Representation Learning for Partially-Observed Systems

Said Ouala[1], Lucas Drumetz[1], Duong Nguyen[1],Bertrand Chapron[2], Ananda Pascual[3], Fabrice Collard[4], Lucile Gaultier[4], Ronan Fablet [1]

IMT Atlantique, Lab-STICC, Brest, France

Ifremer, LOPS, Brest, France

IMEDEA, UIB-CSIC, Esporles, Spain

ODL, Brest, France

Problem Formulation

Context

- Modeling Ocean Surface Dynamics
- Forecasting and Data-Assimilation Applications

Challenges

- Unknown Equations, Too-Complicated to be used
- Non-Linear Dynamics with Sensitive Stability Behaviour
- Partial and/or Noisy Observations

Problem Formulation

Challenges

- Unknown Equations, Too-Complicated to be used ➔ Data-Driven Representaions
- Non-Linear Dynamics with Sensitive Stability Behaviour ➔ Long-term charactarization of the models
- Partial and/or Noisy Observations ➔ Problem dependent, several considerations

Problem Formulation

Challenges

- Temporal sparse data
  - High order integration schemes inference
- Partially Observed Systems ➔ Some componenets, influencing the Dynamics are never observed
  - Stochastic ➔ Stochastic models
  - Deterministic ➔ Embedology

Residual Integration Neural Networks

Proposed Framework

High order integration schemes inference, motivation

- Continuous setup : Estimate Derivatives $\dot{\mathbf{z}}_t = \underset{\text{NN}}{f}(\mathbf{z}_t)$ $\mathbf{z}_{t_{n+1}} = \mathbf{z}_{t_n} + h\underset{\text{NN}}{\Psi}(t_{n+1}, \mathbf{z}_{t_{n+1}}, h)$
- Discrete setup : Transform ODE into discrete equation

Numerical Integration scheme

Which integration scheme to use ?

- Tradeoff numerical complexity/ Precision ?
- Identifiability

Proposed Framework

High order integration schemes inference, jointly to the data driven dynamical model

$$\dot{\mathbf{z}}_t = \underset{NN}{f}(\mathbf{z}_t) \qquad \mathbf{z}_{t_{n+1}} = \mathbf{z}_{t_n} + h\underset{NN}{\Psi}(t_{n+1}, \mathbf{z}_{t_{n+1}}, h)$$

$$\min_{\theta_{NN},c,\beta,\alpha} \sum_{n=1}^{N} \|\mathbf{z}^T_{t_n} - \Psi(\mathbf{z}^T_{t_{n-1}}, \theta_{NN}, c, \beta, \alpha)\|$$

$$\text{subject to} \quad \sum_{i=1}^{\hat{s}} \beta_i = 1, \;\; \forall i, \; 0 < c_i < 1 \text{ and } \sum_{j=1}^{i-1} \alpha_{i,j} = c_i$$

