Machine-aided guessing and gluing of unstable periodic orbits

Journal Club

Will An

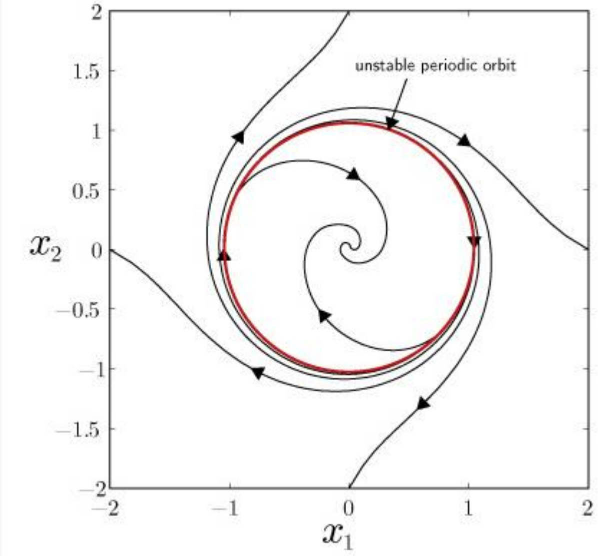
Introduction

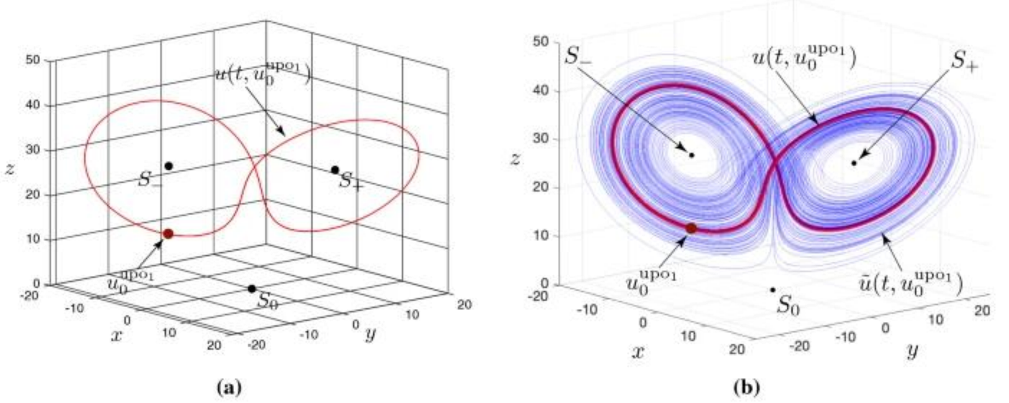
- This paper uses Machin-aided method to obtain initial guesses of Unstable Periodic Orbit (UPO) in the chaotic regime.
- It first uses autoencoder to decrease the dimension of chaotic regime to a latent space and then generate guesses.
- It also utilizes two optimizers to converge the initial guesses to true UPO with a machine precision.
- Besides those, it tries to glue gained UPOs in the latent space to form longer UPOs as well.

Introduction - UPO

- UPOs play an important role in supporting chaotic dynamics in many driven dissipative nonlinear systems.

- It is challenging to identify UPOs in high-dimensional chaotic systems.





Period-1 UPO $u^{\mathrm{upo}_1}(t)$ (red, period $\tau_1 = 1.5586$) stabilized using UDFC method, and pseudo-trajectory $\tilde{u}(t, u_0^{\mathrm{upo}_1})$ (blue, $t \in [0, 100]$) in system (1) with parameters $r = 28, \sigma = 10$, $b = 8/3$. (Color figure online)

Introduction – Finding UPO

- Usually in two steps:
- Define an adequate guess for an UPO
  - recurrency methods (find sub-trajectories in Direct Numerical Simulation (DNS) that almost close in on themselves)
- Converge the guess to a solution of the system
  - Newton algorithm (like gradient descent)
  - loop convergence algorithm

- Disadvantages: recurrency methods are biased towards the same few frequently visited UPOs (short and less unstable ones)
- Newton algorithm could encounter exponential error amplification when time-integrating a chaotic dynamical system.

