Score-based Data Assimilation

François Rozet and Gilles Louppe

LIEGE université

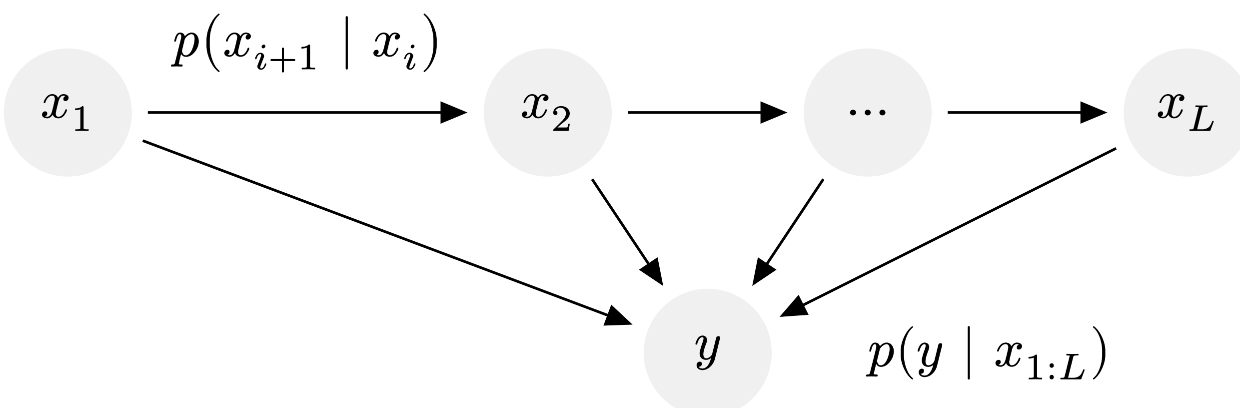
Problem statement

Data assimilation

Data assimilation (DA) addresses the problem of inferring the posterior distribution

$$p(x_{1:L} \mid y) = \frac{p(y \mid x_{1:L})}{p(y)} \underbrace{p(x_1) \prod_{i=1}^{L-1} p(x_{i+1} \mid x_i)}_{\text{Markovian prior}}$$

for dynamical systems (atmospheres, oceans, ...) given noisy or incomplete observations.



How to use **score-based generative modeling** to approximate the posterior $p(x_{1:L} \mid y)$?

How to exploit the **Markovian structure** of $x_{1:L}$?

Score-based generative modeling

1. Data samples $x \sim p(x)$ are continuously (from $t = 0$ to 1) transformed into noise through a stochastic **diffusion process**

$$\mathrm{d}x(t) = f(t)x(t)\,\mathrm{d}t + g(t)\,\mathrm{d}w(t)$$

   such that $p(x(0)) \approx p(x)$ and $p(x(1)) \approx \mathcal{N}(0, I)$ and

$$p(x(t) \mid x) = \mathcal{N}\left(x(t) \mid \mu(t)x, \sigma(t)^2 I\right)$$

2. The **reverse process**

$$\mathrm{d}x(t) = \left[f(t)x(t) - g(t)^2 \nabla_{x(t)} \log p(x(t))\right] \mathrm{d}t + g(t)\,\mathrm{d}w(t)$$

   can be simulated (from $t = 1$ to 0) to generate new data from $p(x(0))$.

3. The **score function** $\nabla_{x(t)} \log p(x(t))$ is approximated with a score network $s_\phi(x(t), t)$ trained to solve

$$\arg\min_\phi \mathbb{E}_{p(x)p(t)p(x(t) \mid x)}\left[\sigma(t)^2 \left\| s_\phi(x(t), t) - \nabla_{x(t)} \log p(x(t) \mid x) \right\|^2\right]$$

To generate trajectories from $p(x_{1:L} \mid y)$, we have to replace $\nabla_{x(t)} \log p(x(t))$ with the **posterior score**

$$\nabla_{x_{1:L}(t)} \log p(x_{1:L}(t) \mid y) = \underbrace{\nabla_{x_{1:L}(t)} \log p(x_{1:L}(t))}_{\text{prior score}} + \underbrace{\nabla_{x_{1:L}(t)} \log p(y \mid x_{1:L}(t))}_{\text{likelihood score}}$$

in the **reverse process**.

