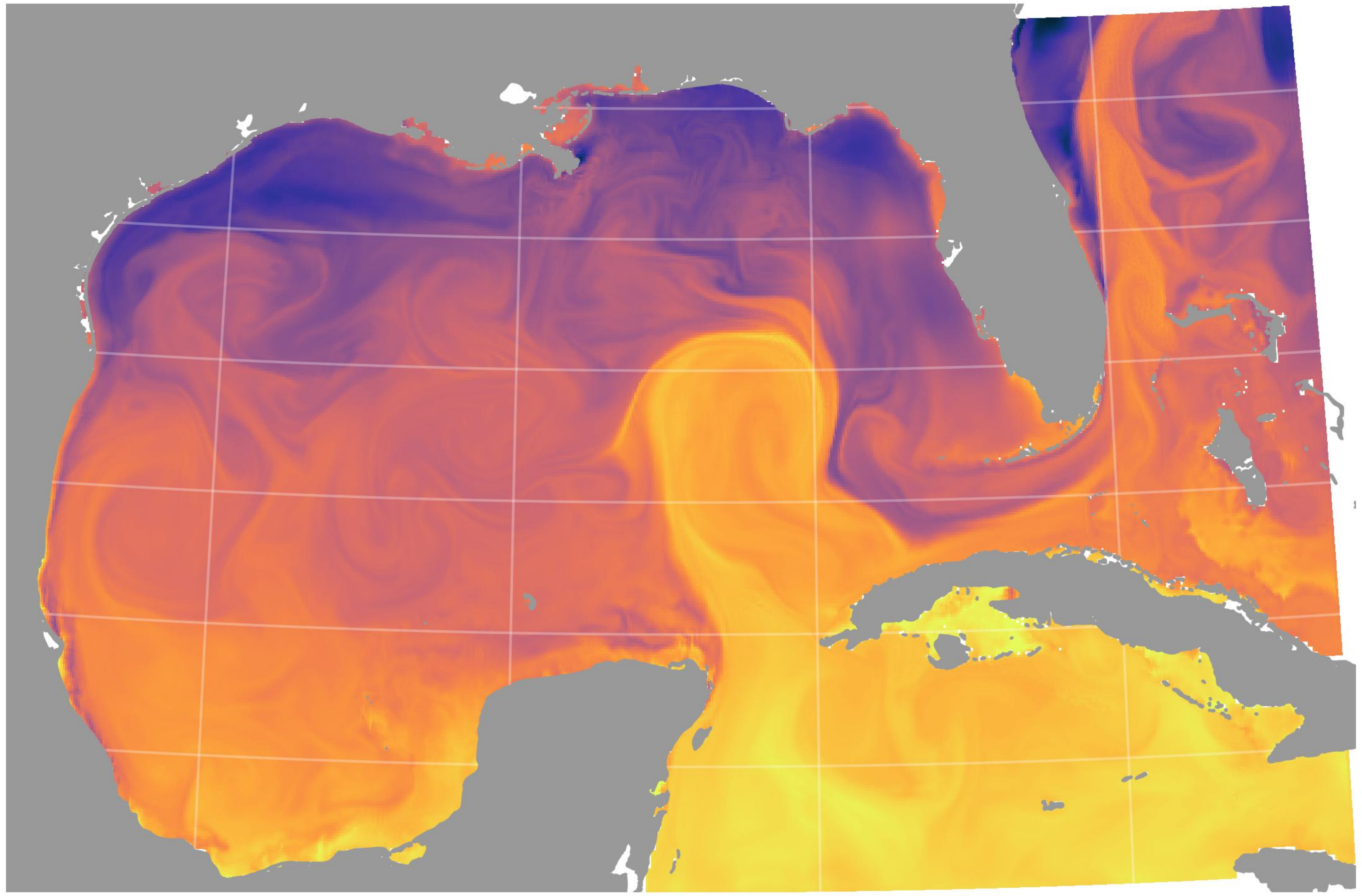
Recurrent Neural Network Emulation for High Resolution Forecasting



Timothy Smith, CIRES / NOAA

Stephen Penny, Sofar Ocean

Tse-Chun Chen, CIRES / NOAA

Jason Platt, UC San Diego

ECMWF-ESA ML Workshop

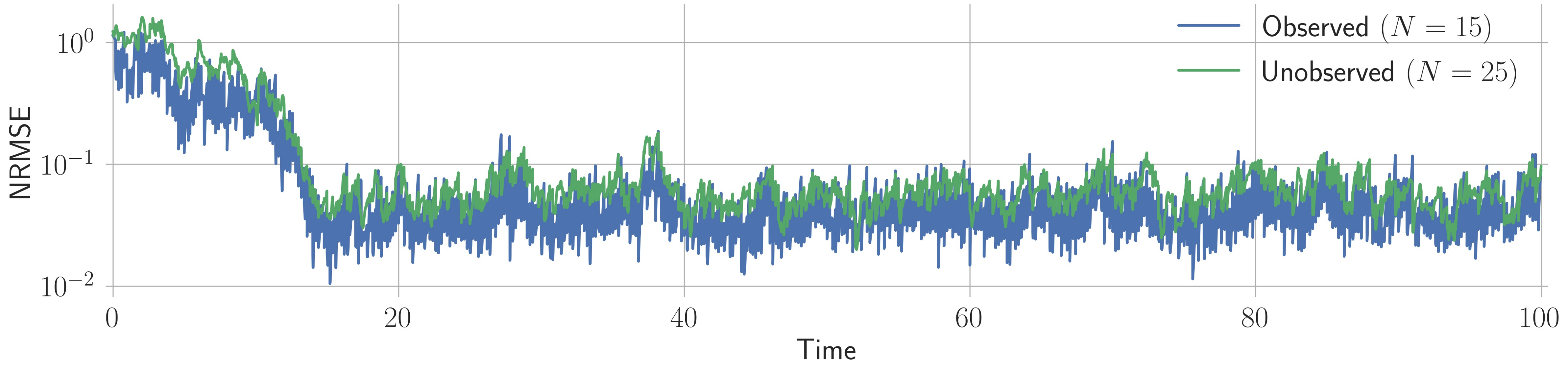
Nov. 16, 2022

Why Recurrent Neural Networks?

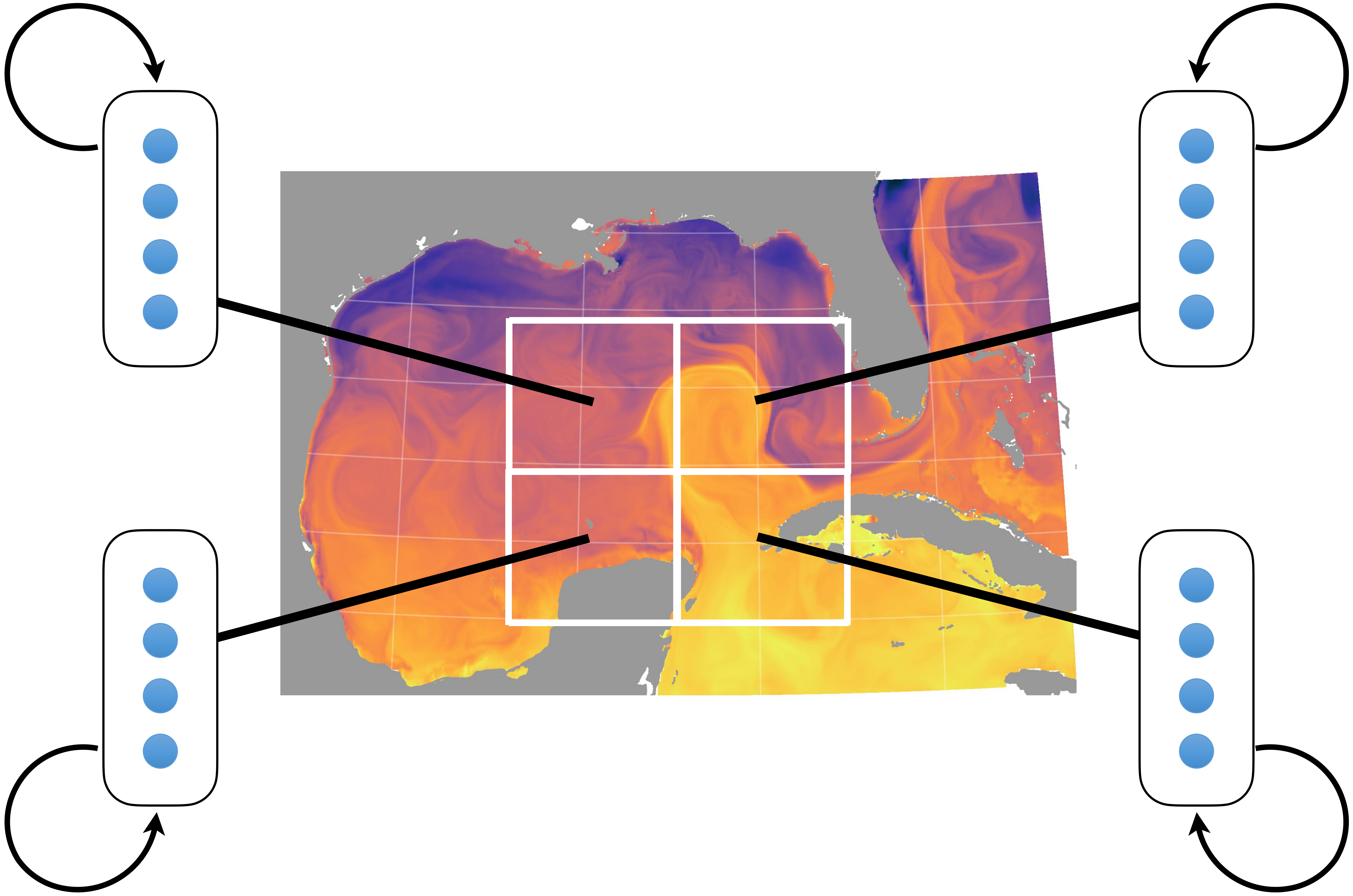
- Motivation: coupled DA presents extreme computational demands, balancing resolution, DA, and coupled model components

- RNNs discussed here show excellent skill in forecasting chaotic dynamics
  (Jason Platt et al., 2022; Griffith et al., 2012; Pathak et al., 2018)

- Successful:
  - Reproduction of Lyapunov spectrum
    (Pathak et al, 2017)
  - Error growth/covariance estimation
  - Integration with DA in Lorenz96
    (Stephen Penny et al., 2022)