Proposed Framework

Case study : sub-sampled Lorenz 63

h = 0.2

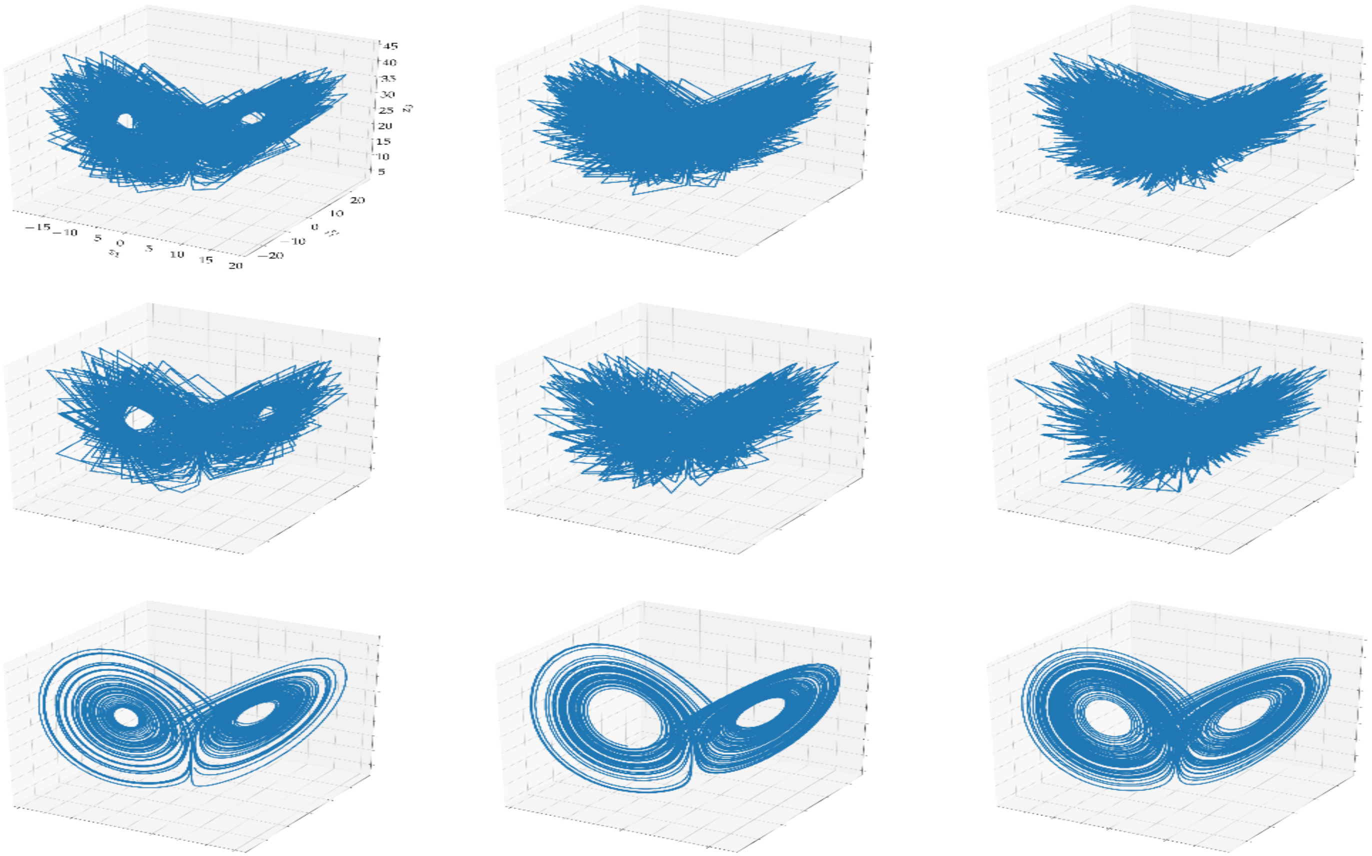
h = 0.3

h = 0.4

Training data

Simulated attractor

Simulated attractor



Proposed Framework

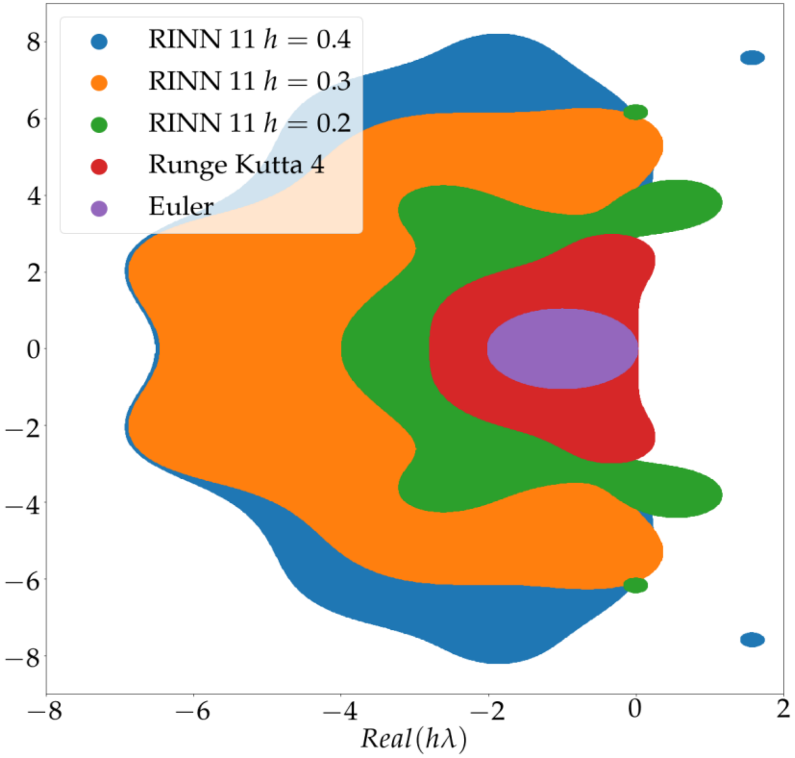
Learnt Integrators

$$\mathcal{R}_{RINN_{h1}}(h\lambda) = 1 + h\lambda + 0.5016(h\lambda)^2 + 0.1613(h\lambda)^3 + 0.0429(h\lambda^4)$$
$$+ 0.00909(h\lambda^5) + 0.001613(h\lambda^6) + ...$$

$$\mathcal{R}_{RINN_{h2}}(h\lambda) = 1 + h\lambda + 0.5020(h\lambda)^2 + 0.1640(h\lambda)^3 + 0.03952(h\lambda^4)$$
$$+ 0.00731(h\lambda^5) + 0.00104(h\lambda^6) + ...$$

$$\mathcal{R}_{RINN_{h3}}(h\lambda) = 1 + h\lambda + 0.5057(h\lambda)^2 + 0.1684(h\lambda)^3 + 0.04296(h\lambda^4)$$
$$+ 0.008299(h\lambda^5) + 0.001342(h\lambda^6) + 1.810^{-4}(h\lambda^7) + 2.035E^{-5}(h\lambda^8)...$$

$$\mathcal{R}_{exp}(h\lambda) = 1 + h\lambda + 0.5(h\lambda)^2 + 0.16666(h\lambda)^3 + 0.04166(h\lambda^4)$$
$$+ 0.008333(h\lambda^5) + 0.001388(h\lambda^6) + 1.984^{-4}(h\lambda^7) + 2.480E^{-5}(h\lambda^8)...$$



Partially Observed Systems

Proposed Framework

- Classical state-of-the-art :
  - Measuring generic independent variables
  - Finding a geometrical reconstruction from a single observed variable
- Issues
  - Independet of a data (or model) driven formulation

Proposed Framework

- Our approach : Project the observation x into a high dimensional space u, with u=[x,$l1$,$l2$,…,ln]
- Solve the Following optimization problem

Fit :