Methods & contributions

How is your blanket?

Let x_{b_i} denote a **Markov blanket** of x_i within a set $x_{1:L}$ such that

$$p\left(x_i \mid x_{\neq i}\right) = p\left(x_i \mid x_{b_i}\right)$$

Consequently,

$$\nabla_{x_i} \log p(x_{1:L}) = \nabla_{x_i} \log p\left(x_i, x_{b_i}\right)$$

Not true for $x_{1:L}(t)$, but there exists $\bar{b}_i \supseteq b_i$ such that

$$\nabla_{x_i(t)} \log p(x_{1:L}(t)) \approx \nabla_{x_i(t)} \log p\left(x_i(t), x_{\bar{b}_i}(t)\right)$$

meaning that each element of the **prior score** can be determined **locally**.

For a first-order Markov chain, $x_{b_i} = \{x_{i-1}, x_{i+1}\}$ and

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

for $k \geq 1$ but $k \ll L$.

How is your blanket?

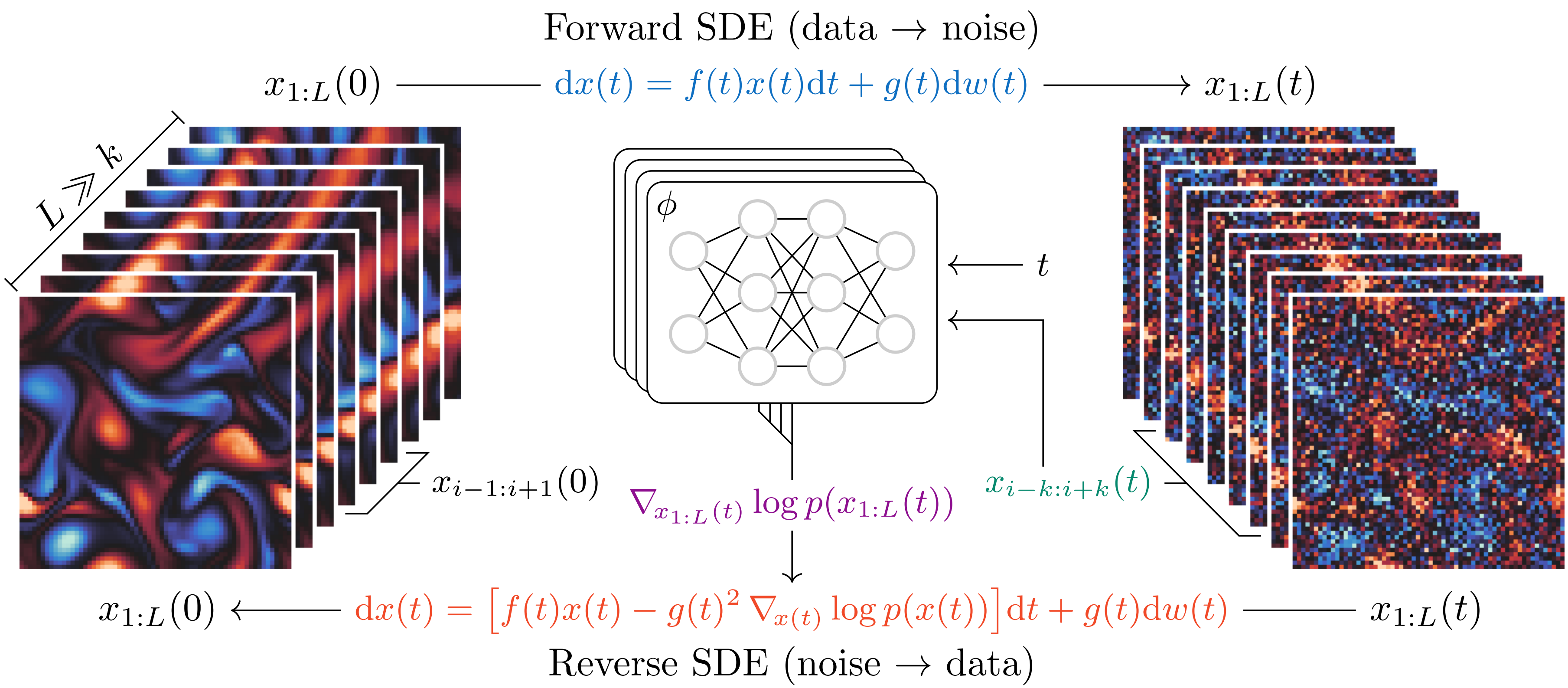


Figure 1. We compose the outputs of a score network $s_\phi(x_{i-k:i+k}(t), t)$ trained over **short segments** $x_{i-k:i+k}$ to approximate the **prior score**.

Stable likelihood score

Assuming a Gaussian observation process $p(y \mid x) = \mathcal{N}\big(y \mid \mathcal{A}(x), \Sigma_y\big)$, Chung et al. (2023) propose the approximation

$$p(y \mid x(t)) \approx \mathcal{N}\big(y \mid \mathcal{A}(\hat{x}(x(t))), \Sigma_y\big)$$

