Physics-Integrated Variational Autoencoders for Robust and Interpretable Generative Modeling

Naoya Takeishi and Alexandros Kalousis

University of Applied Sciences and Arts Western Switzerland (HES-SO)

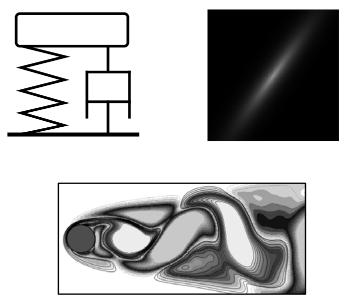
@NeurIPS 2021

Hes·so
Haute Ecole Spécialisée
de Suisse occidentale
University of Applied Sciences and Arts
Western Switzerland

dmml
group

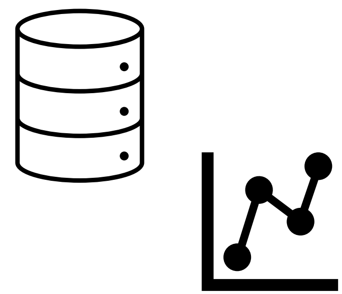
Grey-box / Hybrid Modeling

theory-based model
(physics, chemistry, biology, etc.)



\+

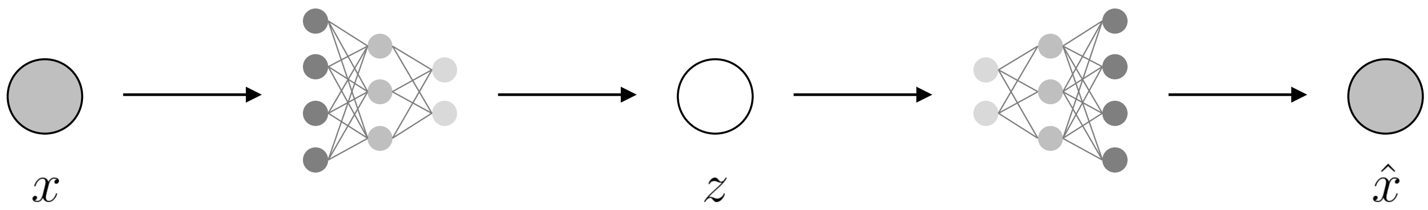
data-driven model



→ **efficient learning, partial interpretability, etc.**

Can we do it for deep generative models?