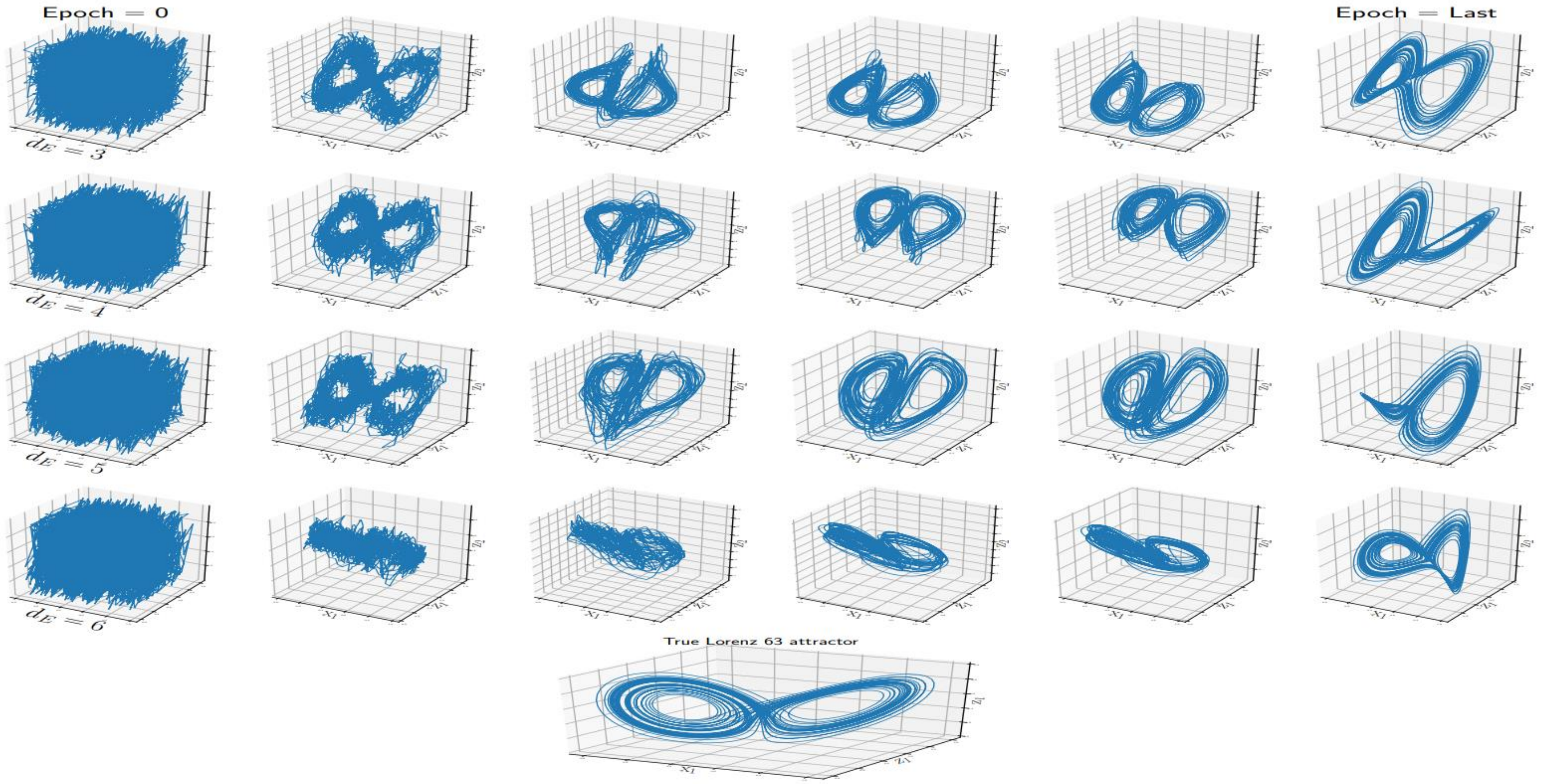
$$\frac{du}{dt} = f_\theta(u)$$

With

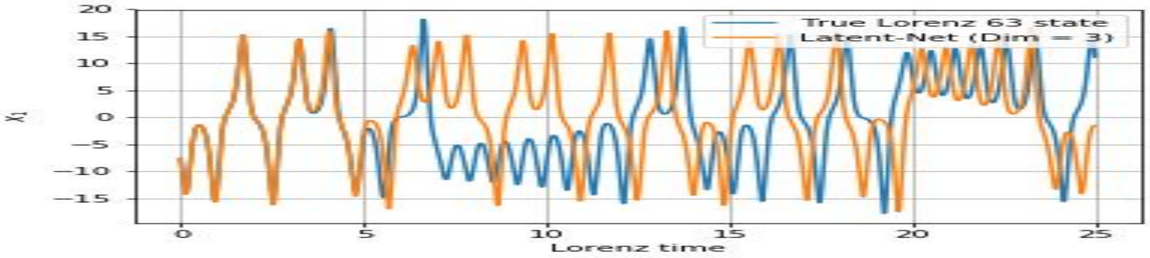
$$u = [x, l1, l2, \dots, ln]$$

$$\theta, l = argmin_{\theta,l}\{\alpha|x(t) \ - G(\int_{t-1}^{t} f(u(t'))dt')| + (1-\alpha)|u(t) - \int_{t-1}^{t} f(u(t'))dt' \ |\}$$

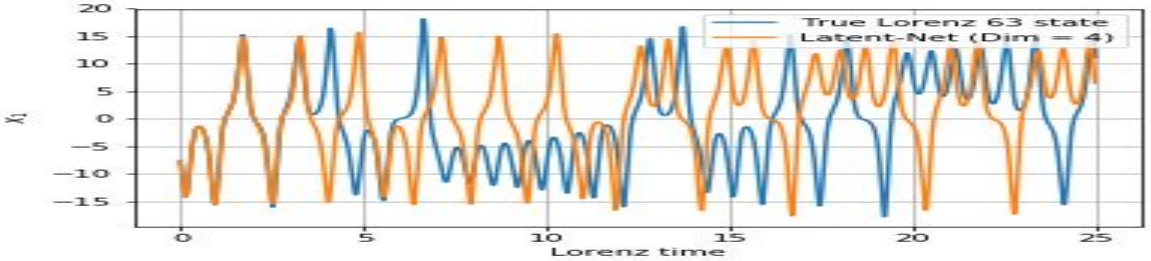
Proposed Framework
Attractor Reconstruction