where Tweedie's formula gives

$$\hat{x}(x(t)) = \mathbb{E}[x \mid x(t)] \approx \frac{x(t) + \sigma(t)^2 s_\phi(x(t), t)}{\mu(t)}$$

which allows to estimate the **likelihood score** in **zero-shot**.

We introduce a **more accurate and more stable** approximation

$$p(y \mid x(t)) \approx \mathcal{N}\left(y \mid \mathcal{A}(\hat{x}(x(t))), \Sigma_y + \frac{\sigma(t)^2}{\mu(t)^2} A \Gamma A^T\right)$$

where Γ depends on the eigendecomposition of Σ_x and $A = \frac{\partial \mathcal{A}}{\partial x}\big|_{\hat{x}(x(t))}$ is the Jacobian of $\mathcal{A}$.

Predictor-Corrector sampling

To simulate the **reverse process** we adopt the exponential integrator (EI) discretization scheme introduced by Zhang et al. (2023)

$$x(t - \Delta t) \leftarrow \frac{\mu(t - \Delta t)}{\mu(t)} x(t) + \left(\frac{\mu(t - \Delta t)}{\mu(t)} - \frac{\sigma(t - \Delta t)}{\sigma(t)} \right) \sigma(t)^2 s_\phi(x(t), t)$$

To prevent errors from accumulating along the simulation, we perform C Langevin Monte Carlo corrections

$$x(t) \leftarrow x(t) + \delta s_\phi(x(t), t) + \sqrt{2\delta}\varepsilon$$

between each step of the discretized **reverse process**.