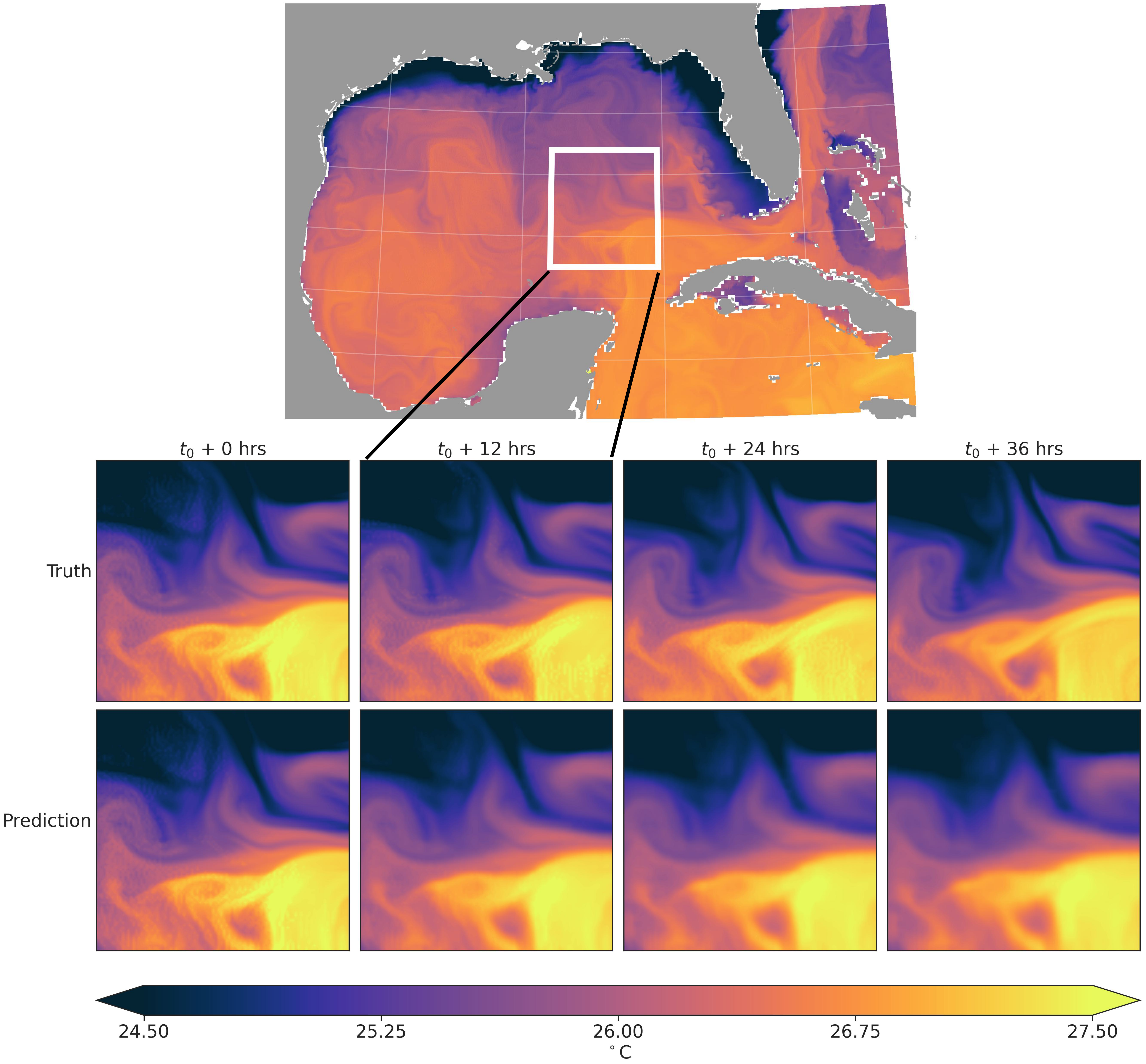


Error in LETKF with RNN surrogate models.
From (Stephen Penny et al., 2022)



RNNs in Geophysical Fluids

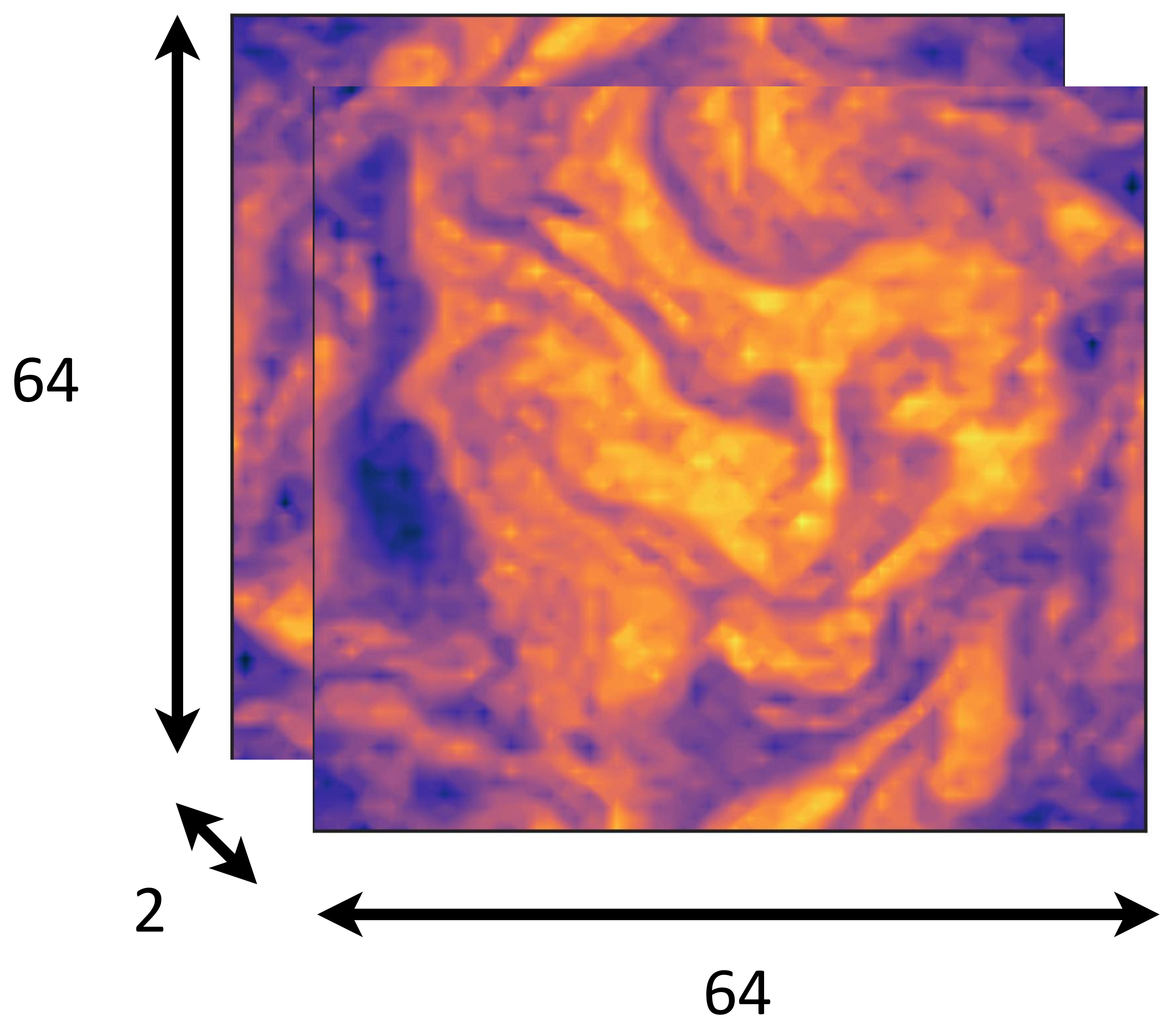
- The picture is less than rosy when used to emulate geophysical fluids, or processes dependent on them
- Here: emulate SST evolution in Gulf of Mexico, based on 1/25 degree reanalysis dataset
- What spatial scales can we expect to recover?
- What's causing smoothing?
  - Is it jumps due to DA?
  - Temporal subsampling (data available every 6hrs)?
  - Multivariate interactions not being captured?



RNNs in Geophysical Fluids

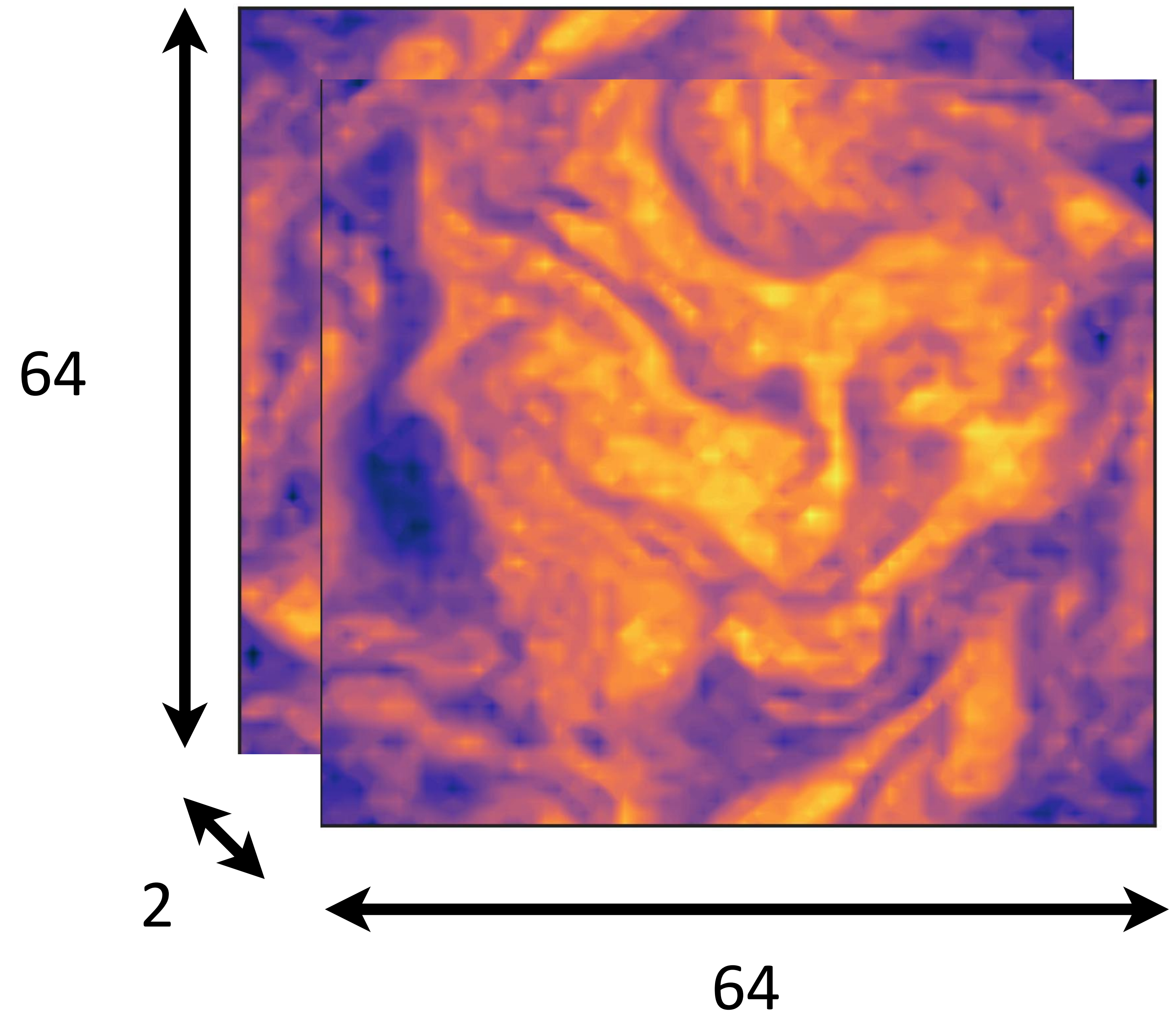
- The picture is less than rosy when used to emulate geophysical fluids, or processes dependent on them
- Here: emulate SST evolution in Gulf of Mexico, based on 1/25 degree reanalysis dataset
- What spatial scales can we expect to recover?
- What's causing smoothing?
  - Is it jumps due to DA?
  - Temporal subsampling (data available every 6hrs)?
  - Multivariate interactions not being captured?

Explore this in a more distilled environment: Surface Quasi-Geostrophic Turbulence



RNNs in Geophysical Fluids

- The picture is less than rosy when used to emulate geophysical fluids, or processes dependent on them
- Here: emulate SST evolution in Gulf of Mexico, based on 1/25 degree reanalysis dataset
- What spatial scales can we expect to recover?
- What's causing smoothing?
  - Is it jumps due to DA?
  - Temporal subsampling (data available every 6hrs)?
  - Multivariate interactions not being captured?