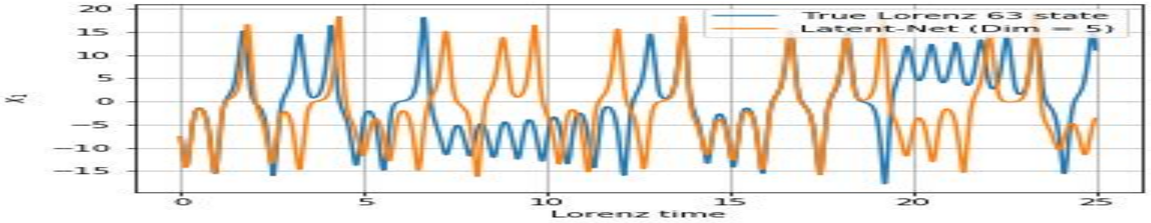
Proposed Framework
Forecast



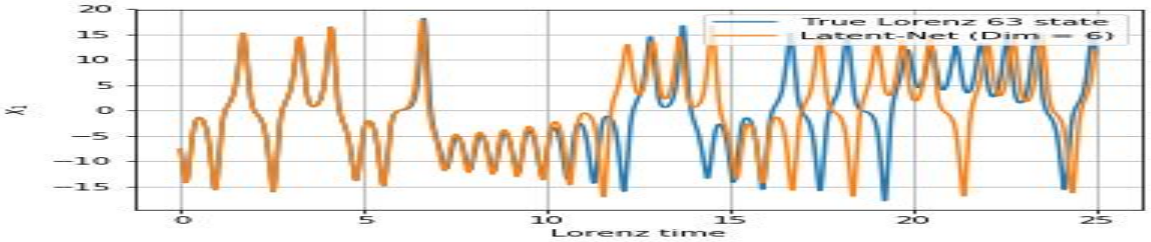
Lyap = 0,82



Lyap = 0,96



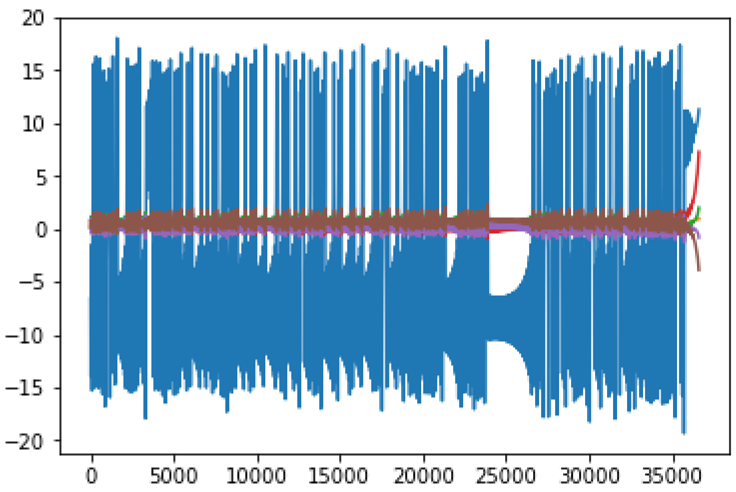
Lyap = 0,82

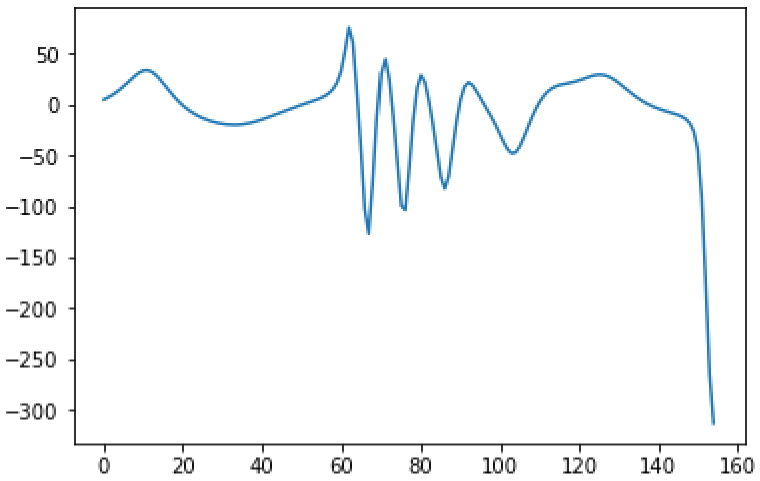


Lyap = 0,87

Proposed Framework
Issues

- Our spawned manifold is not dense in the phase space :/



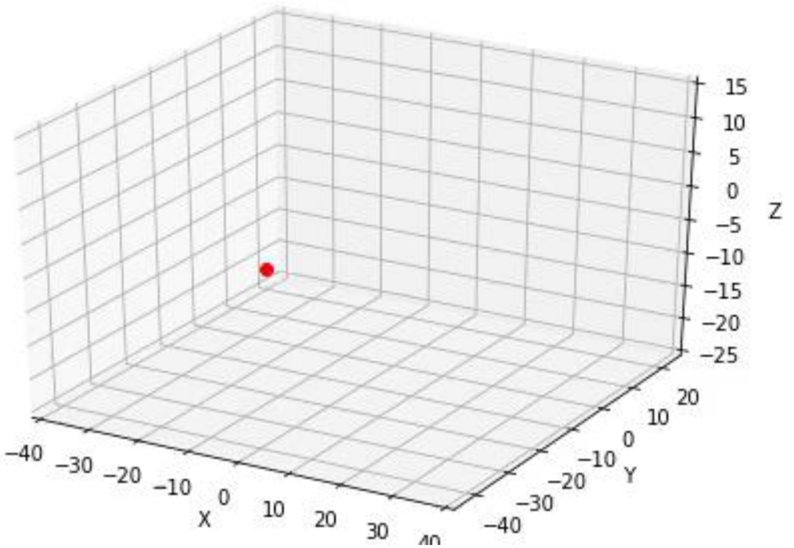


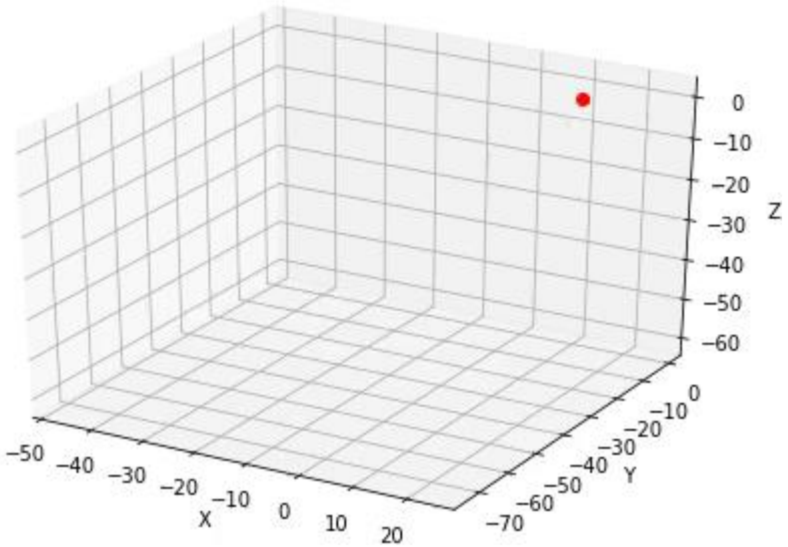
Proposed Framework
Idea

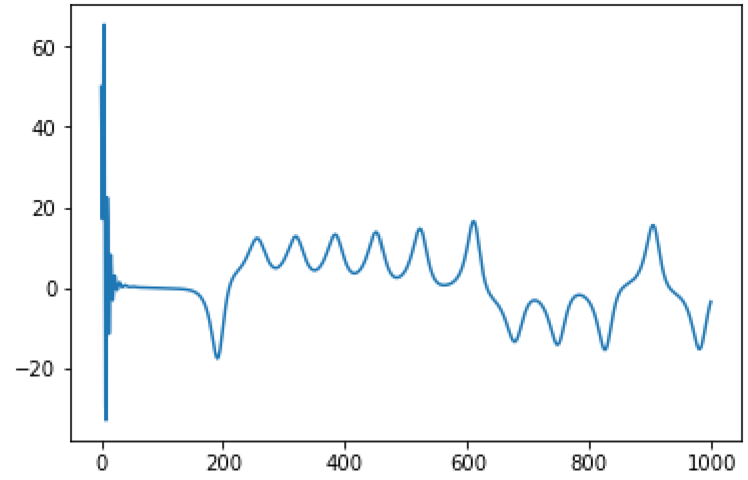
- Boundedness constraints : Constraint the trajectories of the dynamical system to live in a closed ball in the phase space