Results

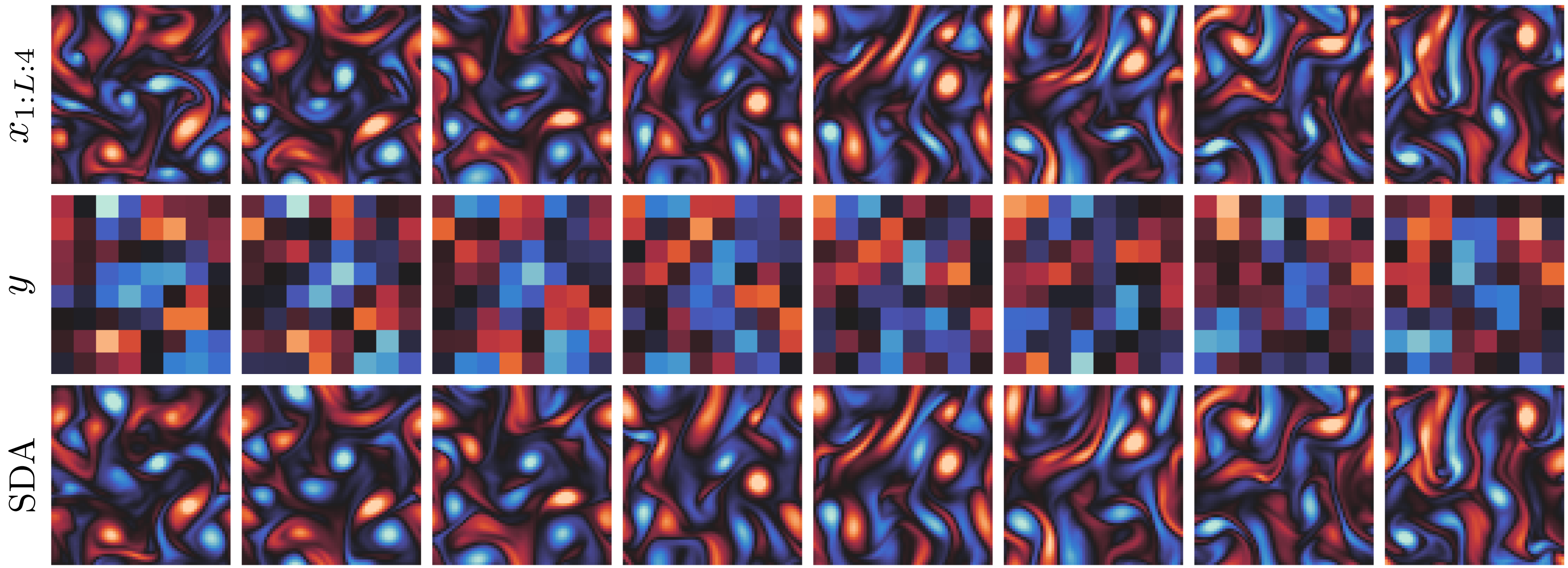


Figure 2. SDA works for challenging high-dimensional problems.