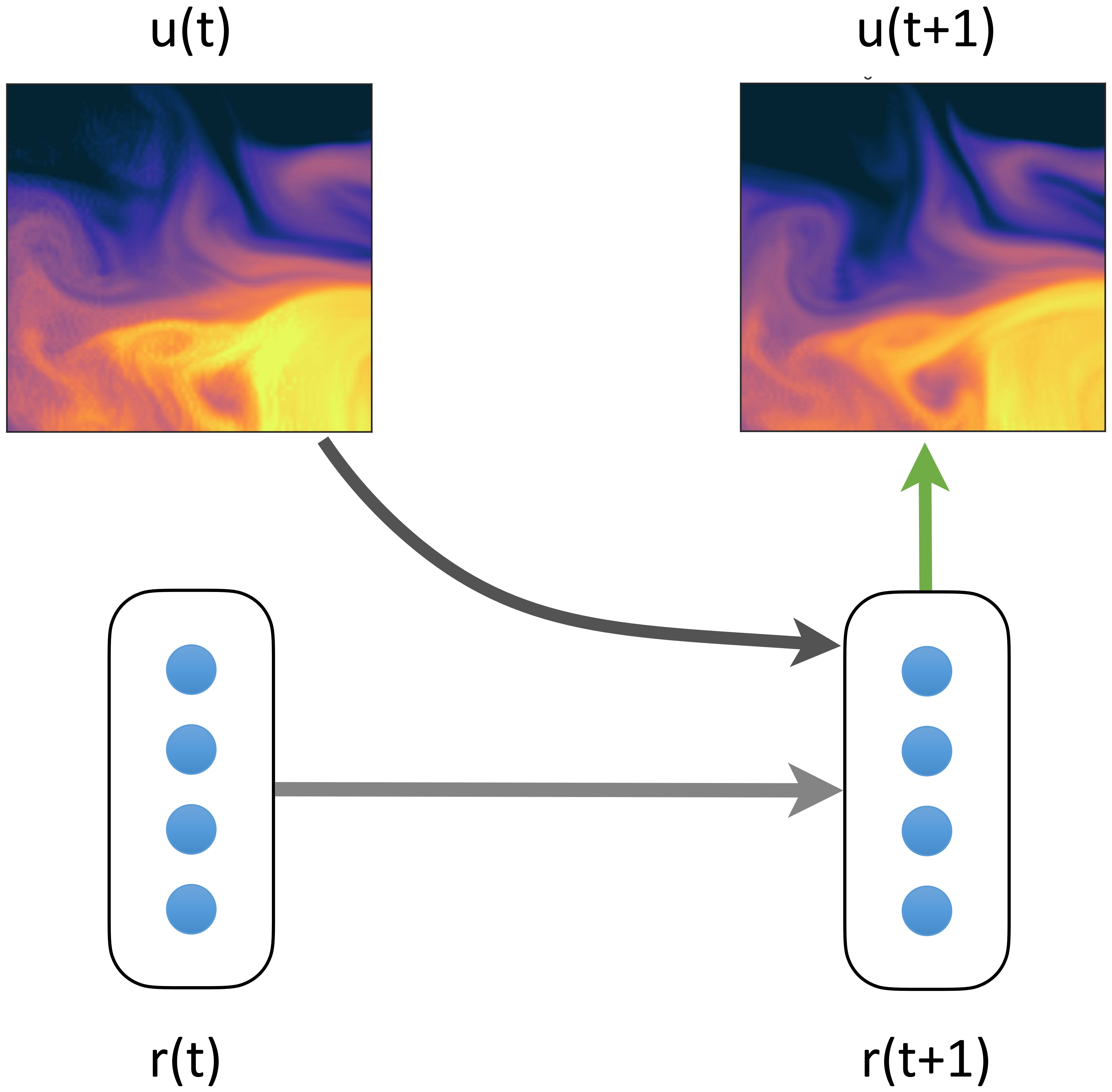
Explore this in a more distilled environment: Surface Quasi-Geostrophic Turbulence

Goals

- Understand how subsampling data impacts resolved scales
- Build intuition behind RNN

Which RNNs?

- Single hidden layer
- Only* train output layer, $\mathbf{W}_{out}$
- Results in fast, linear solve
- … *however, have to optimize/tune global parameters, θ



$$\mathbf{r}(t+1) = f(\mathbf{r}(t), \mathbf{u}(t); \theta)$$

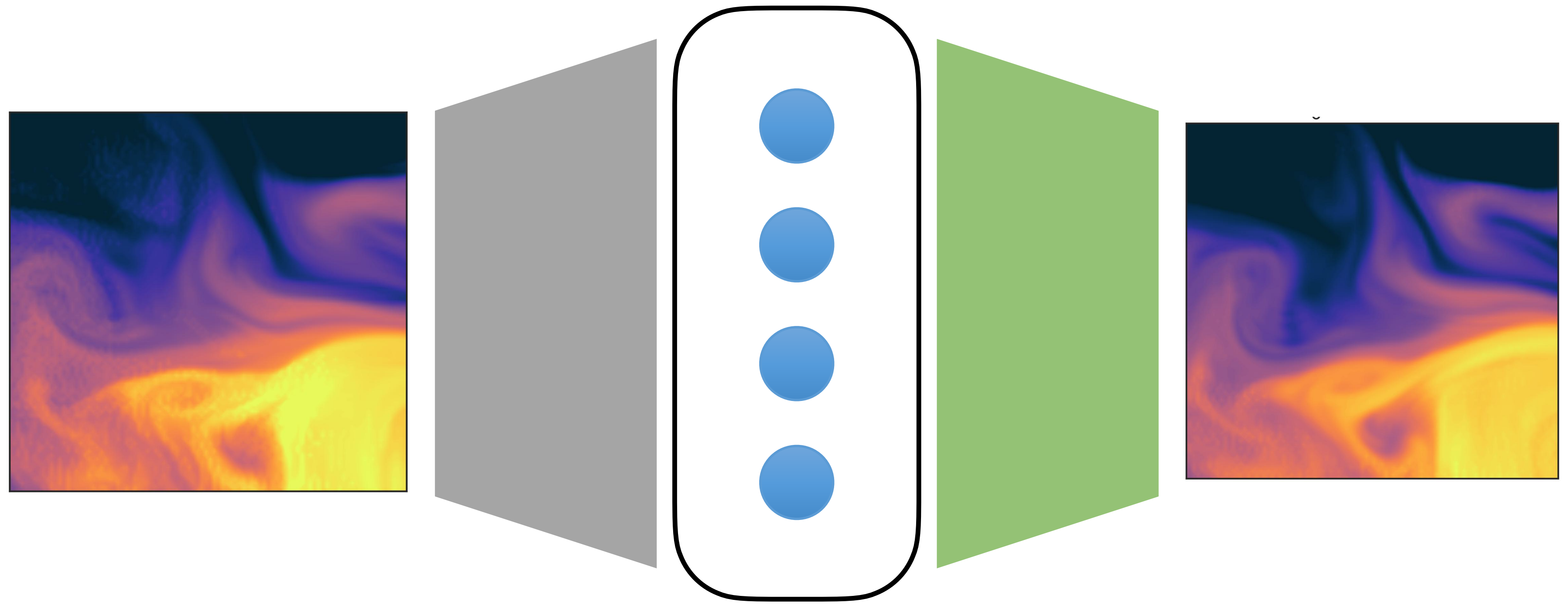
$$\mathbf{u}(t+1) = \mathbf{W}_{out}\mathbf{r}(t+1)$$

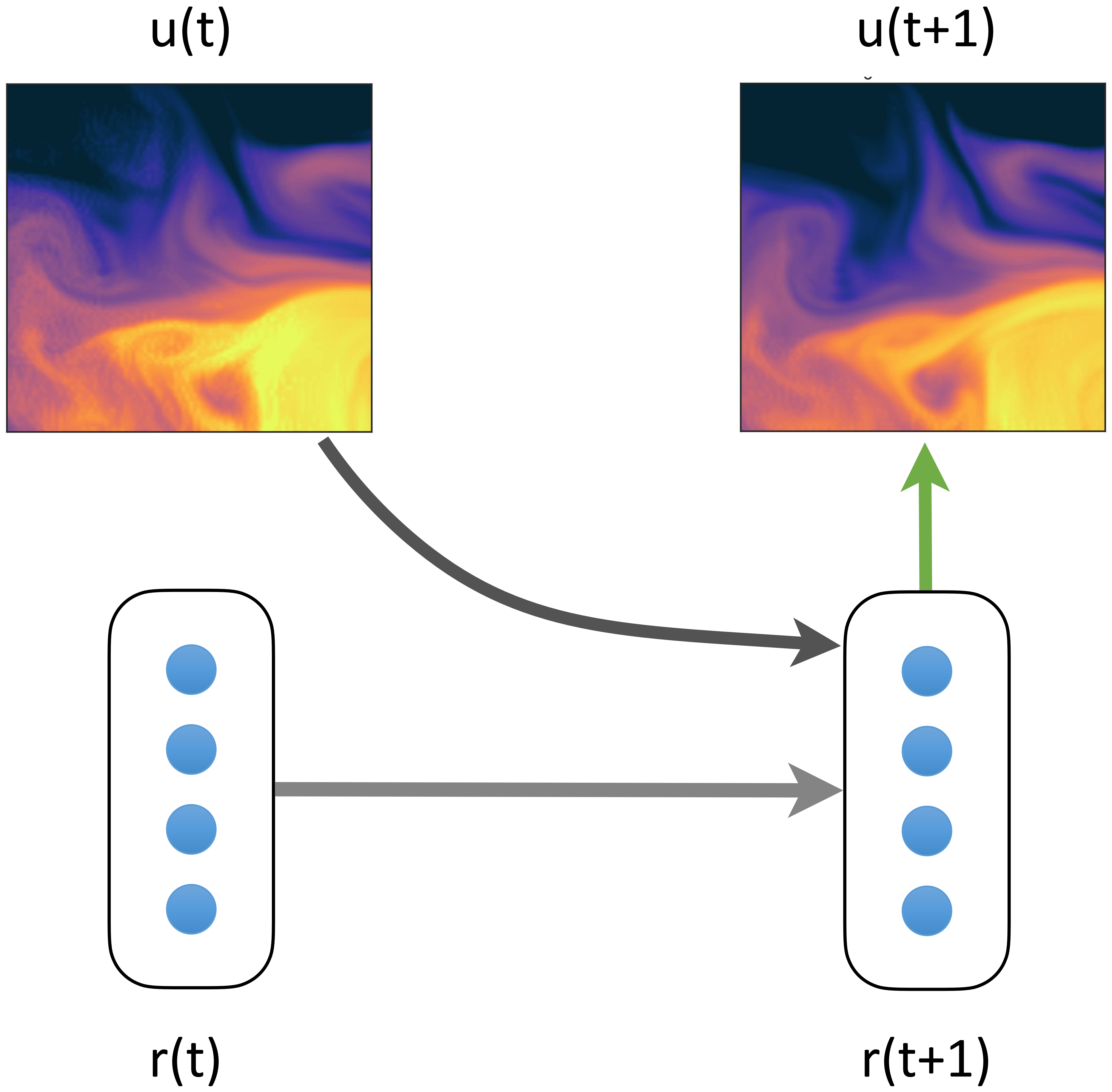
$$\mathcal{J}(\mathbf{W}_{\text{out}}) = \frac{1}{2}\sum_{t}^{\text{time}} \left|\left|\mathbf{W}_{\text{out}}\mathbf{r}_t - \mathbf{y}_t\right|\right|_2^2 + \frac{\beta^2}{2}\left|\left|\mathbf{W}_{\text{out}}\right|\right|_F^2$$

Which RNNs?

- Single hidden layer
- Only* train output layer, $\mathbf{W}_{out}$
- Results in fast, linear solve
- … *however, have to optimize/tune global parameters, θ

- Note: hidden space holds memory => usually higher dimensional than input!