- In practice : Energy preserving non linearity + Negative eigenvalues of the linear part of (a shifted version of) the model (Schlegel et al. 2013):

$$
\begin{aligned}
\hat{\theta}, \mathbf{y}_{1:T} = \arg\min_{\theta} \min_{\{\mathbf{y}_t\}_t} \sum_{t=1}^{T} \|\mathbf{x}_t - G\left(\Phi_{\theta,t}\left(\mathbf{u}_{t-1}\right)\right)\|^2 \\
+ \lambda\|\mathbf{u}_t - \Phi_{\theta,t}(\mathbf{u}_{t-1})\|^2 \\
+ \lambda_2\|\mathbf{u}_t\mathcal{N}(u_t)\|^2 \\
+ \lambda_3\|\mathrm{Relu}(\alpha)/\mathrm{Relu}(\alpha+1)\|^2
\end{aligned}
$$

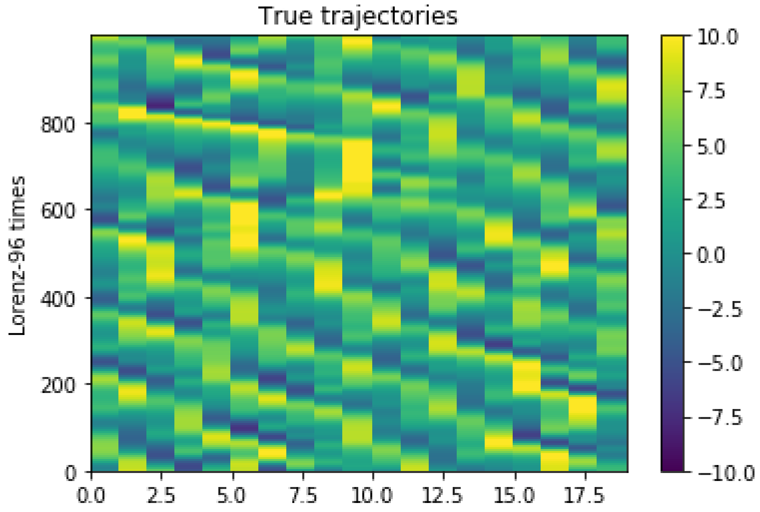
Proposed Constrained Framework Lorenz 63