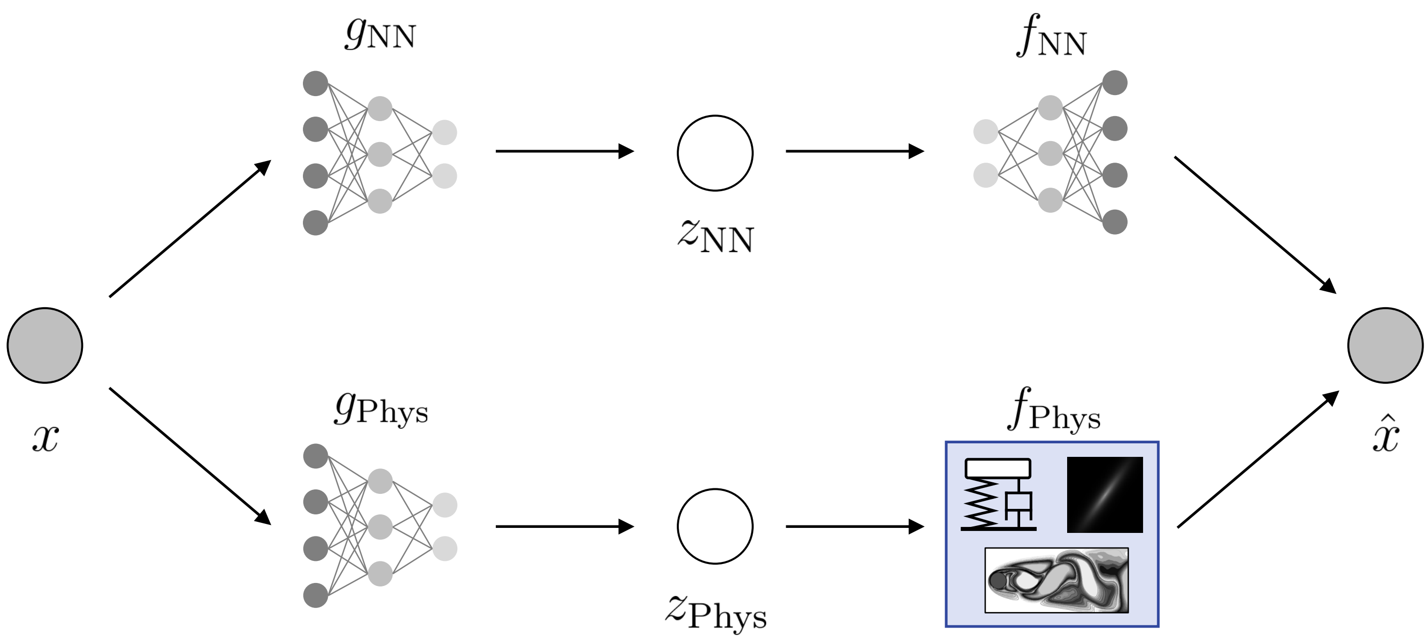
cf. Vanilla (V)AE



$$\mathbb{E}z = g(x)$$

$$\mathbb{E}\hat{x} = f(z)$$

Physics-Integrated VAEs

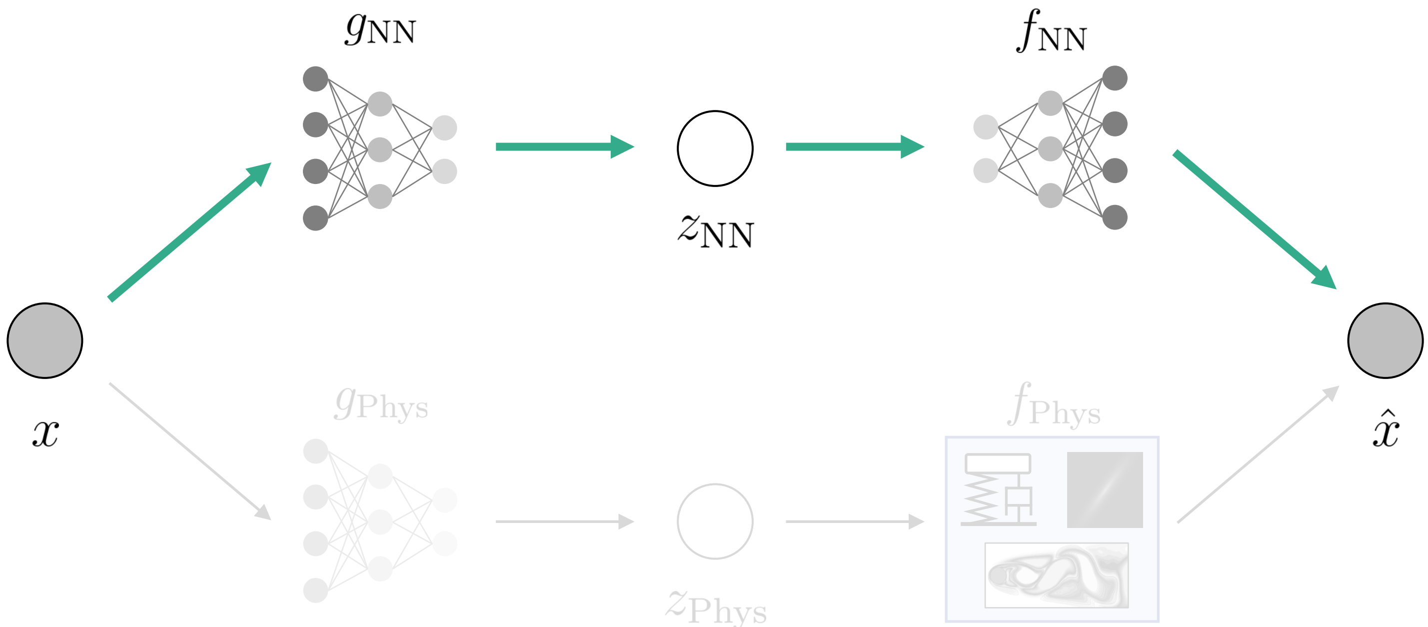


$$\mathbb{E}z_{\mathrm{NN}} = g_{\mathrm{NN}}(x)$$

$$\mathbb{E}z_{\mathrm{Phys}} = g_{\mathrm{Phys}}(x)$$