Method – New Way of Finding UPO

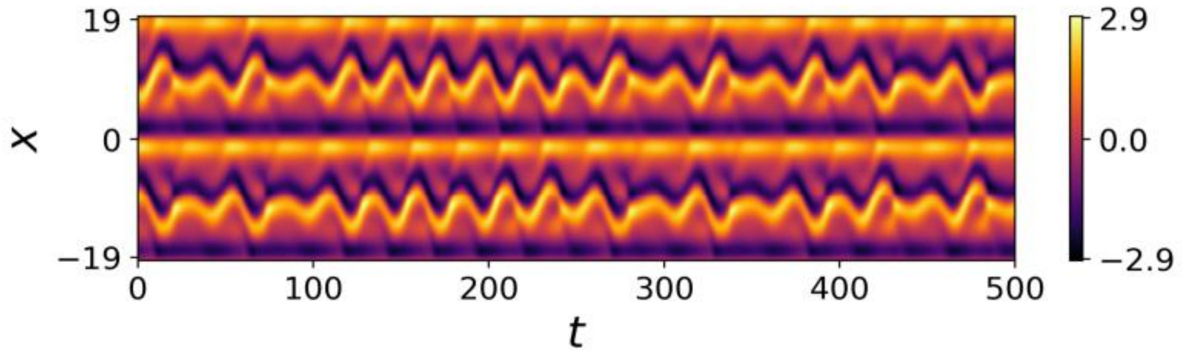
- 1. First obtain data from PDE simulation
- 2. Train an autoencoder with data given.
- 3. Do dimensionality reduction with trained autoencoder and find proper orthogonal decomposition (POD) modes in latent space.
- 4. Define loop guesses L: $\boldsymbol{L}(s) = \bar{\boldsymbol{h}} + \sum_{k=1}^{K} a_k(s)\boldsymbol{\xi}_k$
- 5. Decode the guesses back to original phase space.
- 6. Use loop convergence (adjoint solver) + newton optimizer to converge the guesses to true UPOs.

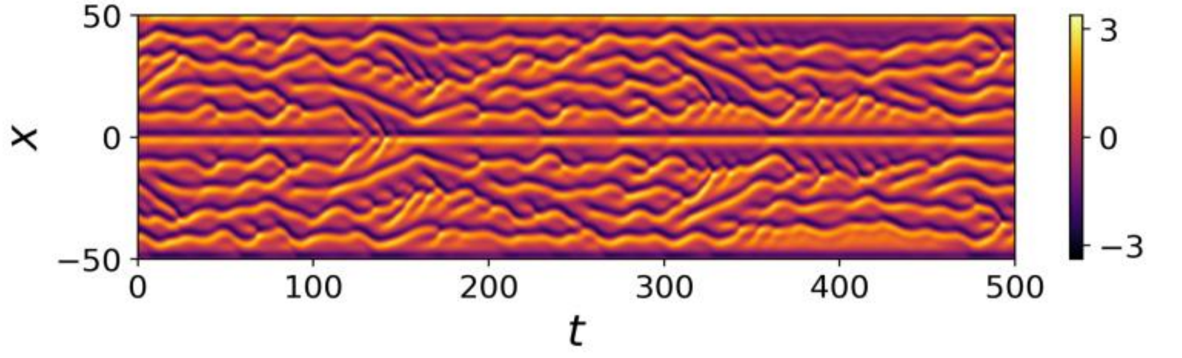
Method – PDE Model

$$u_t + uu_x + u_{xx} + u_{xxxx} = 0$$

- 1D Kuramoto-Sivashinsky equation (KSE) is used.
- The spatial domain is L-periodic and L determines the nonlinear property.