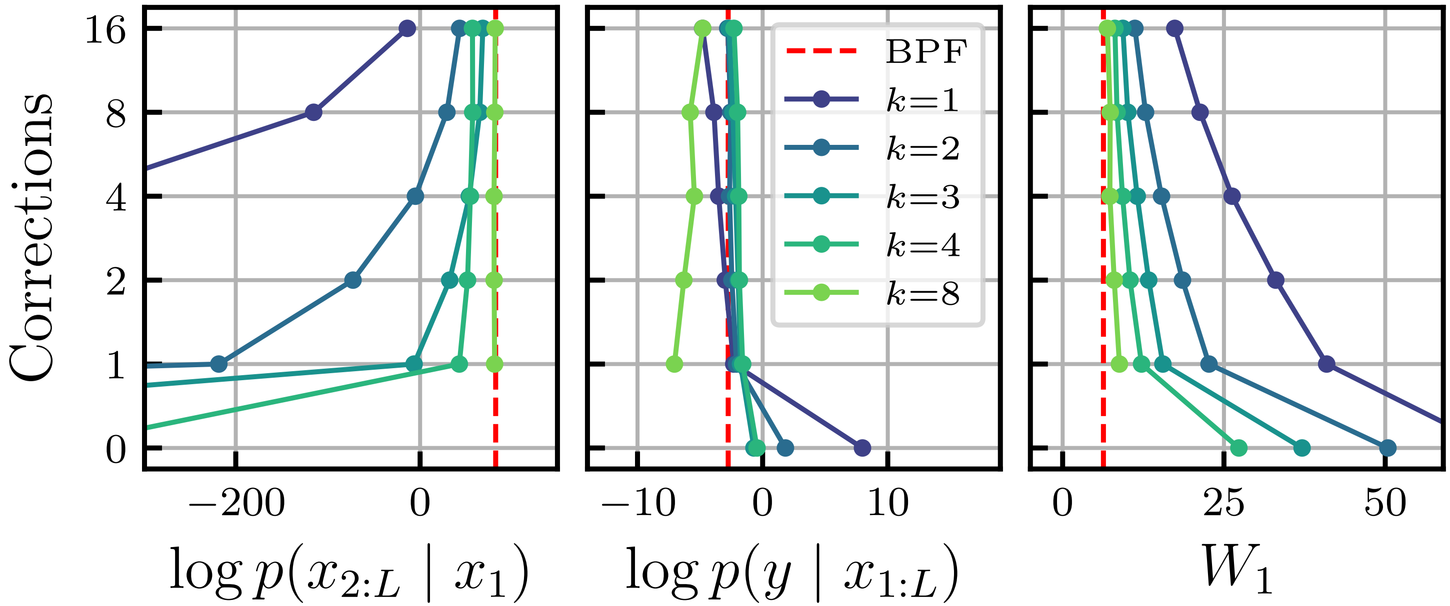


Figure 3. SDA converges to the true posterior as k and C increase.