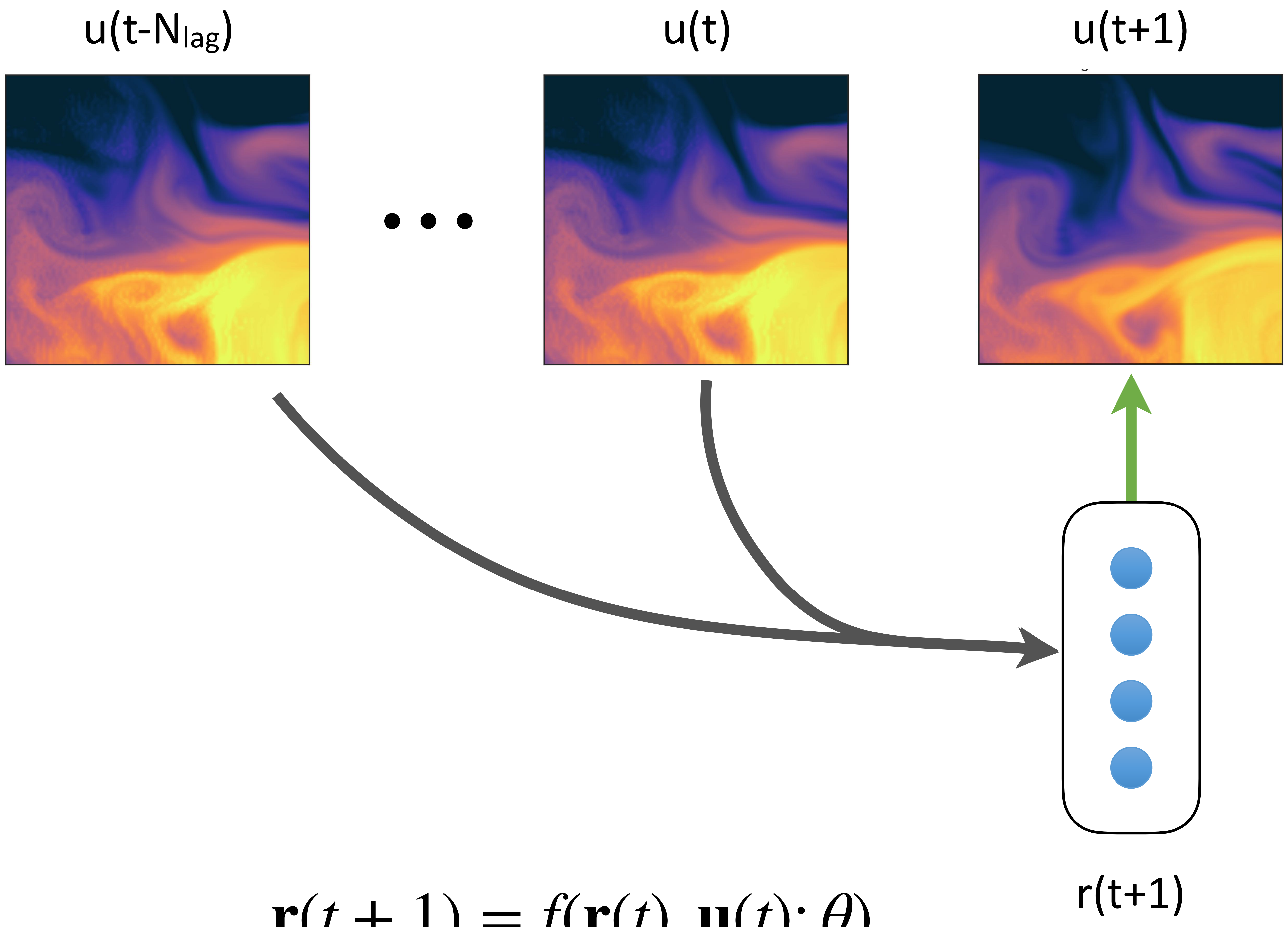
$$\mathbf{r}(t+1) = f(\mathbf{r}(t), \mathbf{u}(t); \theta)$$

$$\mathbf{u}(t+1) = \mathbf{W}_{out}\mathbf{r}(t+1)$$

$$\mathcal{J}(\mathbf{W}_{\text{out}}) = \frac{1}{2}\sum_{t}^{\text{time}} \left|\left| \mathbf{W}_{\text{out}}\mathbf{r}_t - \mathbf{y}_t \right|\right|_2^2 + \frac{\beta^2}{2}\left|\left| \mathbf{W}_{\text{out}} \right|\right|_F^2$$

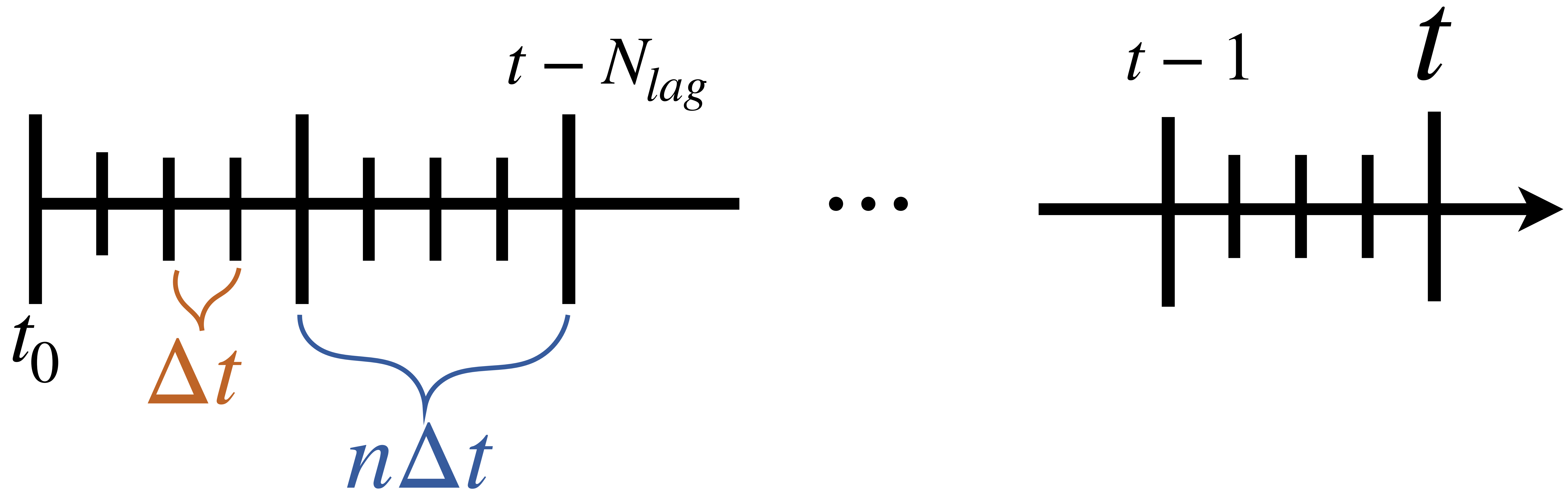
Nonlinear Vector Auto-Regression

- Hidden state filled directly from input state, we control:
  - interactions between system nodes or grid cells (e.g. polynomial)
  - memory via number of time lagged states to include