- L = 39: Low-dimensional Chaos
- L = 100: Hyperchaos





Method - Autoencoder

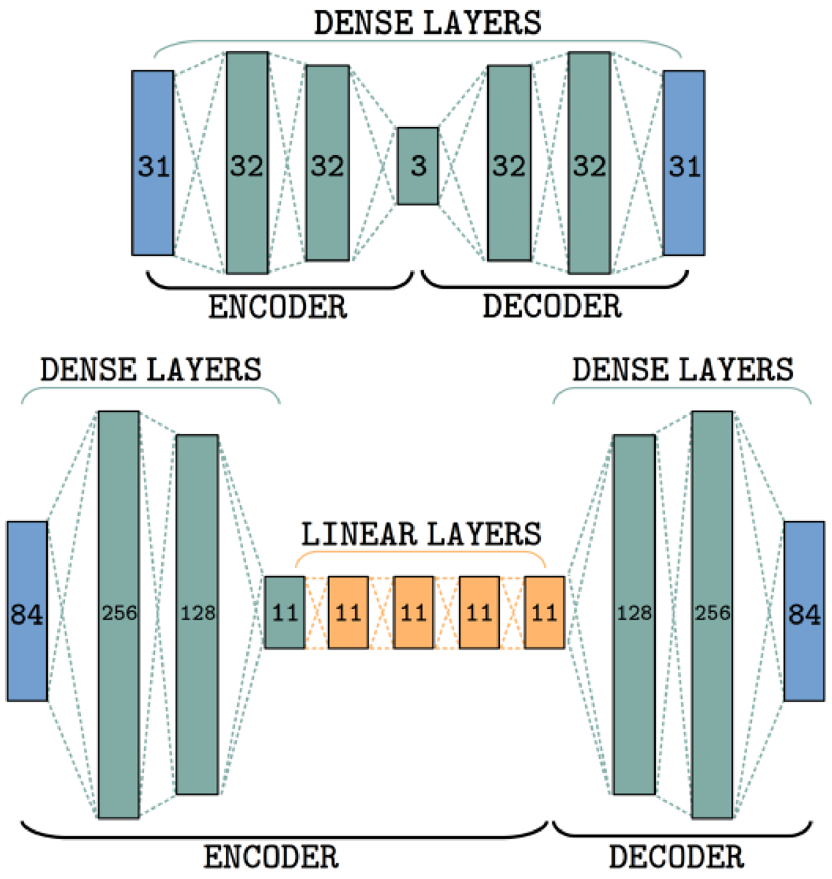


FIG. 2. Autoencoder architecture for $L = 39$ (top) and $L = 100$ (bottom, with linear layers for implicit rank minimization [44]). The number in each layer indicates the number of nodes. The dense layers use ReLU activation function.

All layers except the linear layers: $\mathrm{ReLU}(x) = \max\{0, x\}.$

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N} \frac{||\mathcal{D} \circ \mathcal{E}(\boldsymbol{y}_n) - \boldsymbol{y}_n||^2}{||\boldsymbol{y}_n||^2 + \epsilon}$$

Method – Loop Guess

- Do proper orthogonal decomposition in latent space.
- Consider a long time-series stacked in a matrix U: $\boldsymbol{U} \in \mathbb{R}^{p \times N}$
  - where p is total time step and N is dimension of latent space. $\{\boldsymbol{u}_i\}_{i=1}^{p}$, and $\boldsymbol{u}_i \in \mathbb{R}^N$.
- Make it zero-mean $\tilde{\boldsymbol{u}}_i = \boldsymbol{u}_i - \bar{\boldsymbol{u}}$,
- Get covariance matrix C $\boldsymbol{C} = \frac{1}{p-1} \tilde{\boldsymbol{U}}^T \tilde{\boldsymbol{U}} \in \mathbb{R}^{N \times N}$
- And get eigenvectors (modes) $\boldsymbol{C}\boldsymbol{\phi}_k = \lambda_k \boldsymbol{\phi}_k$

- Eigenvectors/modes Φ can be seen as fluctuations around the mean flow

Method – Loop Guess

- And then guesses of loops L are generated as:

$$\boldsymbol{L}(\boldsymbol{x}, s) = \bar{\boldsymbol{u}} + \sum_{k=1}^{N} a_k(s, \{X_{m,k}\})\phi_k(\boldsymbol{x})$$

  - Where statistical properties (mean and covariance) are retained

$$\mathbb{E}_{X,s}[\boldsymbol{L}] = \bar{\boldsymbol{u}}$$

$$\mathrm{cov}_{X,s}(\boldsymbol{L}) := \boldsymbol{C}^{(\boldsymbol{L})} = \boldsymbol{C}$$