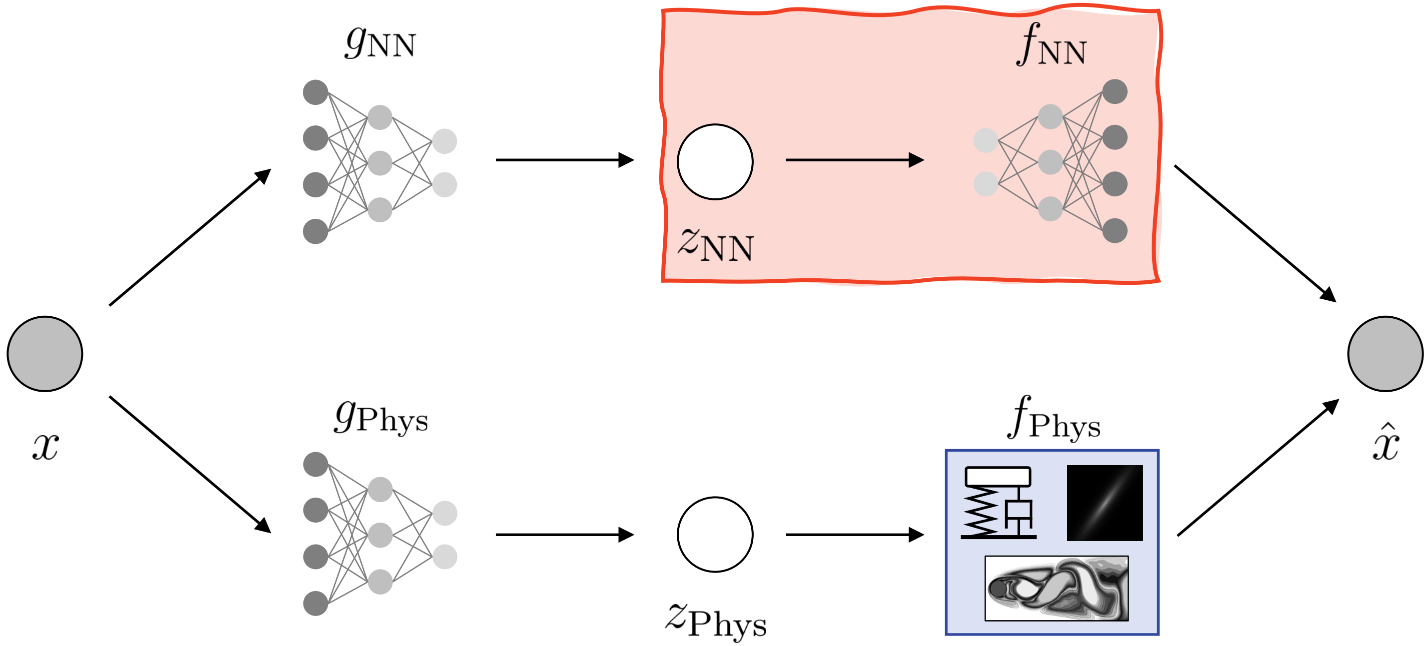
$$\mathbb{E}\hat{x} = \mathcal{F}\left[f_{\mathrm{NN}}, f_{\mathrm{Phys}}; z_{\mathrm{NN}}, z_{\mathrm{Phys}}\right]$$

Physics-Integrated VAEs: Issue



Physics part may be ignored

Regularization (1)

