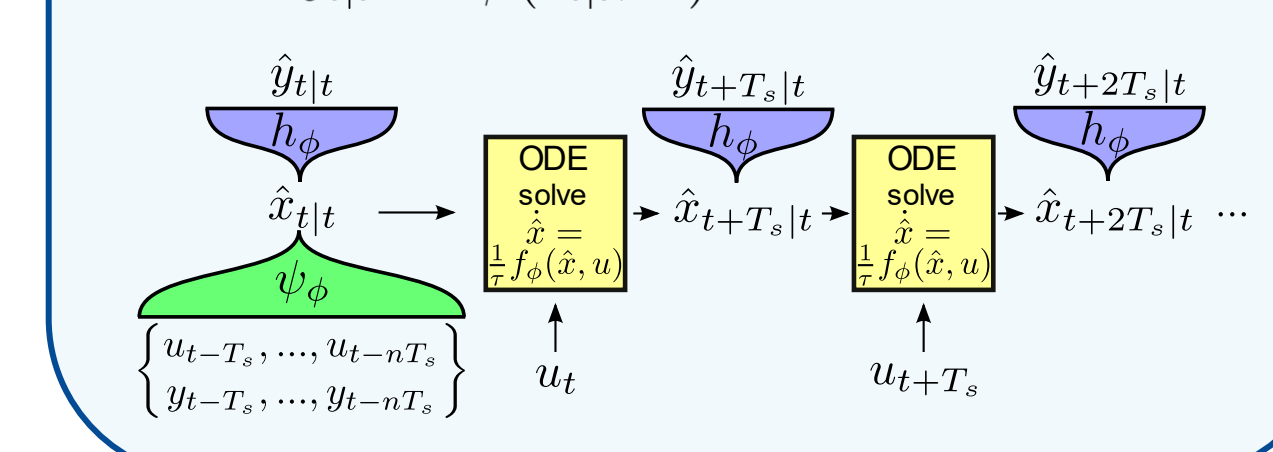


CT SUBNET model equations

$$\hat{x}_{t|t} = \psi_\phi(u_{t-nT_s}^{t-T_s}, y_{t-nT_s}^{t-T_s})$$

$$\hat{x}_{t+T_s|t} = \text{ODEsolve} \left\{ \frac{1}{T_s} f_\phi(\cdot, u_t), \hat{x}_{t|t}, T_s \right\}$$