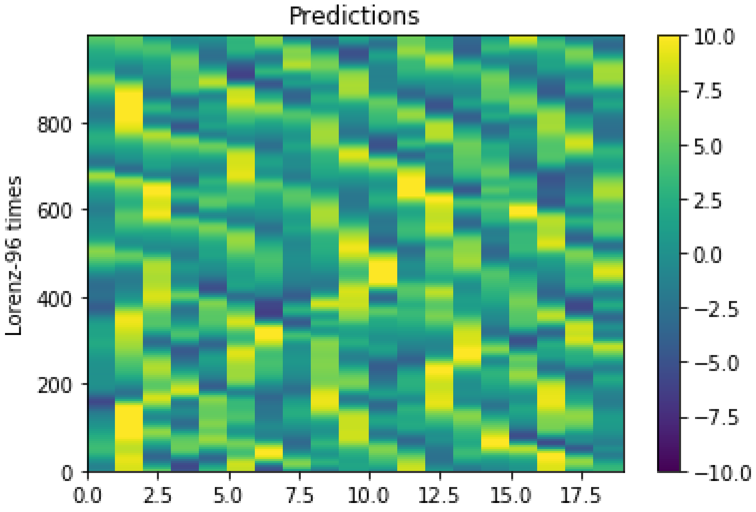


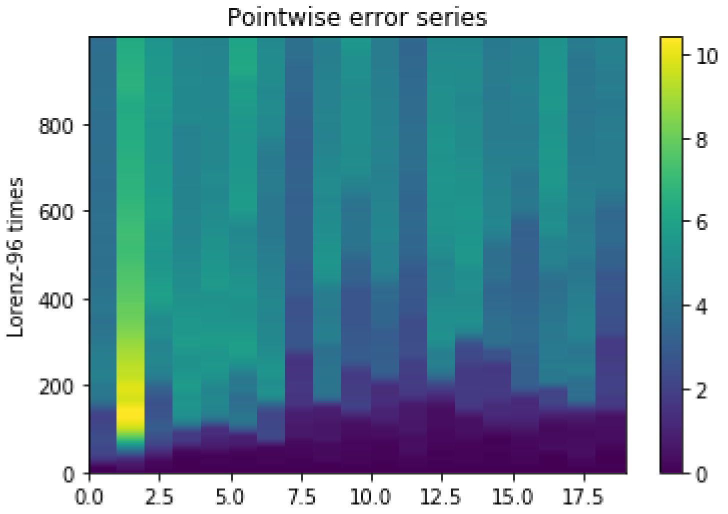


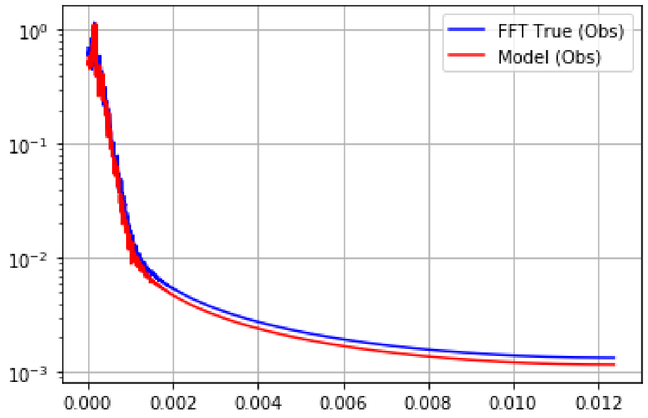


Proposed Framework
Lorenz 96







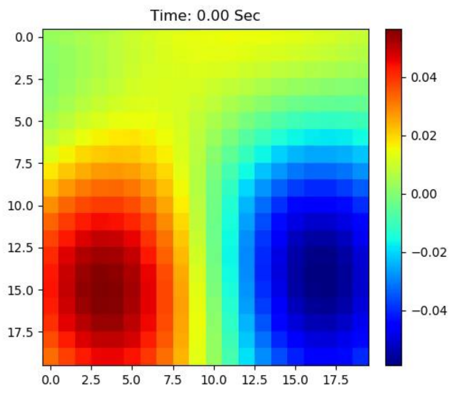


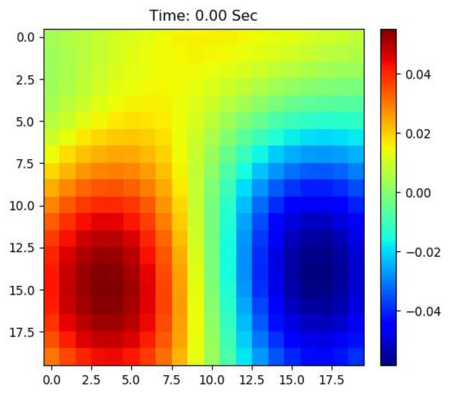
Proposed Framework
SWE

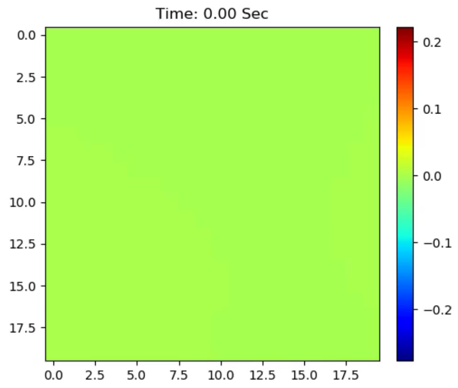
True Shallow water

Model Simulation

Error (RMSE)