- Some more details:

$$a_k(s, A_{:,k}, B_{:,k}) = \sum_{m=0}^{M} \alpha_m [A_{m,k}\cos(ms) - B_{m,k}\sin(ms)]$$

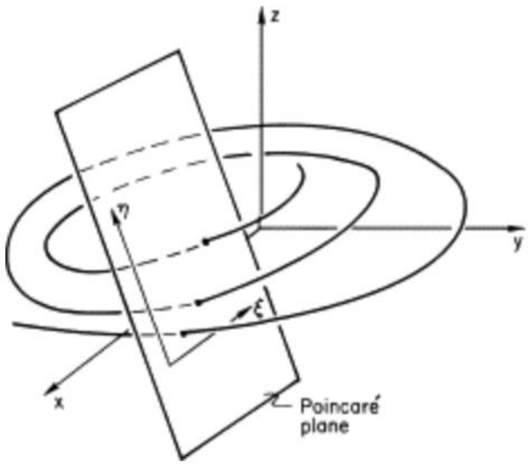
$$A_{:,k}, B_{:,k} \sim \mathcal{N}\left(0, \lambda_k \left(\sum_{m=0}^{M} \alpha_m^2\right)^{-1}\right)$$

Method – Loop Guess

- Here, larger m means higher frequency term, which will tend to generate longer guesses (extra ‘twists’ or ‘turns’)

$$\boldsymbol{L}(\boldsymbol{x}, s) = \bar{\boldsymbol{u}} + \sum_{k=1}^{N} a_k(s, \{X_{m,k}\})\phi_k(\boldsymbol{x})$$

$$a_k(s, A_{:,k}, B_{:,k}) = \sum_{m=0}^{M} \alpha_m[A_{m,k}\cos(ms) - B_{m,k}\sin(ms)]$$



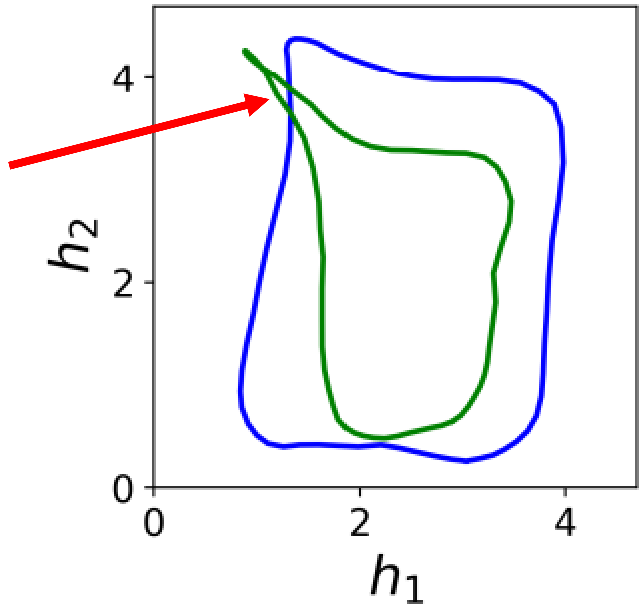
- They verify that M = p, where p is the # of intersections in Poincare sections.
- Basically, larger M (p) means longer UPO guess.

Method – Optimizer

- It uses adjoint solver + newton optimizer to converge the cost J.
- Dynamical system: $\mathbf{u}(t) = \boldsymbol{f}^t(\mathbf{u}_0) = \mathbf{u}_0 + \int_0^t \boldsymbol{F} dt'$
- Period T: $\boldsymbol{f}^T(\mathbf{u}) - \mathbf{u} = \mathbf{0}$

- Then by rescale: $\tilde{\mathbf{u}}(\mathbf{x}, s) := \mathbf{u}(\mathbf{x}, sT).$
- And combine equations, we get residual vector r: $\boldsymbol{r} = \boldsymbol{F}(\tilde{\mathbf{u}}) - \frac{1}{T}\frac{\partial \tilde{\mathbf{u}}}{\partial s}$
- And cost J: $J := \int_0^1 \int_{\mathcal{X}} \boldsymbol{r} \cdot \boldsymbol{r} \, d\boldsymbol{r} ds$

Method – Latent Gluing of UPOs



- It glues two orbits in latent space to have a longer guess.