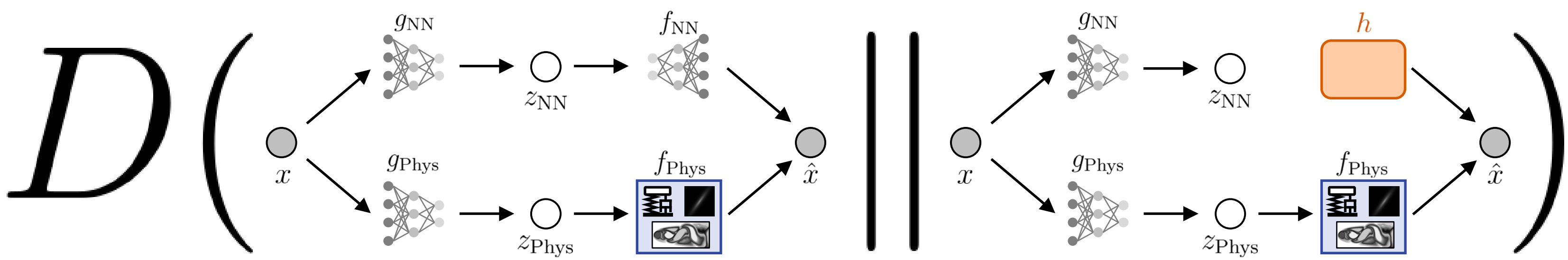


Flexibility of trainable part should be somehow suppressed

Regularization (1) cont'd

Minimize

e.g.,
- $h(x) = x$
- $h(x) = 0$
- $h(x) = Wx$



original model

"physics-only" reduced model

$$\mathbb{E} z_{\mathrm{NN}} = g_{\mathrm{NN}}(x)$$
$$\mathbb{E} z_{\mathrm{Phys}} = g_{\mathrm{Phys}}(x)$$
$$\mathbb{E} \hat{x} = \mathcal{F}\left[f_{\mathrm{NN}}, f_{\mathrm{Phys}}; z_{\mathrm{NN}}, z_{\mathrm{Phys}}\right]$$

$$\Downarrow$$

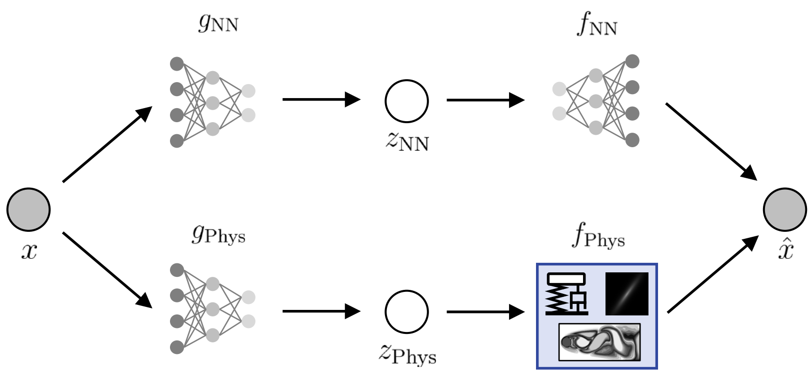
$$p(\hat{x} \mid X_{\mathrm{data}})$$

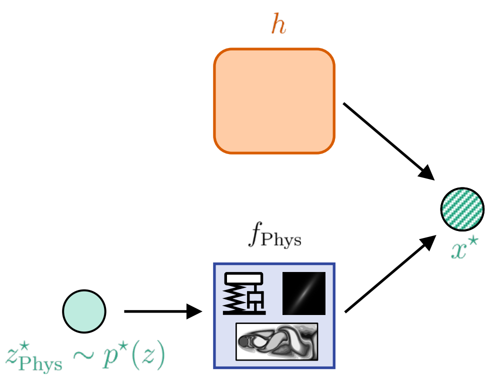
$$\cancel{\mathbb{E} z_{\mathrm{NN}} = g_{\mathrm{NN}}(x)}$$
$$\mathbb{E} z_{\mathrm{Phys}} = g_{\mathrm{Phys}}(x)$$
$$\mathbb{E} \hat{x} = \mathcal{F}\left[\underset{h}{\cancel{f_{\mathrm{NN}},}} f_{\mathrm{Phys}}; \cancel{z_{\mathrm{NN}},} z_{\mathrm{Phys}}\right]$$