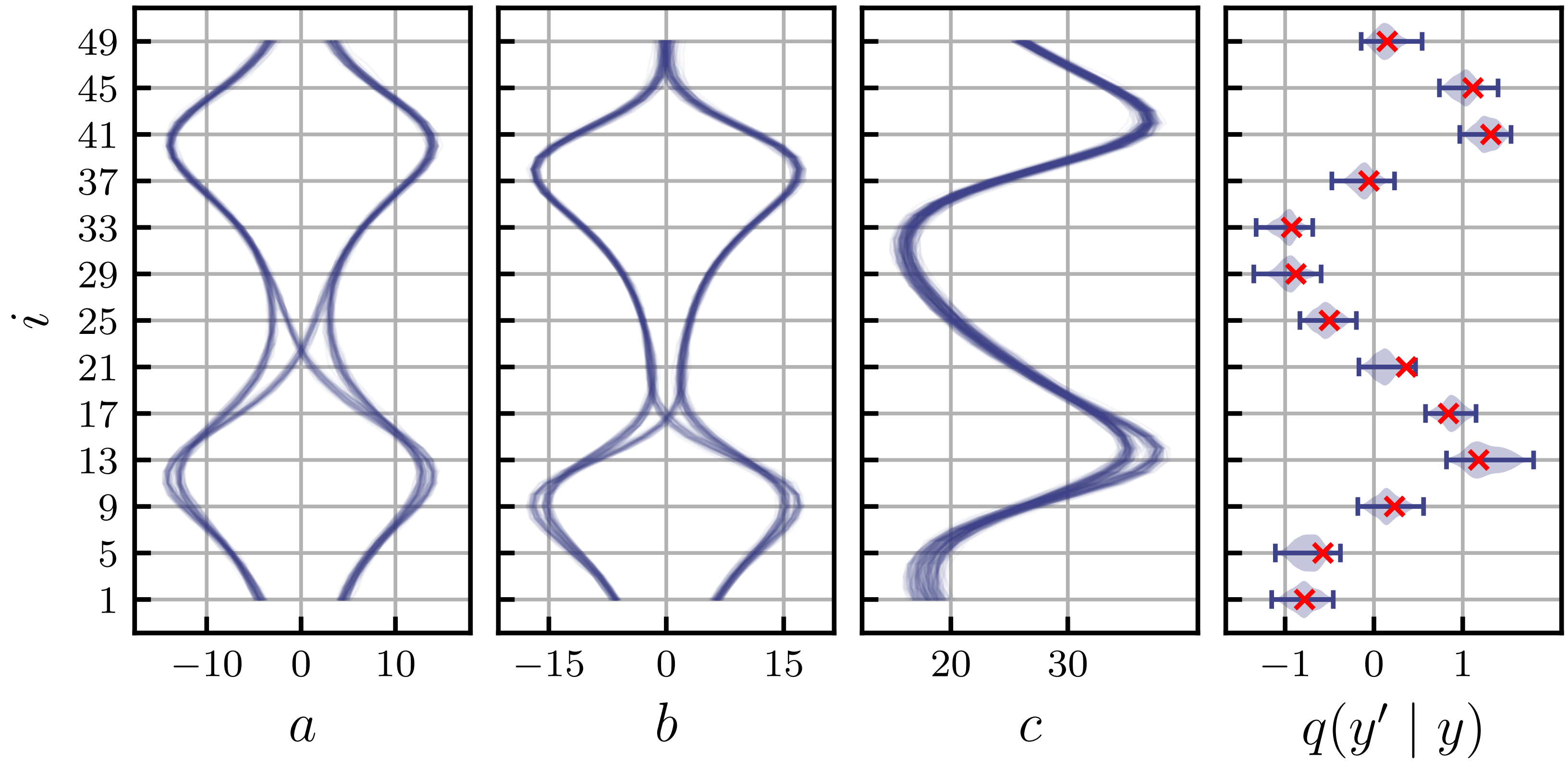


Figure 4. SDA inference is diverse and consistent with the observation.

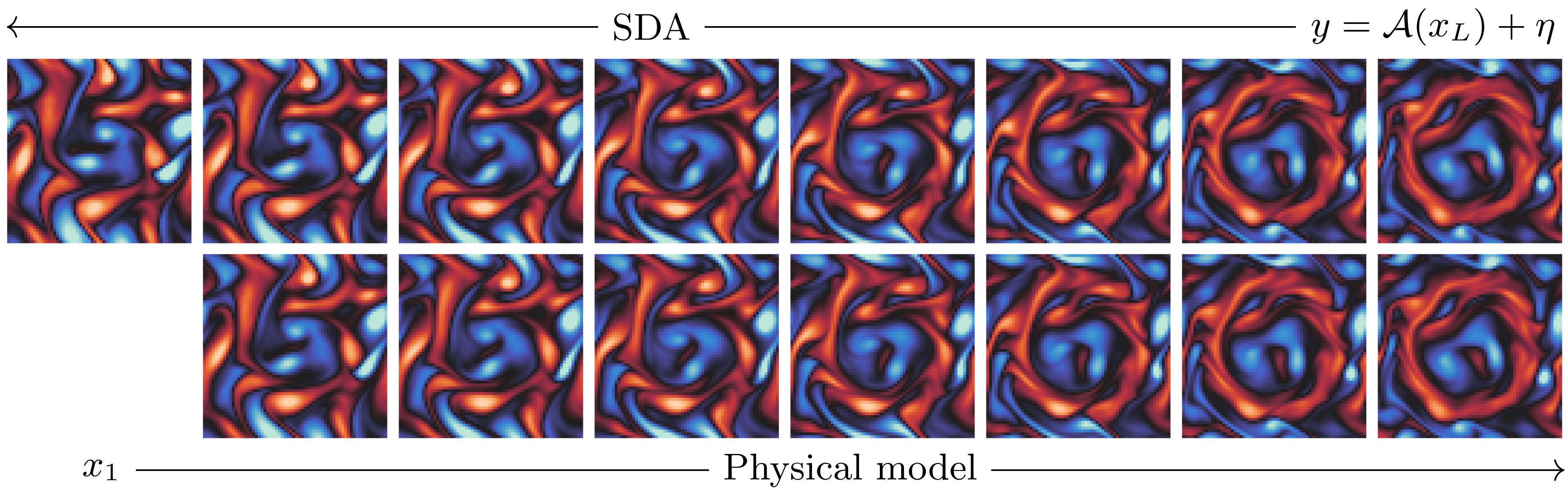


Figure 5. SDA inference is consistent with the physical model.