$$\mathbf{r}(t+1) = f(\mathbf{r}(t), \mathbf{u}(t); \theta)$$

$$\mathbf{r}(t+1) = [u_1^2 \; u_2^2 \; u_1 u_2 \; u_1 \; u_2 \; 1]$$

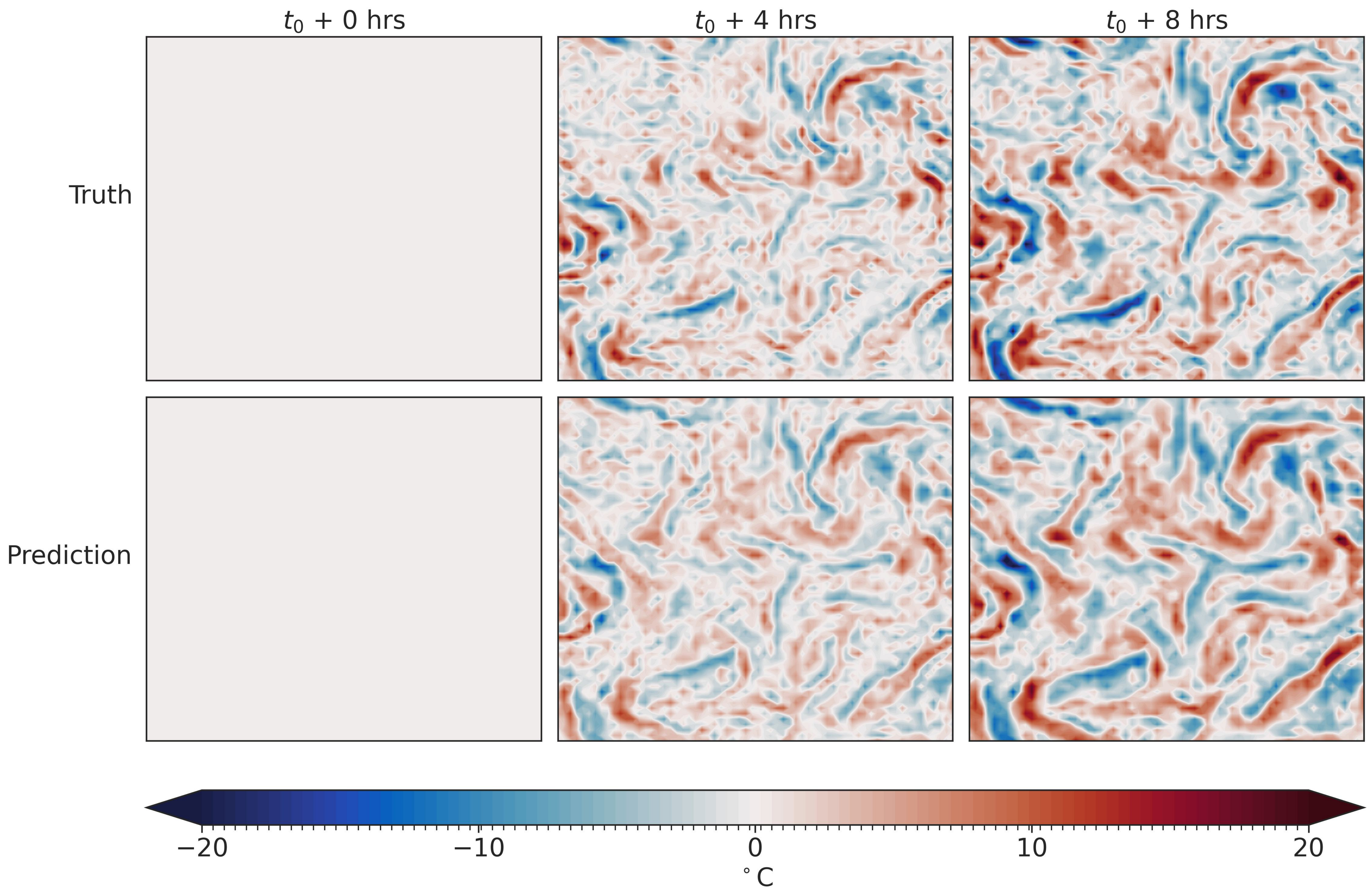


$\Delta t = 5$ min

n = {1, 2, 4, 8, 16}

Subsampling: n=16

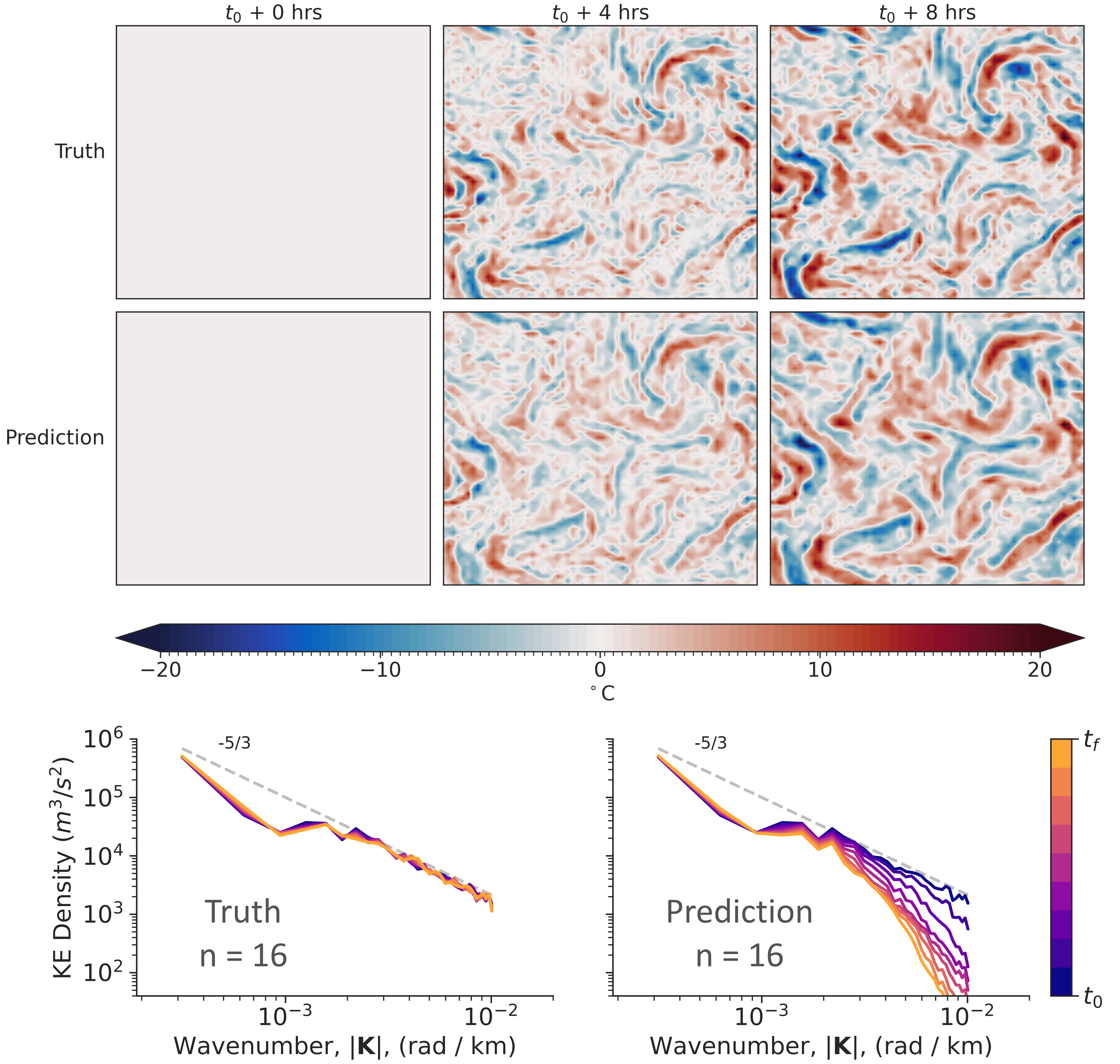
- See similar picture to GoM results: qualitative resemblance, but overly smooth
- Kinetic energy density spectrum: as time progresses, small scale features are damped



Subsampling: n=16

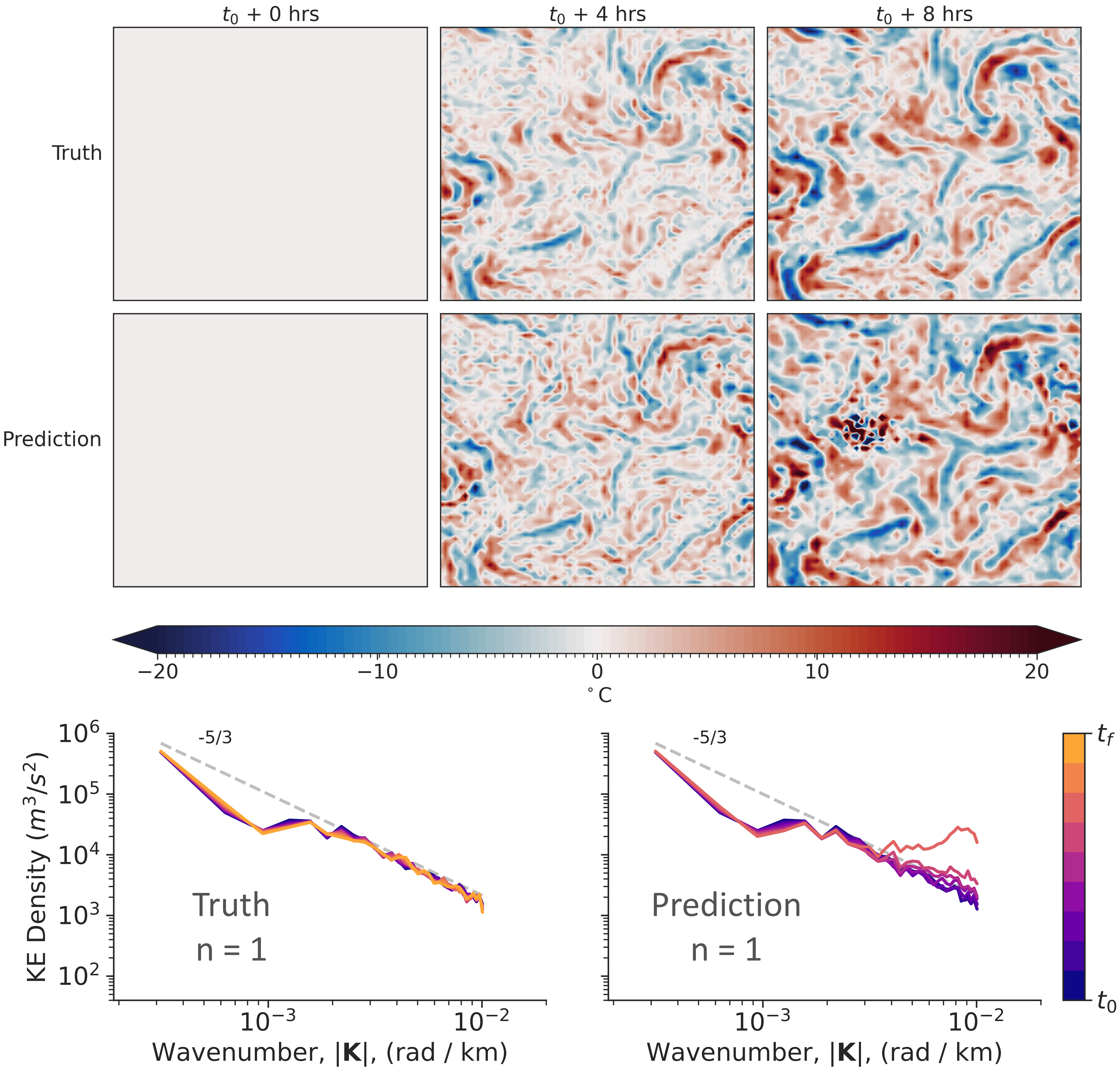
- See similar picture to GoM results: qualitative resemblance, but overly smooth

- Kinetic energy density spectrum: as time progresses, small scale features are damped