$$\Downarrow$$

$$p_{\mathrm{reduced}}(\hat{x} \mid X_{\mathrm{data}})$$

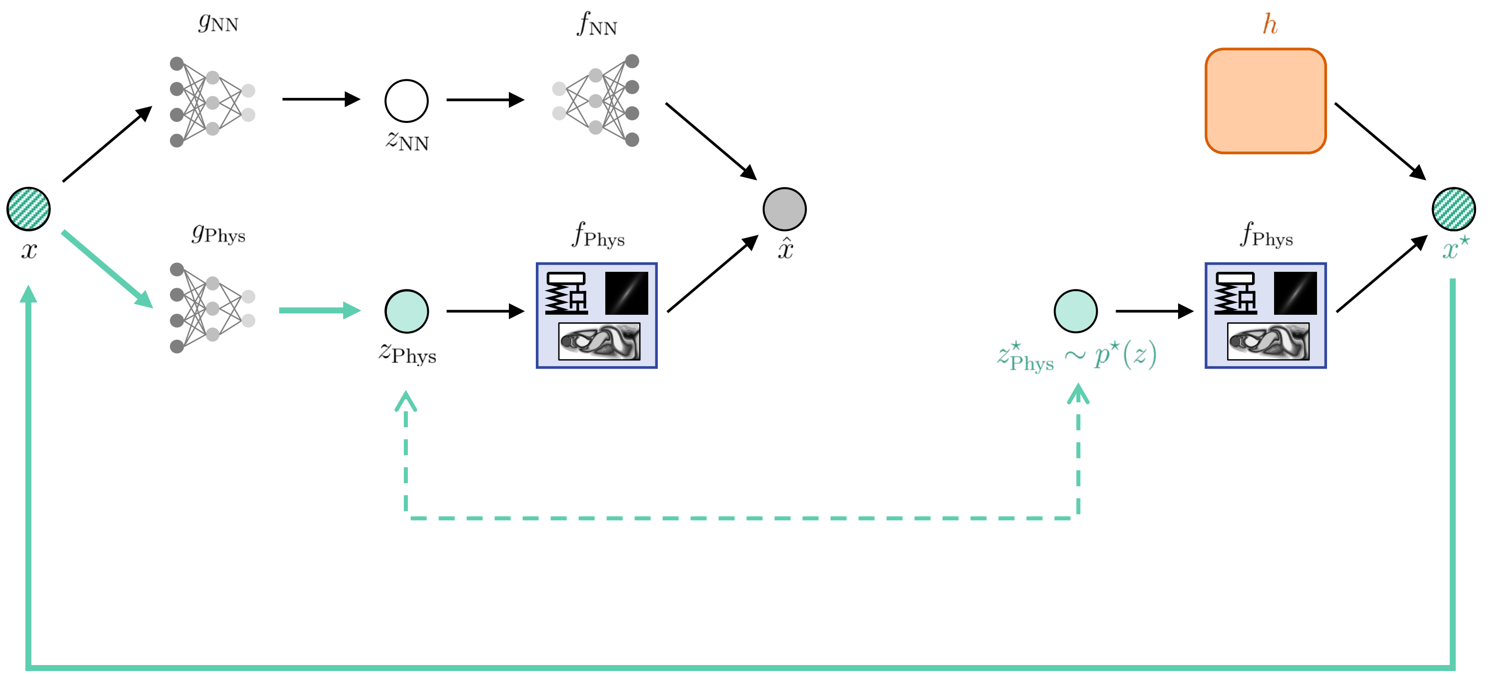
Regularization (2)