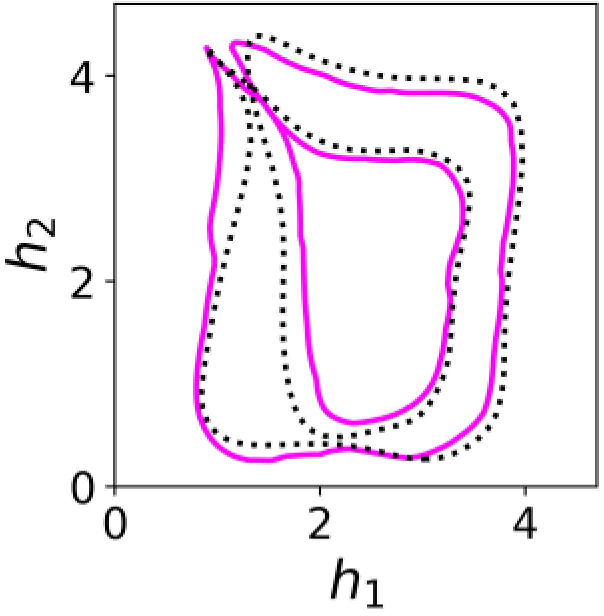
- First find time steps I,J where two UPOs are closest

$$I, J = \arg\min_{i,j} ||\boldsymbol{L}_1^{(i)} - \boldsymbol{L}_2^{(j)}||_2$$

- Then glue them:

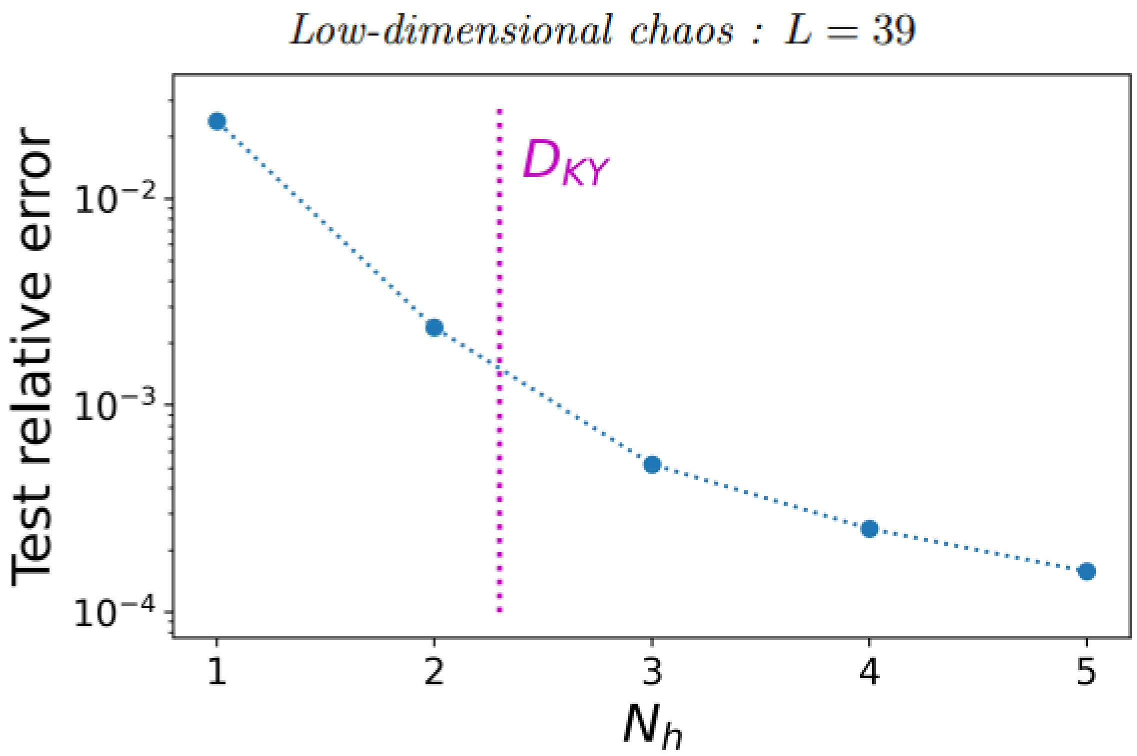
$$\boldsymbol{G}_0 = \begin{pmatrix} \boldsymbol{L}_1^{(1:I)} \\ \boldsymbol{L}_2^{((J+1):end)} \\ \boldsymbol{L}_2^{(1:J)} \\ \boldsymbol{L}_1^{((I+1):end)} \end{pmatrix}$$