$$\hat{y}_{t|t} = h_\phi(\hat{x}_{t|t}, u_t)$$



Results

Conclusions

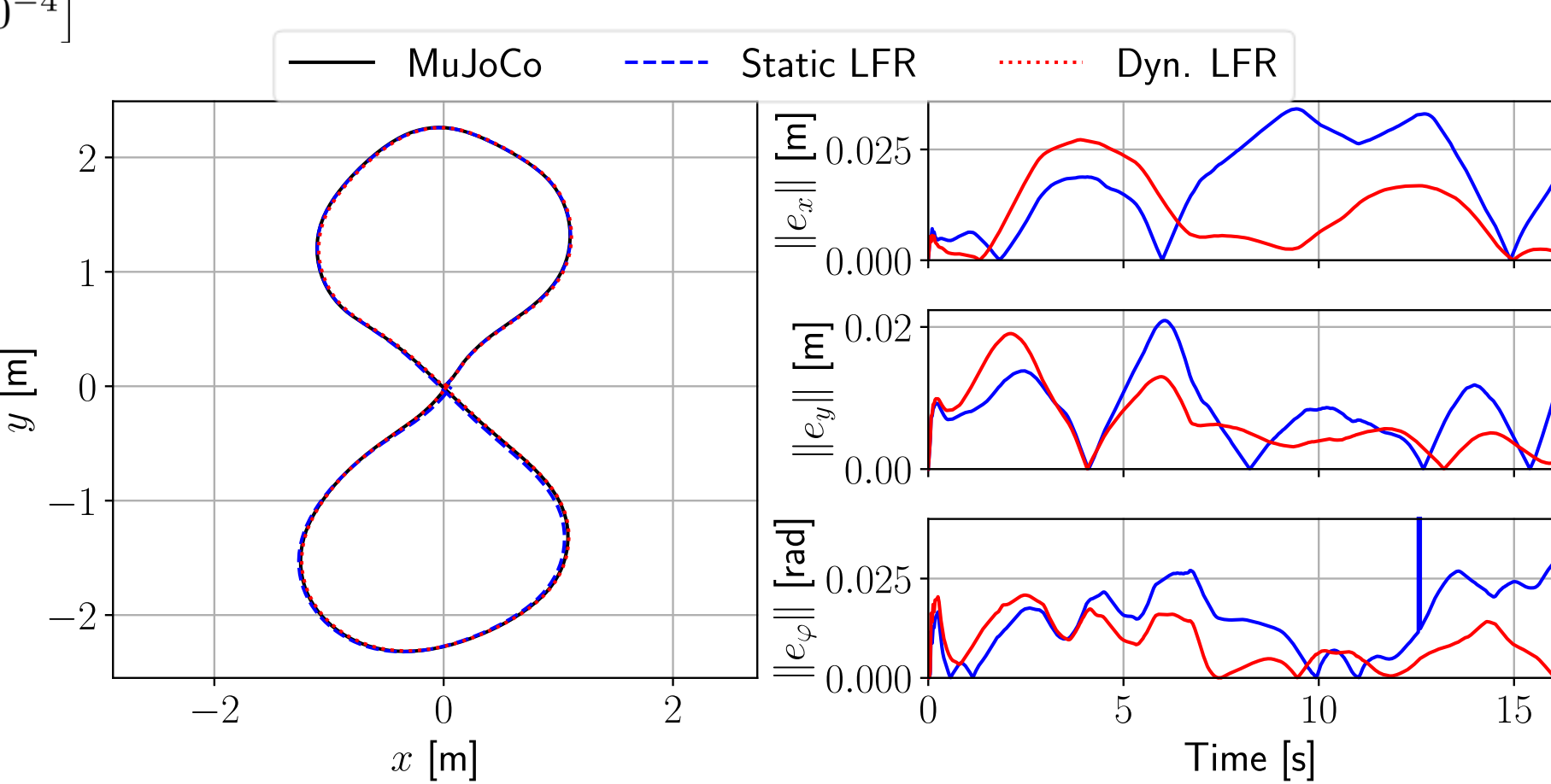
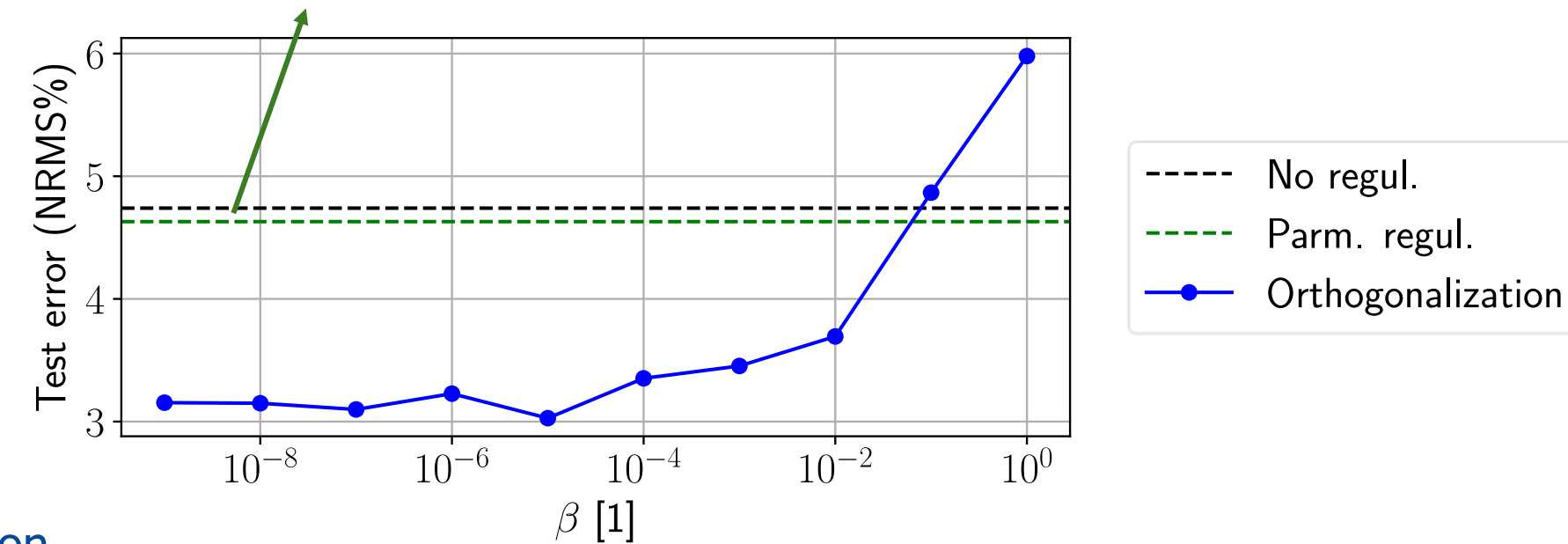
- Remarkable results considering 16-18 seconds long simulations
- Regularization and orthogonalization schemes are effective
- Dynamic LFR-based structure outperformed even the CT SUBNET

Model	Test error (NRMS)		
	Baseline	Tuned phys. parms.	Regularization
First-principle model	37.91%	-	-
DT SUBNET	2.12%	-	-
CT SUBNET	1.75%	-	-
Static additive	5.57%	4.74%	4.63%
Static multiplicative	3.04%	2.81%	2.59%
Static LFR-based	2.84%	2.17%	2.17%
Dynamic additive	2.49%	2.21%	2.13%
Dynamic multiplicative	2.35%	2.28%	2.21%
Dynamic LFR-based	1.16%	0.97%	0.94%

Hidden states: $n_h = 2$

Hyperparameters were tuned empirically.

Tuned regularization: $\alpha = 10^{-4}$



Another aspect: convergence speed

- CT SUBNET has shown slow convergence
- Static and dynamic augmentation structures converged similarly, dynamic has a small edge over static
- DT SUBNET is a bit slower than both model augmentation methods