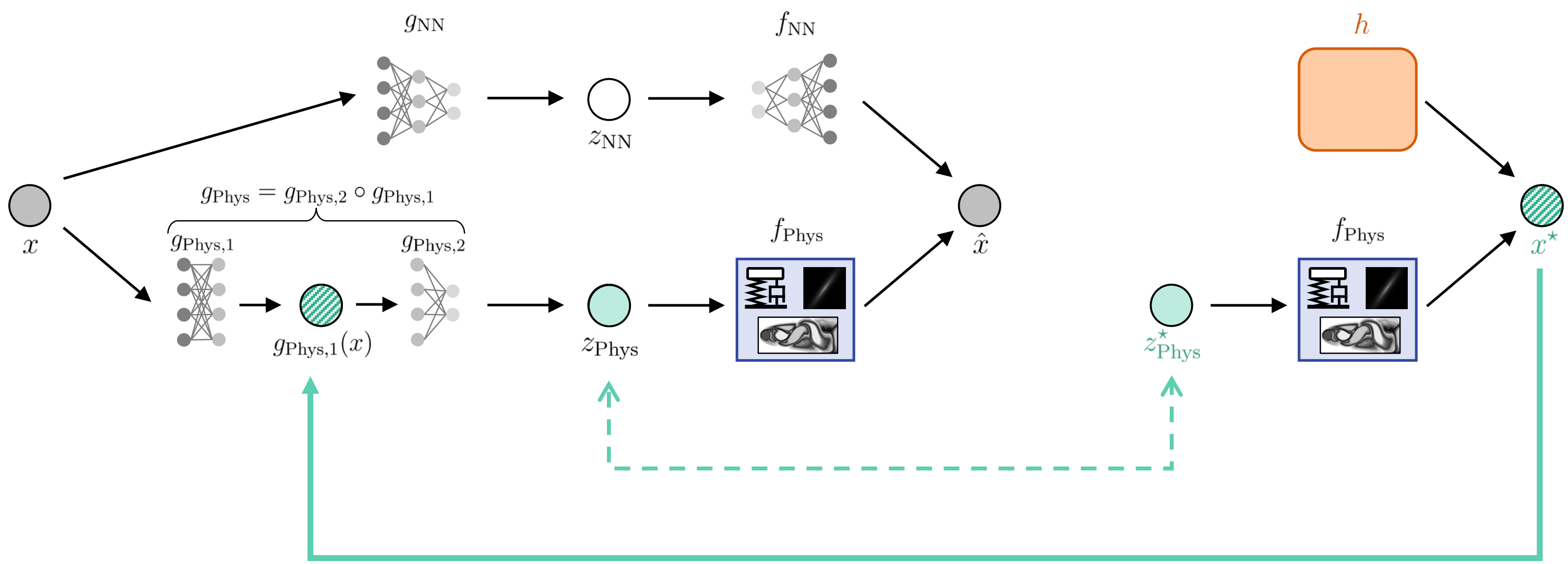
Physics part can also be used for data augmentation,

Regularization (2)