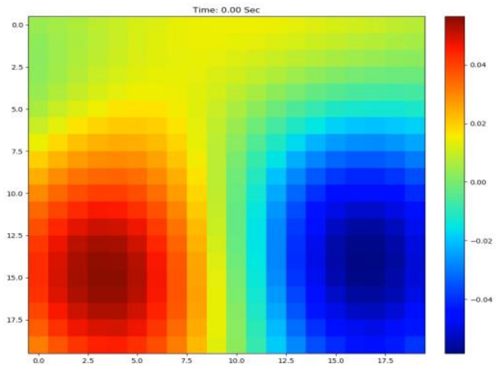




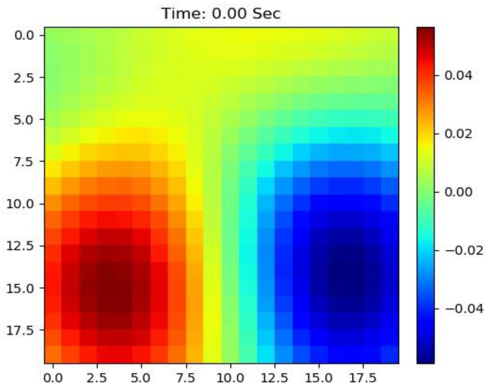


Proposed Framework
SWE

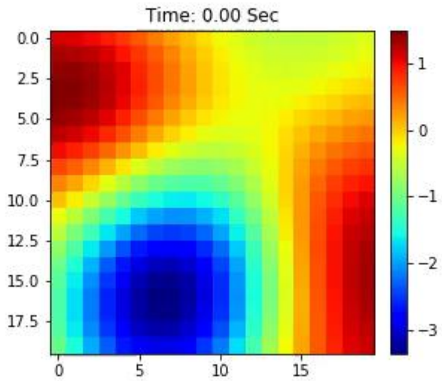
Model Simulation #1



Model Simulation from a perturbed initial condition

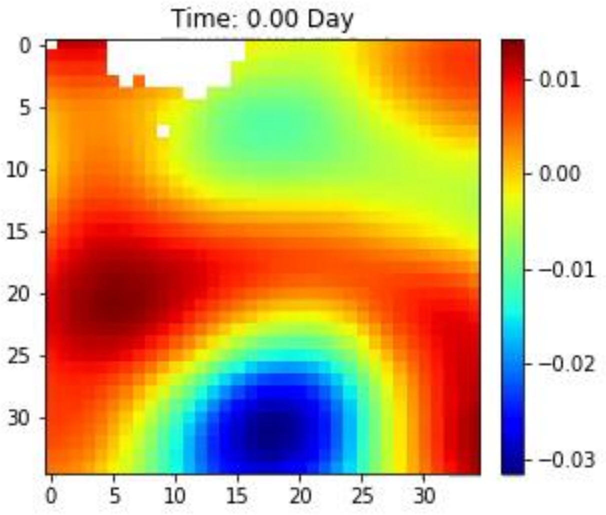


Model Simulation from a far initial condition

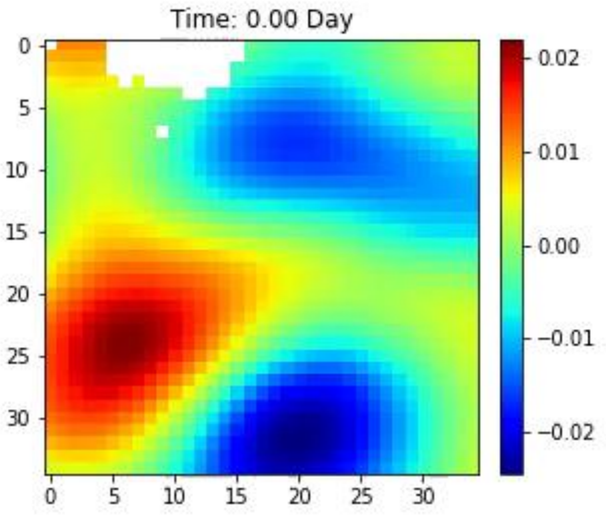


Proposed Framework
SLA-A

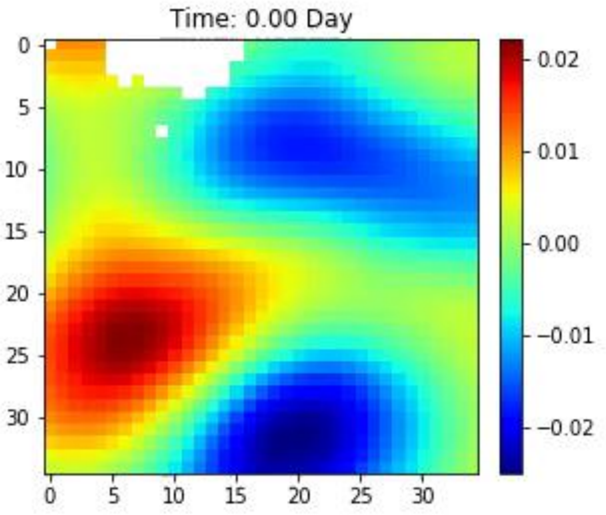
True State



Model Simulation



Model Simulation from a perturbed initial condition



Proposed Framework
Koopman

- Our approach : Project the observation x into a high dimensional space u, with u=[x,$l1$,$l2$,…,ln]
- Solve the Following optimization problem

Fit :

$$\frac{du}{dt} = Au$$

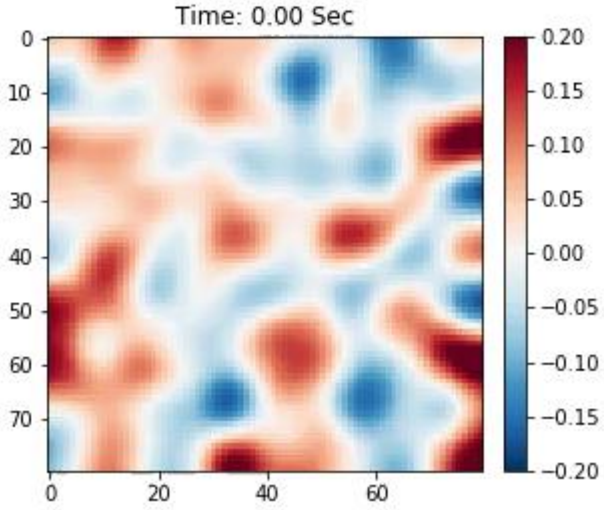
With

$$u = [x, l1, l2, \dots, ln]$$

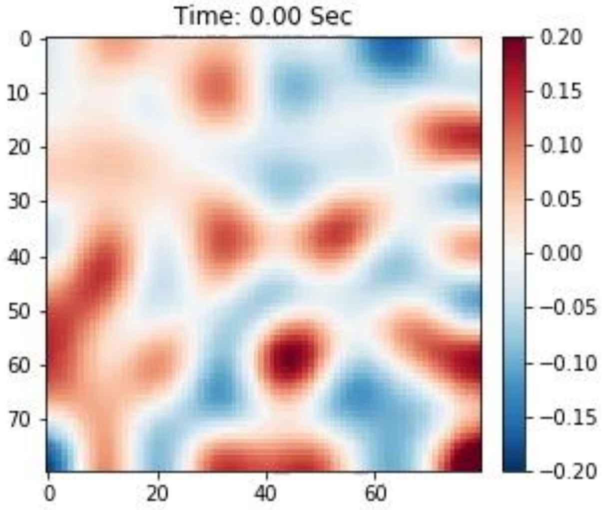
$$\theta, l = argmin_{\theta,l} \left\{ \alpha \left| x(t) - G\left(\int_{t-1}^{t} Au(t')dt' \right) \right| + (1-\alpha) \left| u(t) - \int_{t-1}^{t} Au(t')dt' \right| \right\}$$