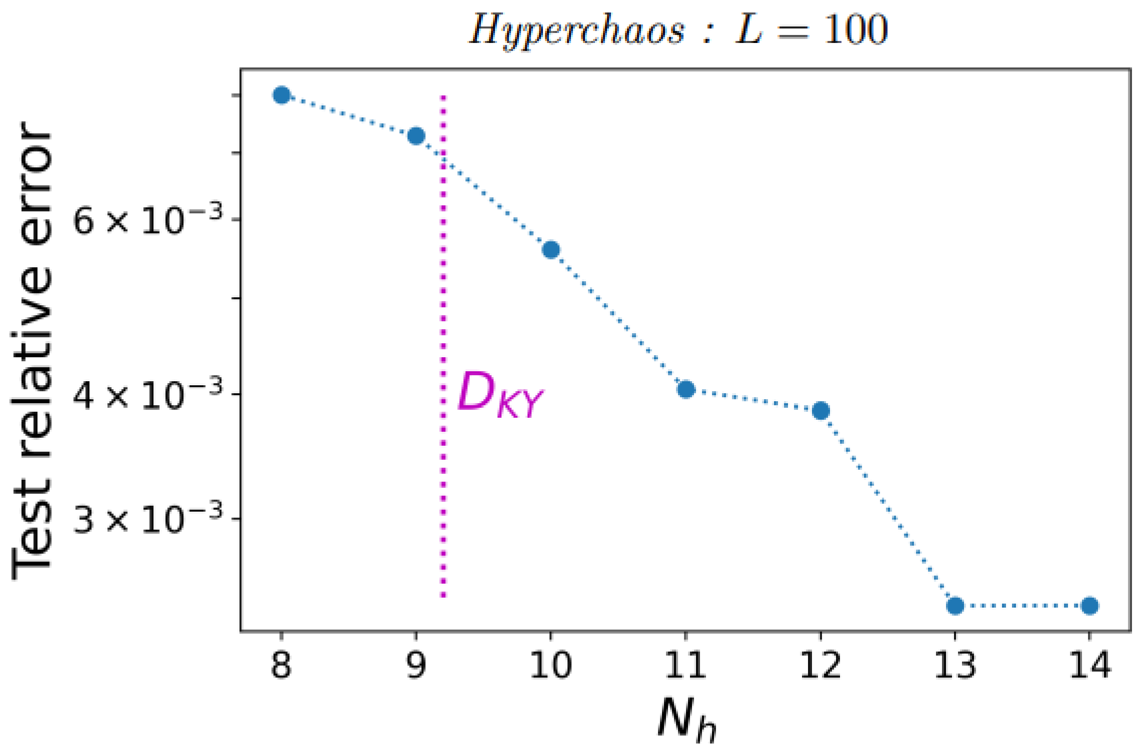
- Finally, to solve the discontinuity, they set high frequency modes to zero keep only the lowest 1/6 positive