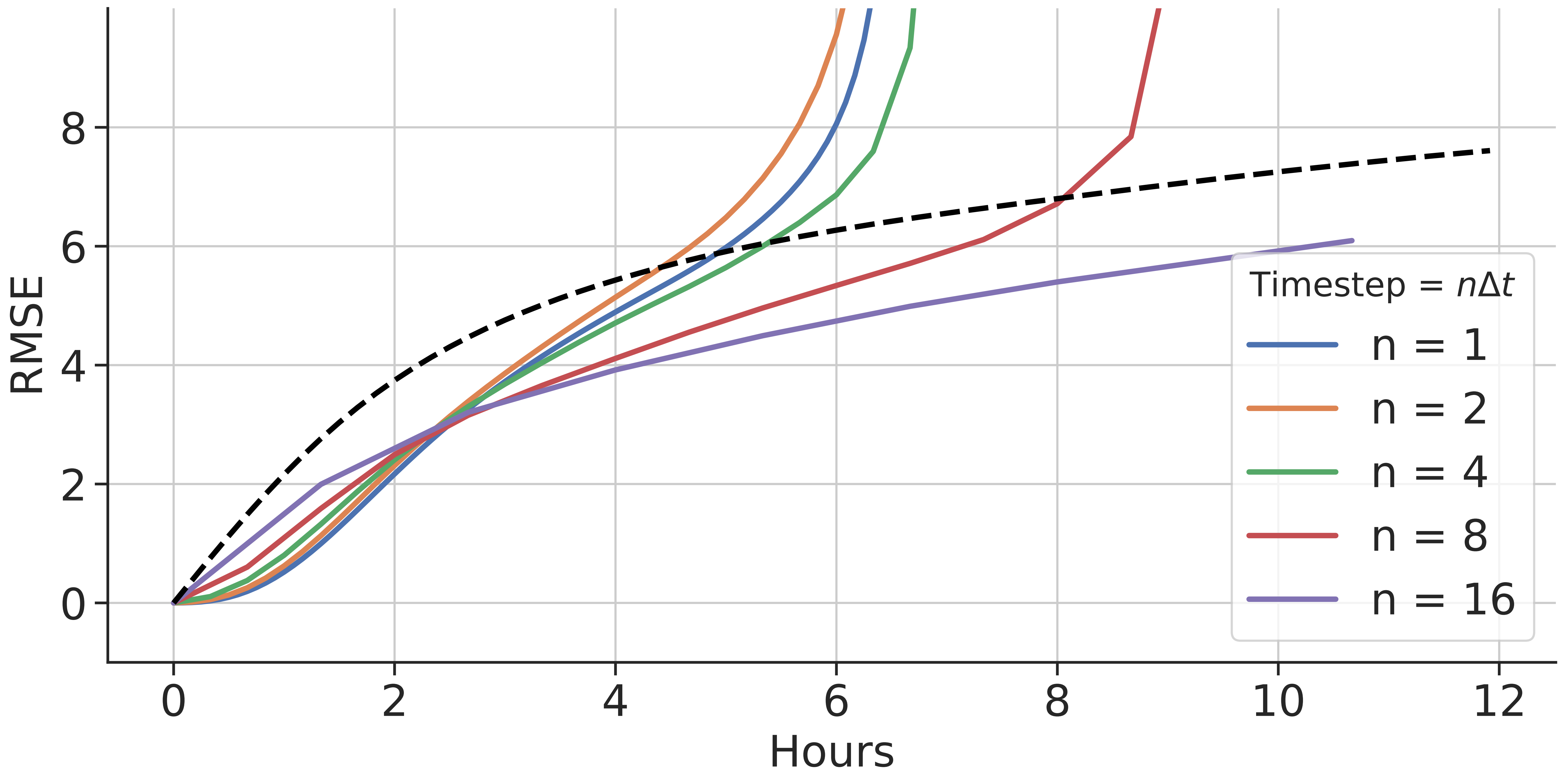


No subsampling: n=1

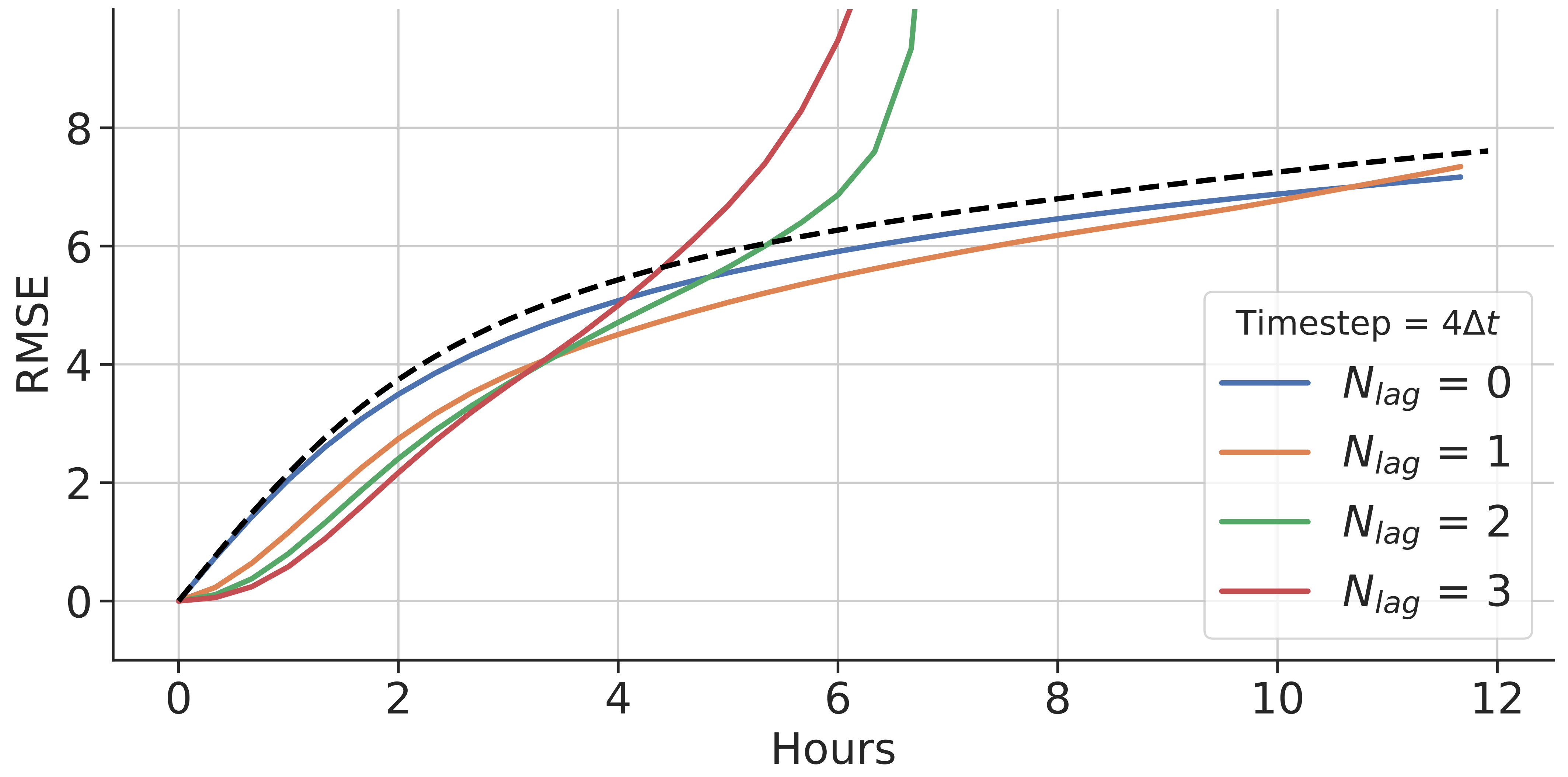
- Good skill at first
- Eventually, small scale artificial instabilities are generated
- Spectrum shows larger amplitude than expected at small scales



Tradeoff: Smooth & Stable -vs- Early Skill & Blowup



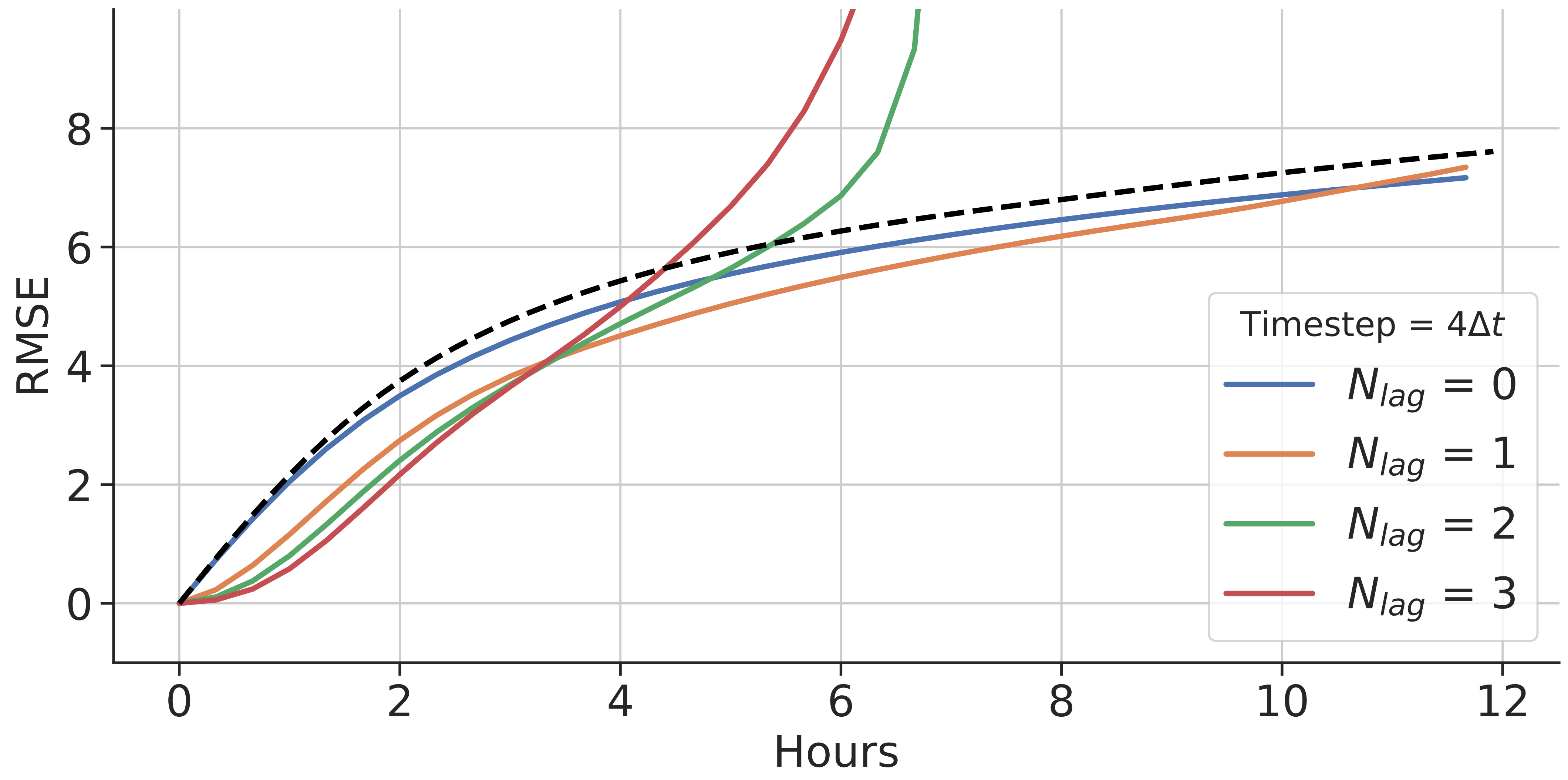
Memory ~ Sampling



Memory ~ Sampling

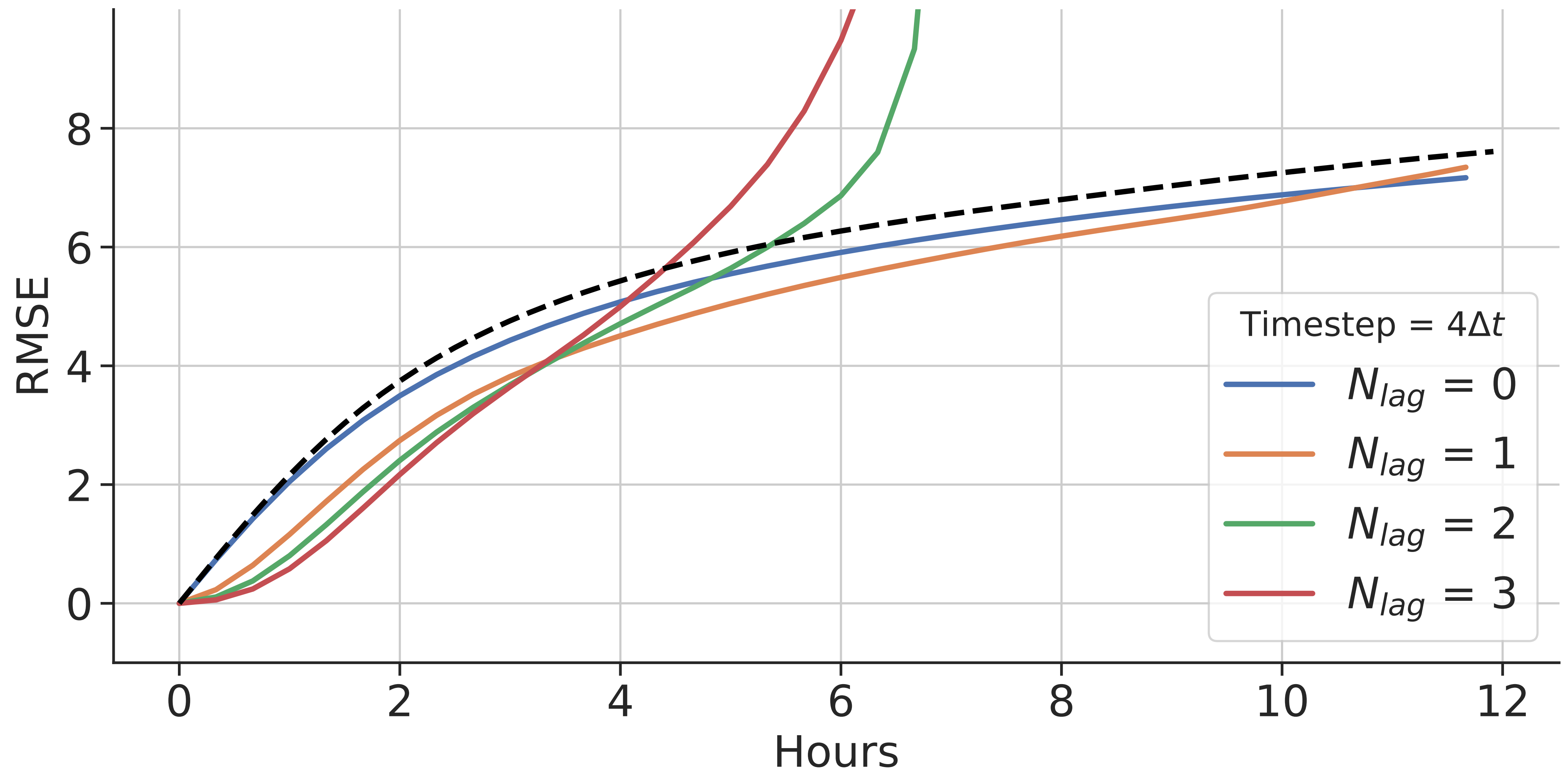
- Up to a point, subsampled output can recover skill of predictions with no subsampling