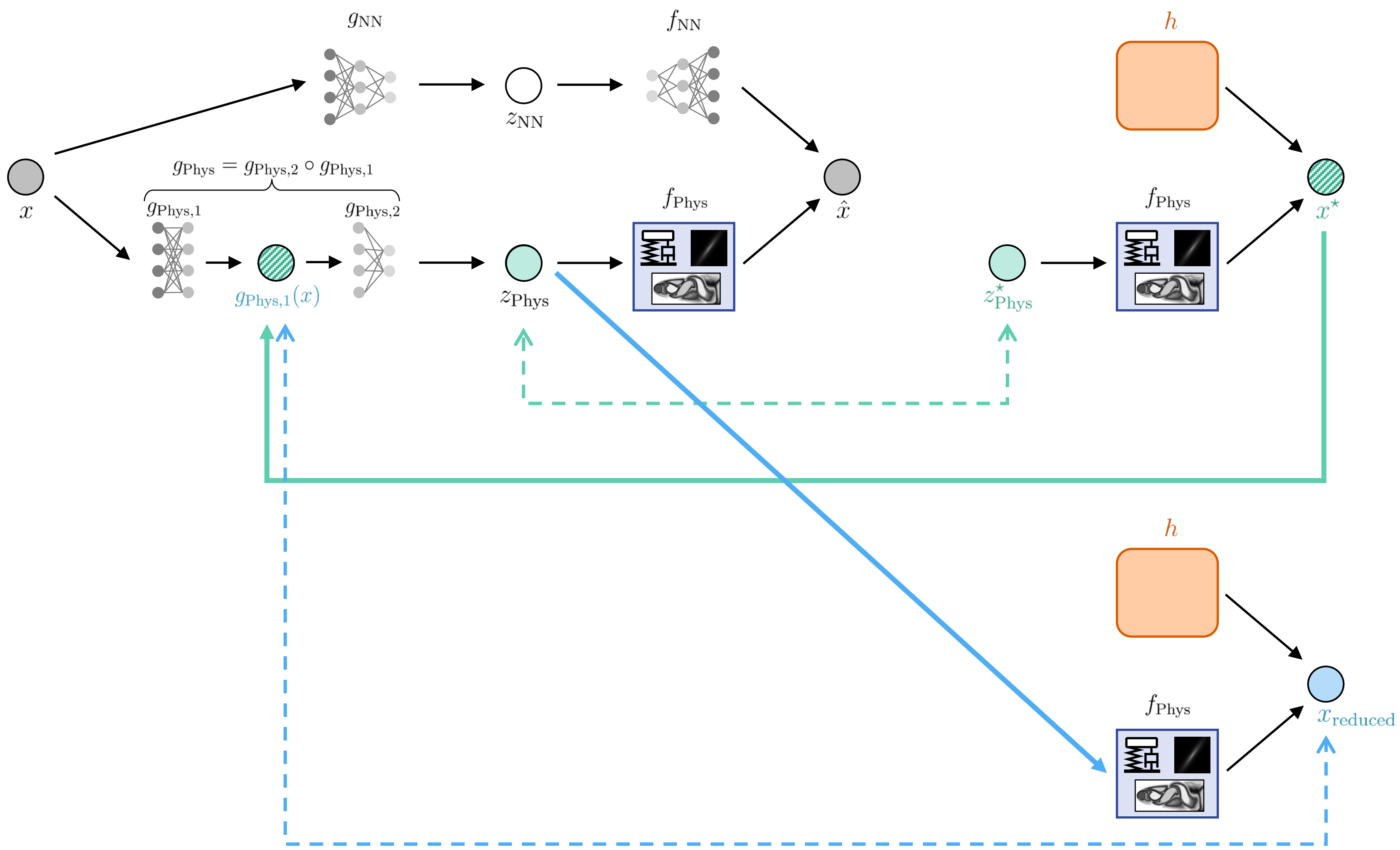


Physics part can also be used for data augmentation,
but x and $x^\star$ would have different natures ☹

Regularization (2) cont'd



Regularization (2) cont'd



Experiment (1)

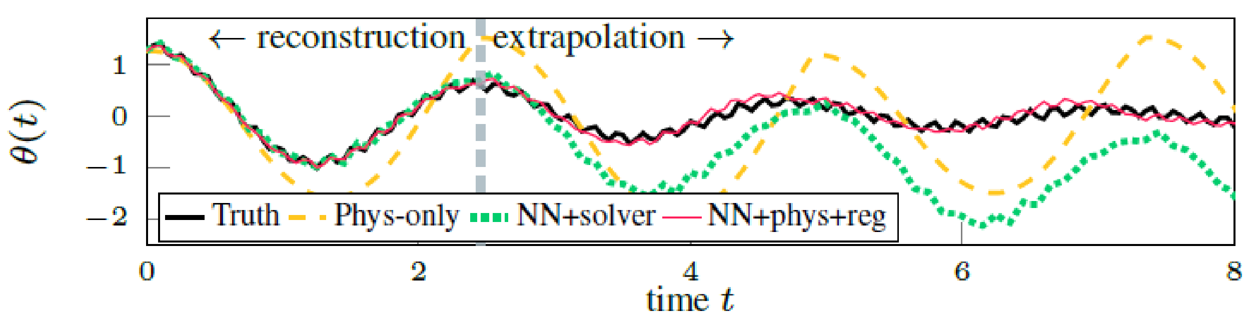
- Data: each data point = sequence of pendulum's angle ϑ

$$\ddot{\vartheta} + \omega^2 \sin\vartheta + b\dot{\vartheta} - A\omega^2 \cos(2\pi\phi t) = 0$$