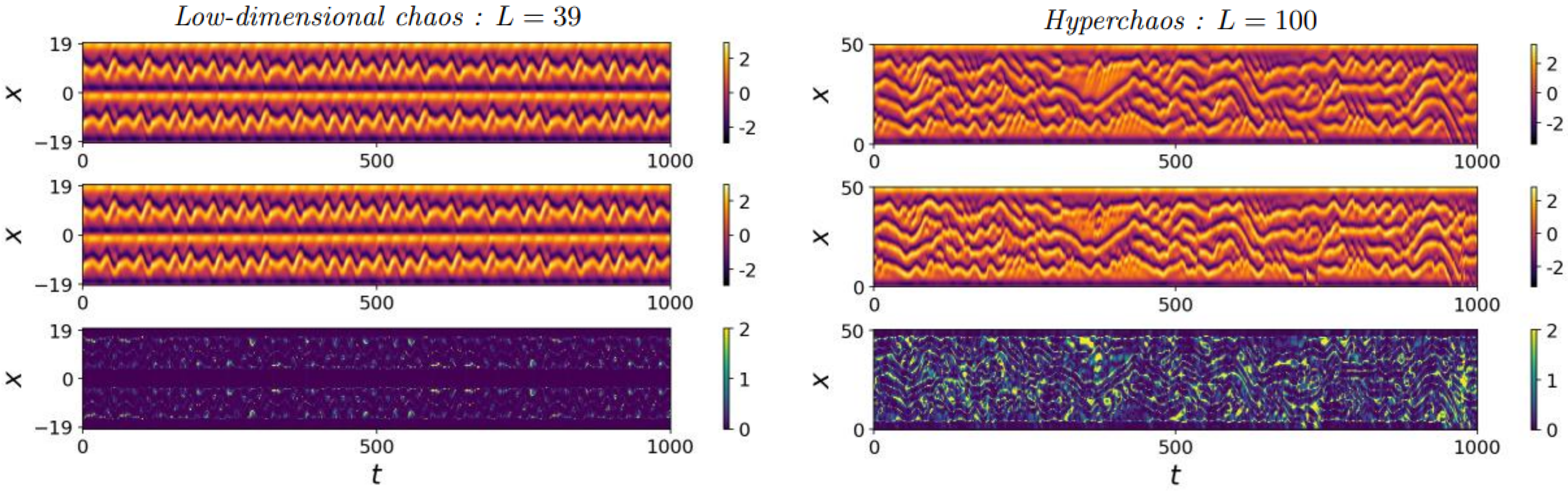
Result – Dimensionality Reduction ($N_x \rightarrow N_h$)

- D_{KY} is the dimension for chaotic attractor, the N_h should be larger than that to retain nonlinear property ($N_h > D_{KY}$)