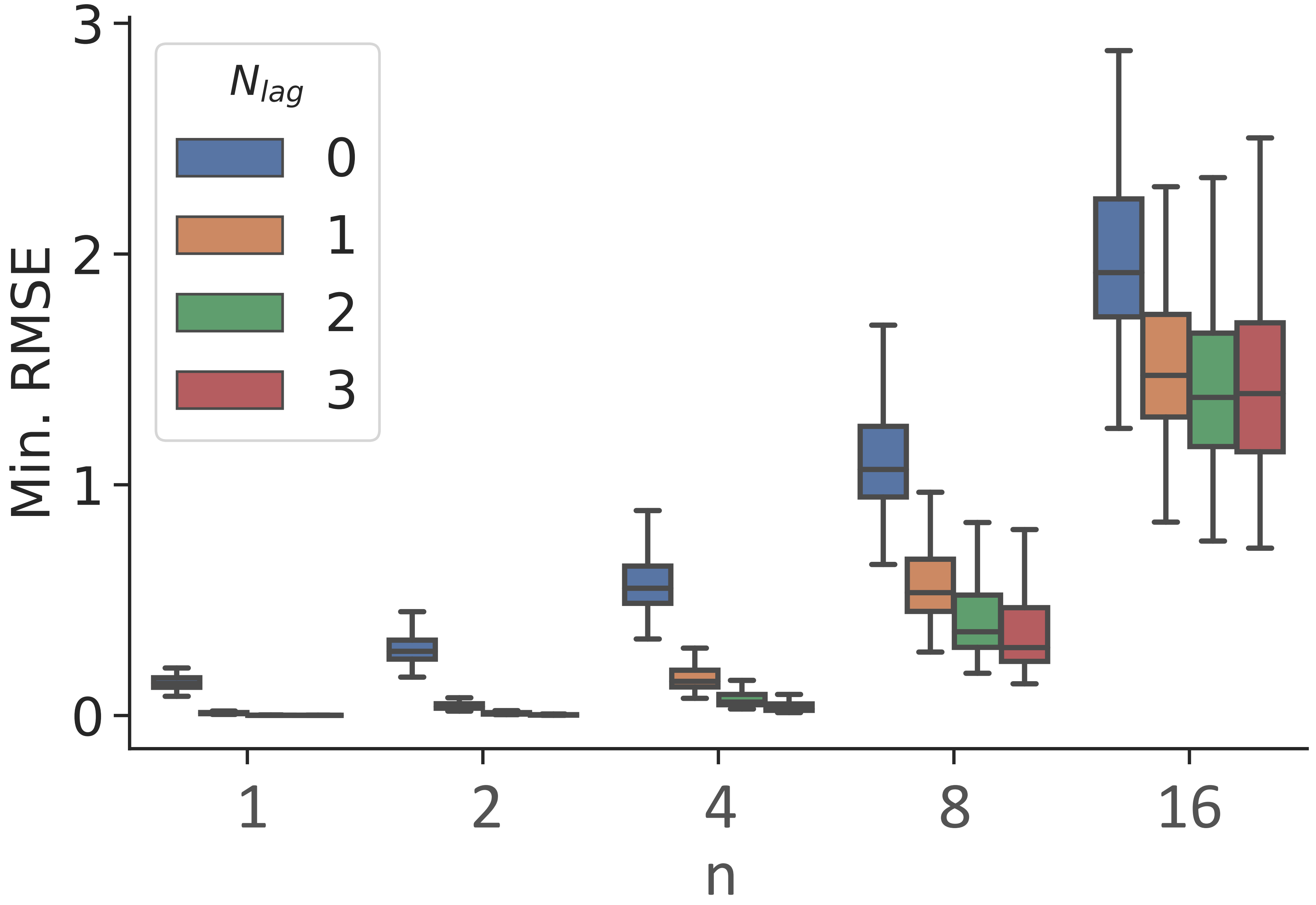
- But, simple quadratic polynomials do not contain nonlinearity to handle these time lagged states



Memory ~ Sampling

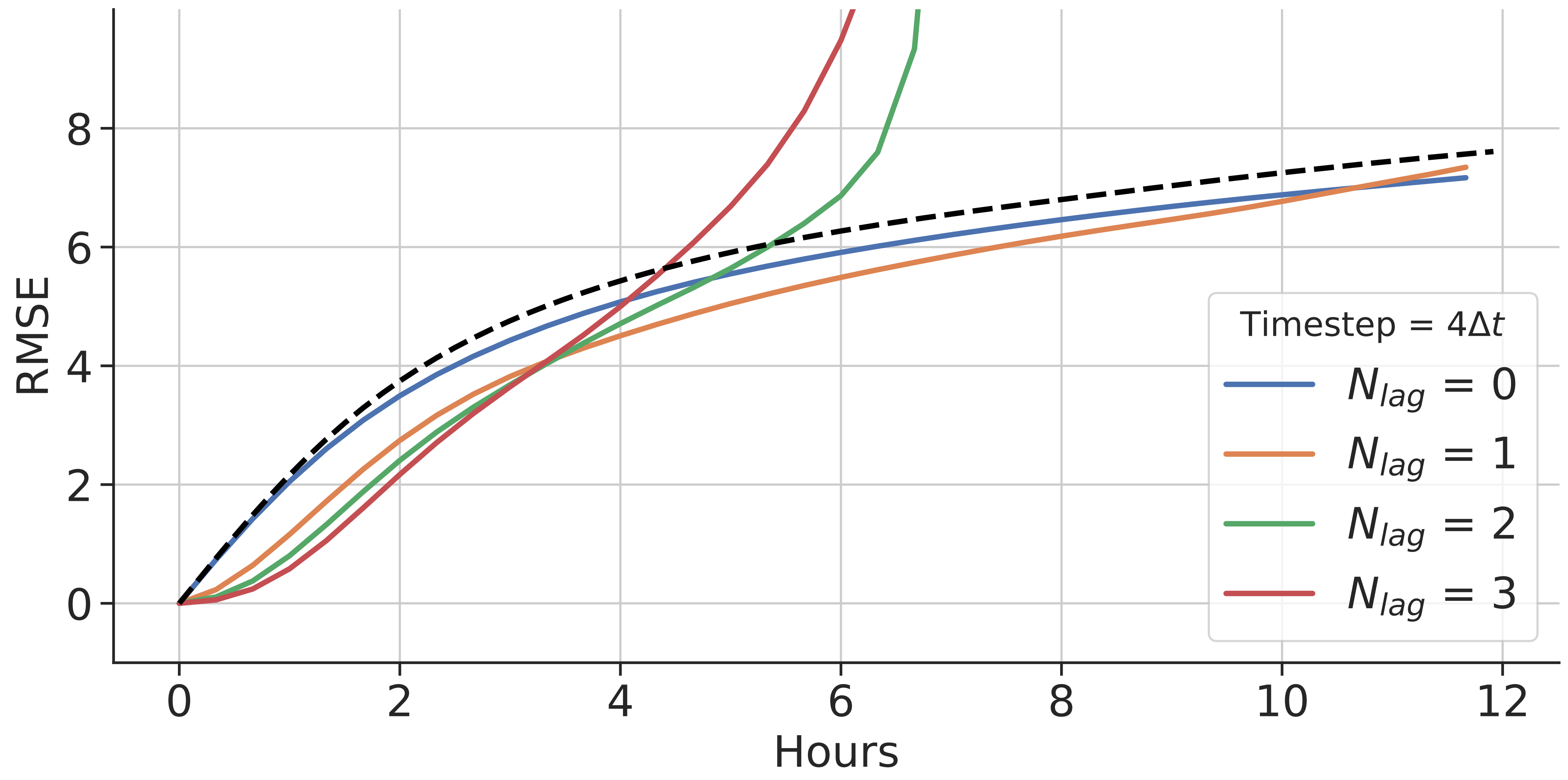
- Up to a point, subsampled output can recover skill of predictions with no subsampling
- But, simple quadratic polynomials do not contain nonlinearity to handle these time lagged states

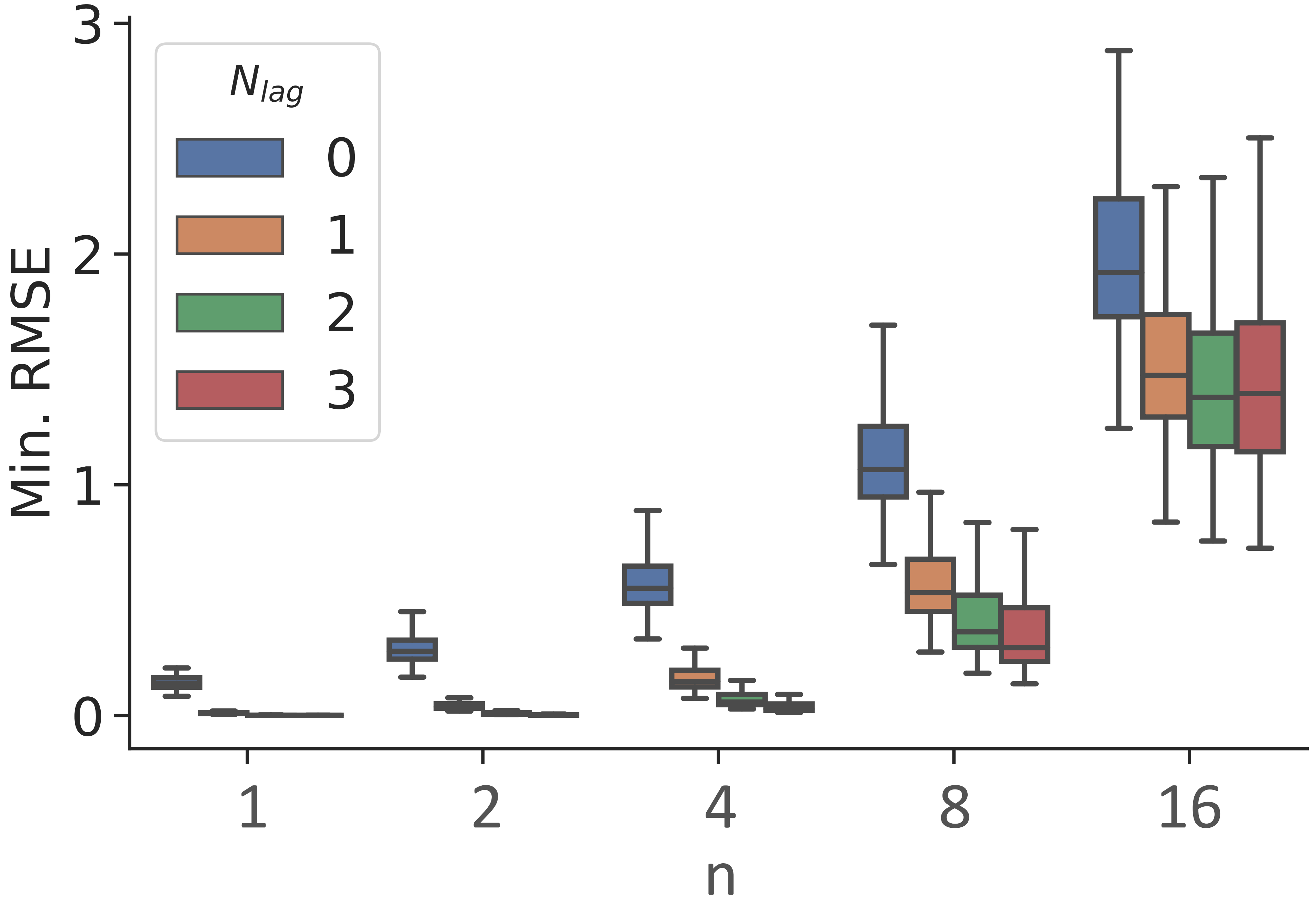




Memory ~ Sampling

- Up to a point, subsampled output can recover skill of predictions with no subsampling
- But, simple quadratic polynomials do not contain nonlinearity to handle these time lagged states
- Inaccuracy or uncertainty in nonlinearity is detrimental to NVAR (Zhang & Cornelius, 2022)





How to get decent early prediction skill with stability?