- Encoder = neural nets
- Decoder = $f_{\mathrm{NN},2}\left(\mathrm{ODESolve}_\vartheta \left[f_{\mathrm{Phys}}(\vartheta, z_{\mathrm{Phys}}) + f_{\mathrm{NN},1}(\vartheta, z_{\mathrm{NN}}) = 0 \right],\ z_{\mathrm{NN}} \right)$

where $f_{\mathrm{Phys}}(\vartheta, z_{\mathrm{Phys}}) = \ddot{\vartheta} + z_{\mathrm{Phys}}^2 \sin\vartheta$

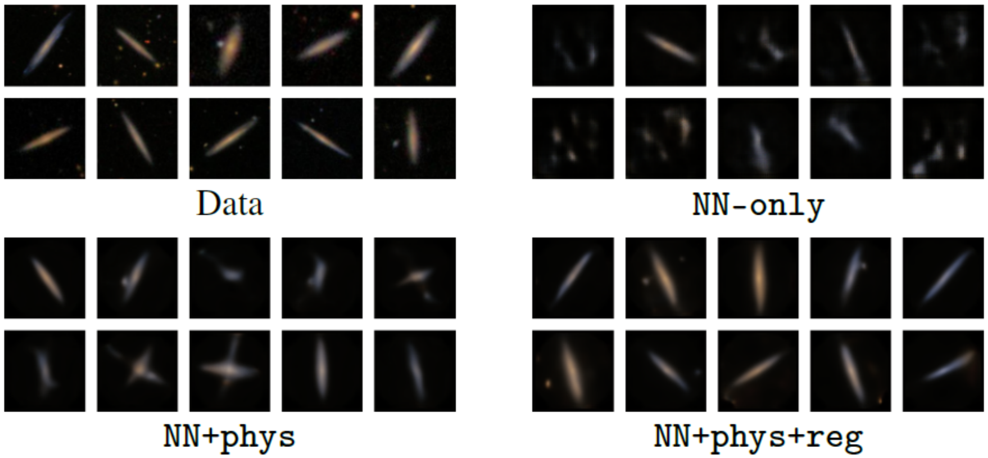
training data is only $0 \le t \le 2.5$



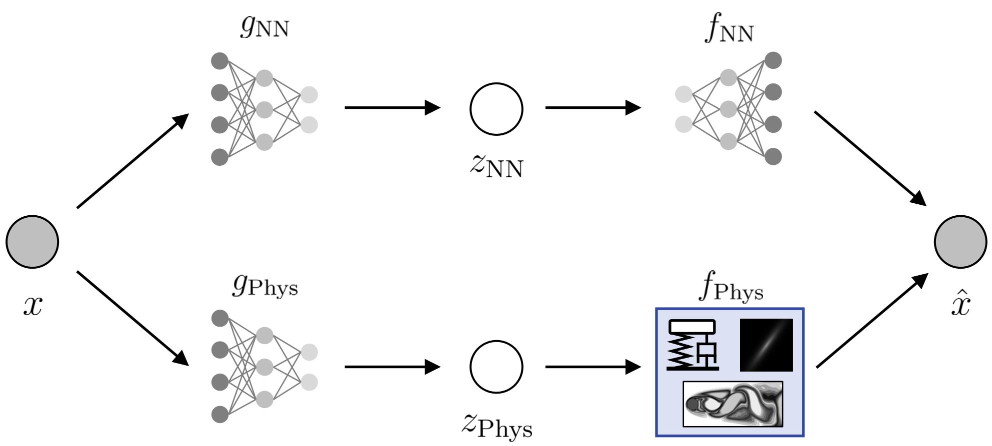
Experiment (2)

- Data: Images of galaxy

- Encoder = CNN
- Decoder = $f_{\mathrm{NN}}\left(f_{\mathrm{Phys}}\left(z_{\mathrm{Phys}}\right), z_{\mathrm{NN}}\right)$
  - f_{NN} = U-Net
  - f_{Phys} = Gaussian profile, $z_{\mathrm{Phys}} = [a, b, \theta, I]$
    - ✓ a & b = semi-major/minor axes of ellipse, θ = tilt, I = intensity



Summary



- In grey-box modeling with flexible ML models, one should be careful so that physics models are not ignored

- We have presented the method for generative models
  - applicable to non-additive combination of models
  - similarly applicable to various types of generative models