Proposed Koopman-Framework SWE

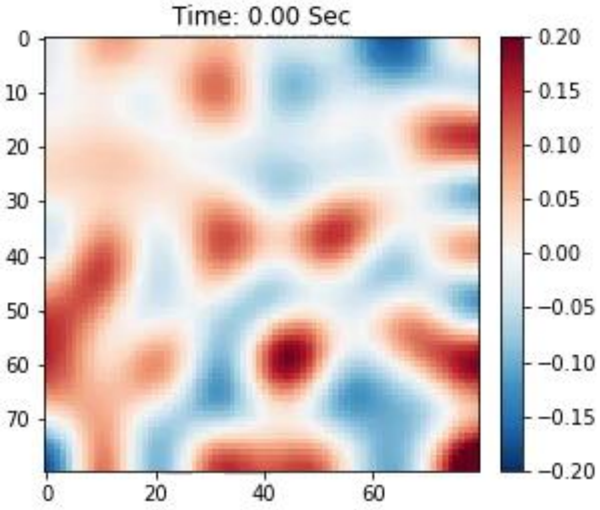
True State



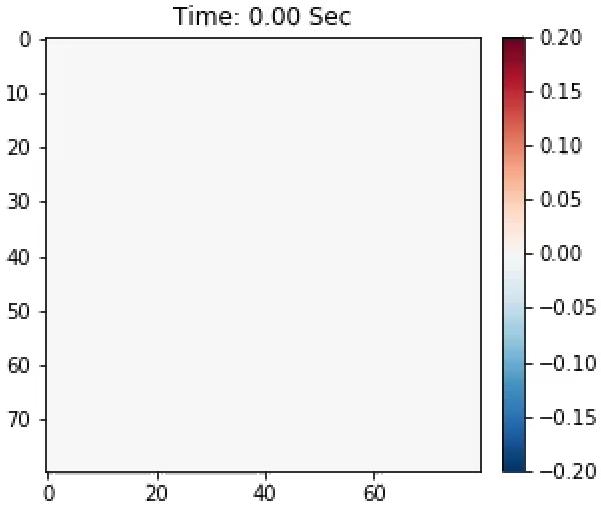
Projection



Koopman

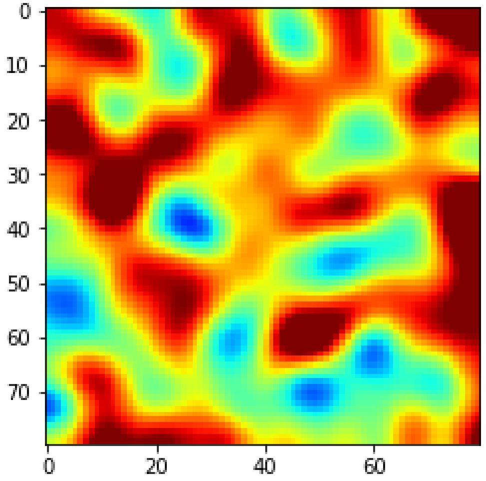


Error (MAE)

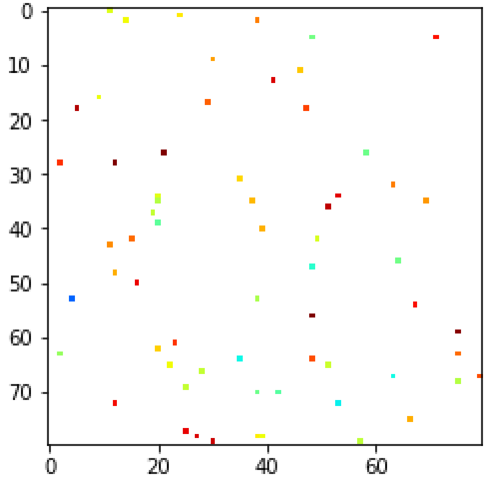


Proposed Koopman Data Assimilation

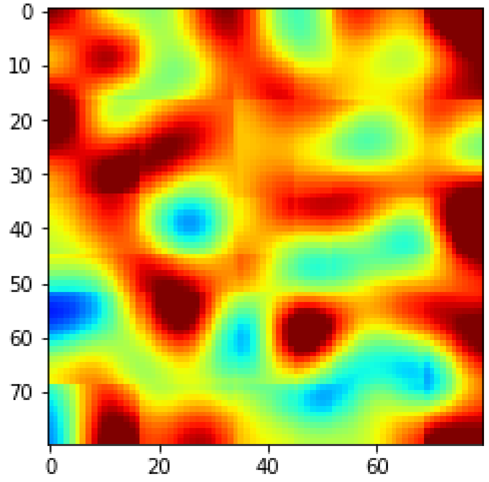
Ground truth



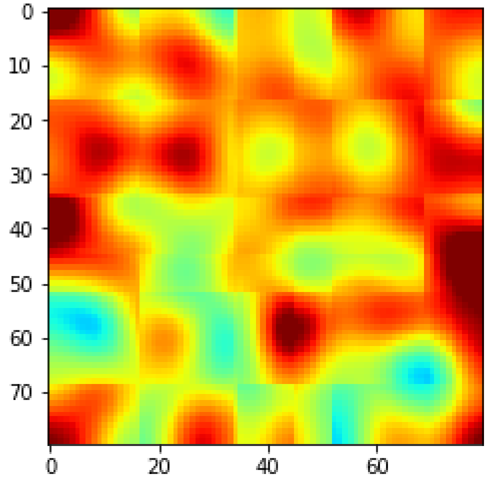
Observation



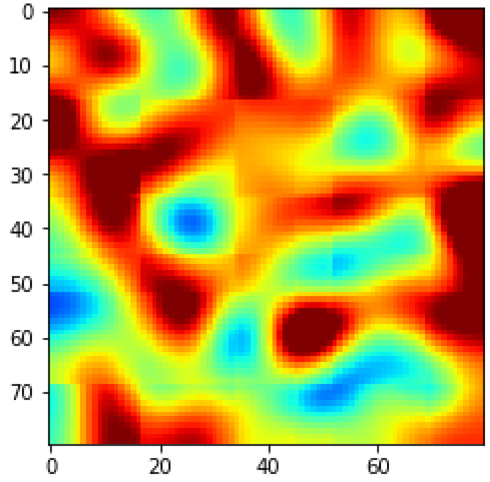
Projection



AnDA



Koopman KF



Résumé

- The proposed method allows the identifications of ODEs for partially observed systems
- The ODE is parametrized as a LQM model, in order to garentee boundedness
- In the Linear case, our method is equivalent to learn both a Koopman operator and its observables

Perspectives

- Constrained formulation of latent states to explain specific variability (high resolution scales, POD truncation modes …etc.)
- Model formulation is dependent on finding an appropriate Lyapunov function