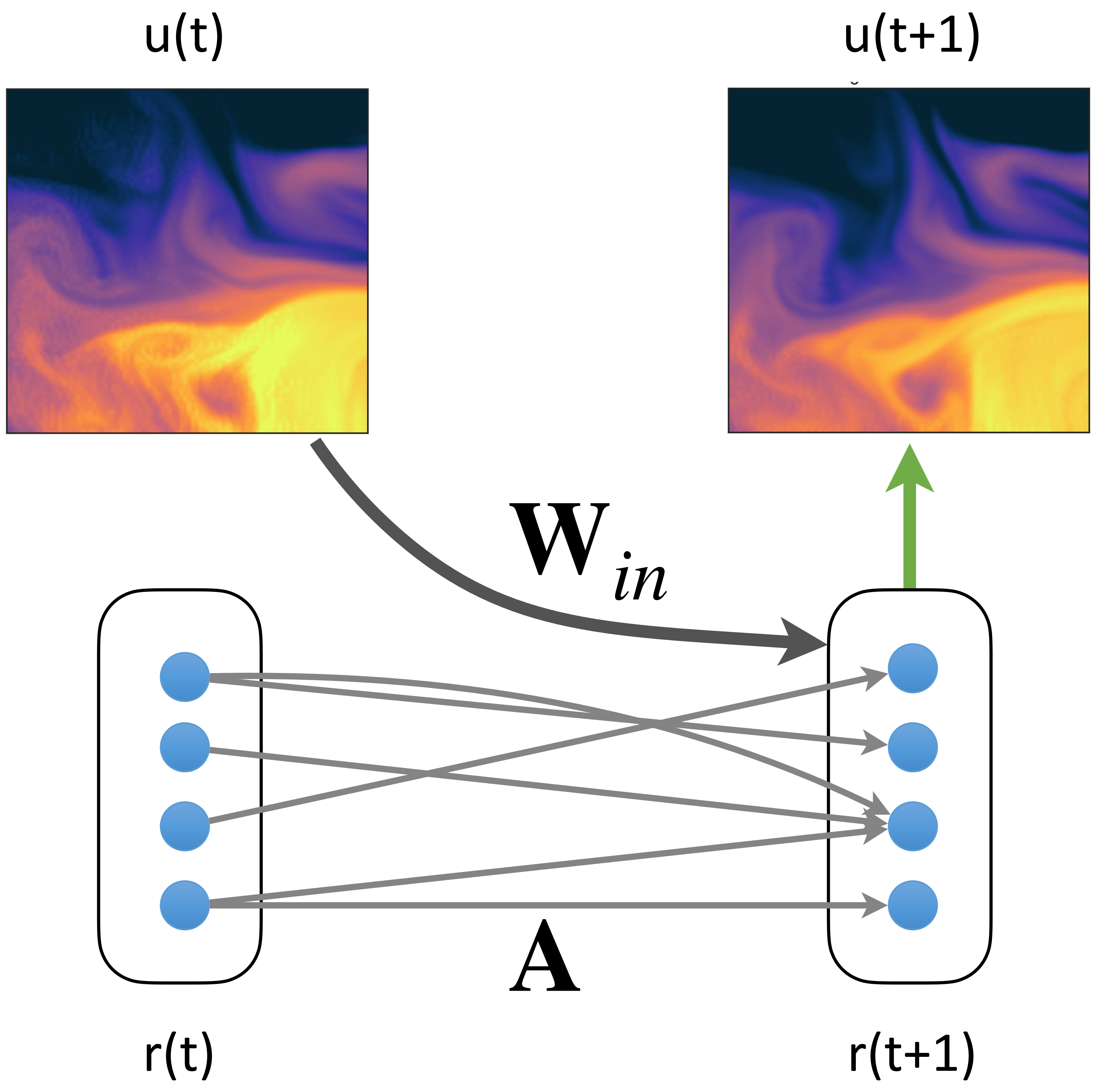
Reservoir Computing

Generate randomly:

- $\hat{\mathbf{A}} \sim \mathscr{U}[-1,1]$, sparse
  then re-scale by desired spectral radius
- $\mathbf{W}_{in} \sim \mathscr{U}[-\sigma, \sigma]$, dense

Connection to what we've seen:

- RC is a generalization of NVAR (Bollt, 2021)
- Well-tuned NVAR reproduces time stepping stencil (Tse-Chun Chen et al., 2022)

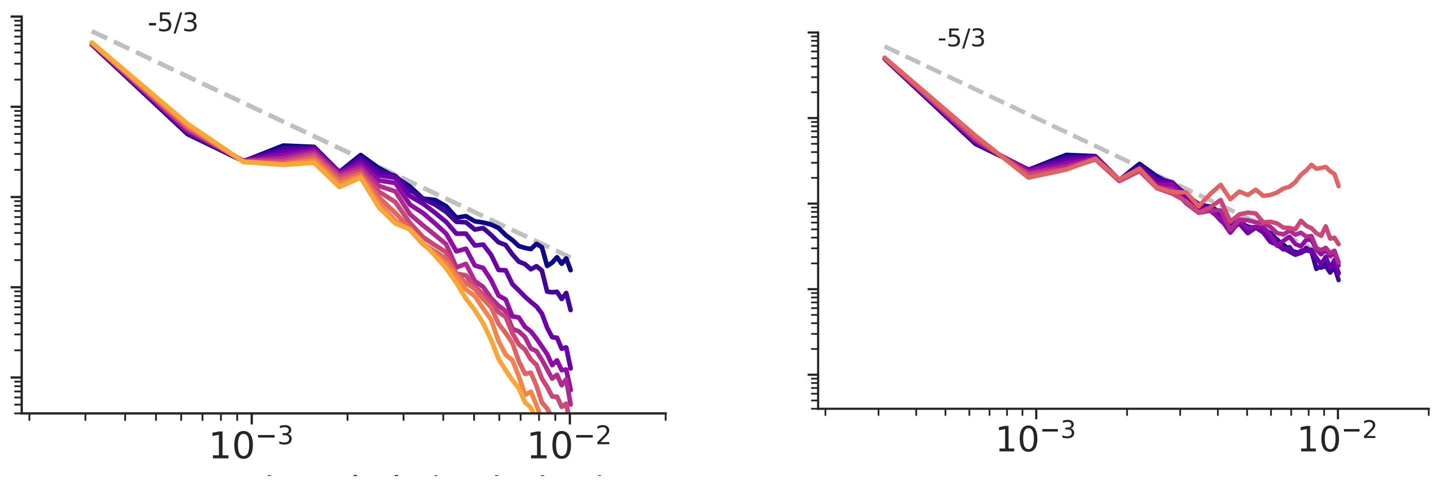


$$\mathbf{r}(t+1) = f(\mathbf{r}(t), \mathbf{u}(t); \theta)$$

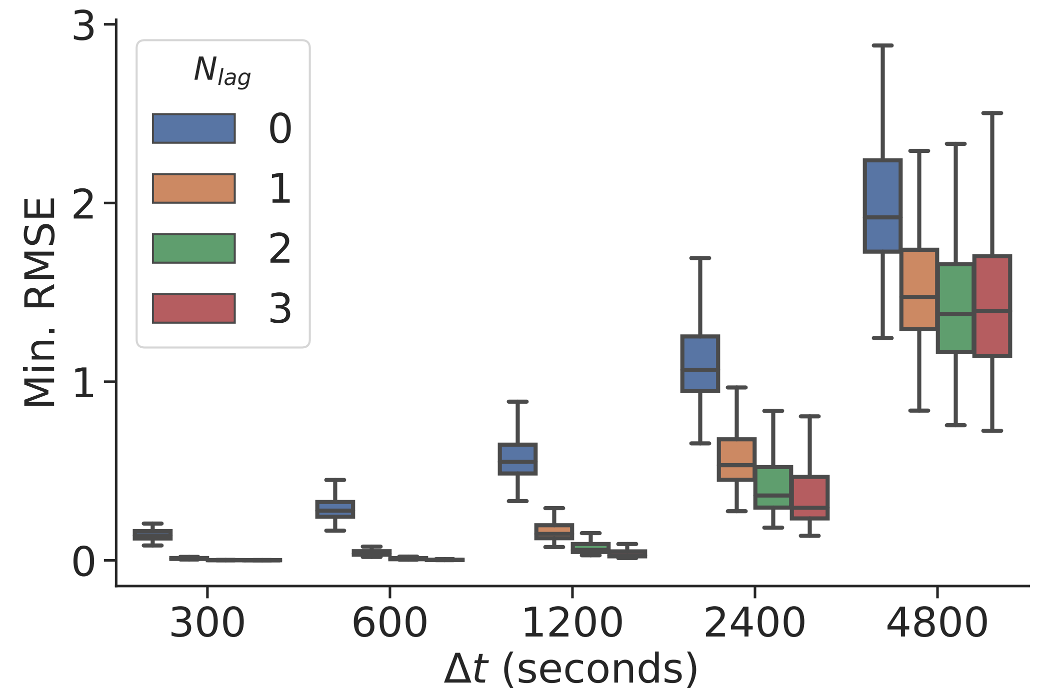
$$\mathbf{r}(t+1) = \tanh\left(\mathbf{A}\mathbf{r}(t) + \mathbf{W}_{in}\mathbf{u}(t) + \mathbf{b}\right)$$

Summary and conclusions

- Temporal sampling naturally lends itself to smoothing or instability, something to consider when training on reanalysis datasets



- NVAR provides intuition behind hidden state, memory effects, but suffers from heavy-handed architecture and scaling issues



- RC shows promise, but strongly dependent on optimization of hyperparameters

- Outlook: conservation laws or information to constrain this process? Or do we really need to train those individual matrix weights…

References

Bollt, E. (2021). On explaining the surprising success of reservoir computing forecaster of chaos? The universal machine learning dynamical system with contrast to VAR and DMD. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *31*(1), 013108. https://doi.org/10.1063/5.0024890

Chen, T.-C., Penny, S. G., Smith, T. A., and Platt, J. A. 'Next Generation' Reservoir Computing: an Empirical Data-Driven Expression of Dynamical Equations in Time-Stepping Form. Preprint: https://arxiv.org/abs/2201.05193.

Griffith, A., Pomerance, A., & Gauthier, D. J. (2019). Forecasting chaotic systems with very low connectivity reservoir computers. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *29*(12), 123108. https://doi.org/10.1063/1.5120710

Pathak, J., Lu, Z., Hunt, B. R., Girvan, M., & Ott, E. (2017). Using machine learning to replicate chaotic attractors and calculate Lyapunov exponents from data. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, *27*(12), 121102. https://doi.org/10.1063/1.5010300

Pathak, J., Hunt, B., Girvan, M., Lu, Z., & Ott, E. (2018). Model-Free Prediction of Large Spatiotemporally Chaotic Systems from Data: A Reservoir Computing Approach. *Physical Review Letters*, *120*(2), 024102. https://doi.org/10.1103/PhysRevLett.120.024102

Penny, S. G., Smith, T. A., Chen, T.-C., Platt, J. A., Lin, H.-Y., Goodliff, M., & Abarbanel, H. D. I. (2022). Integrating Recurrent Neural Networks With Data Assimilation for Scalable Data-Driven State Estimation. *Journal of Advances in Modeling Earth Systems*, *14*(3), e2021MS002843. https://doi.org/10.1029/2021MS002843

Platt, J. A., Penny, S. G., Smith, T. A., Chen, T.-C., & Abarbanel, H. D. I. (2022). A systematic exploration of reservoir computing for forecasting complex spatiotemporal dynamics. *Neural Networks*, *153*, 530–552. https://doi.org/10.1016/j.neunet.2022.06.025

Zhang, Y., & Cornelius, S. P. (2022). A Catch-22 of Reservoir Computing. arXiv. Retrieved from http://arxiv.org/abs/2210.10211