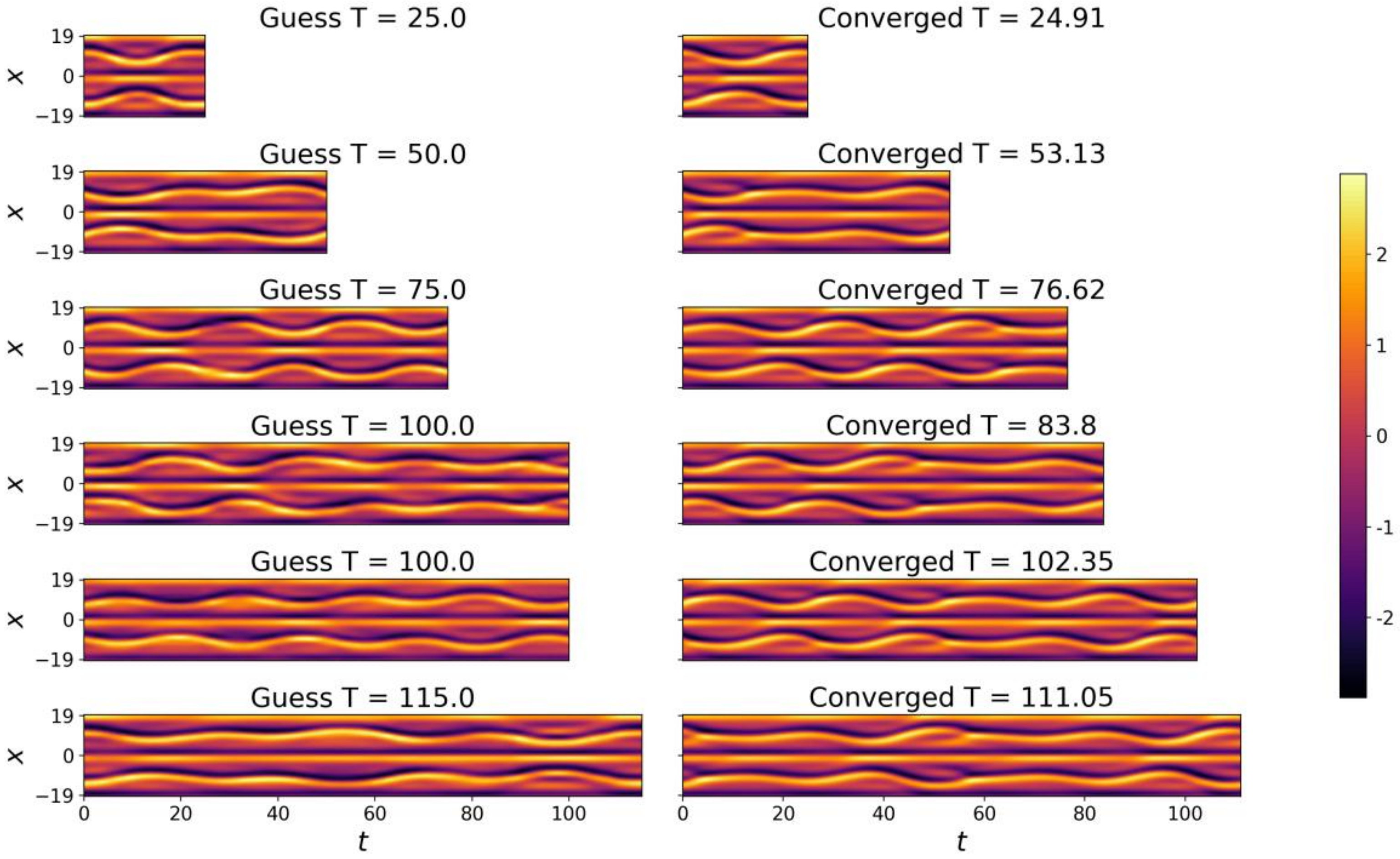


Result – Dimensionality Reduction

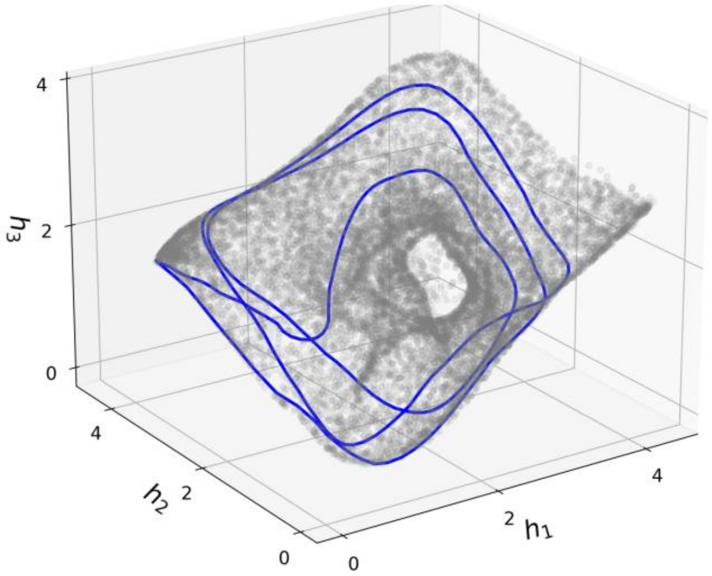


- Top: Original phase space
- Middle: Autoencoder output
- Bottom: Their difference

Result – Loops Guessing (L = 39)



Example of guesses and converged ones



Example of a UPO

Result – Loops Guessing (L = 39)

Type	Count	Percentage	p
24.91	35	17.5	1
25.37	104	52.0	1
No convergence	6	3.0	
Fixed points	55	27.5	

70%

Optimizers get stuck in a local minimum, or require more time to converge

Guessed period 25

Type	Count	Percentage	p
24.91	35	7.0	1
25.37	20	4.0	1
50.37	78	15.6	2
52.04	91	18.2	2
53.13	157	31.4	2
57.23	1	0.2	2
No convergence	48	9.6	
Fixed points	70	14.0	

76%

Guessed period 50

p = 1 reappear when the loop converges to a double periodic orbit

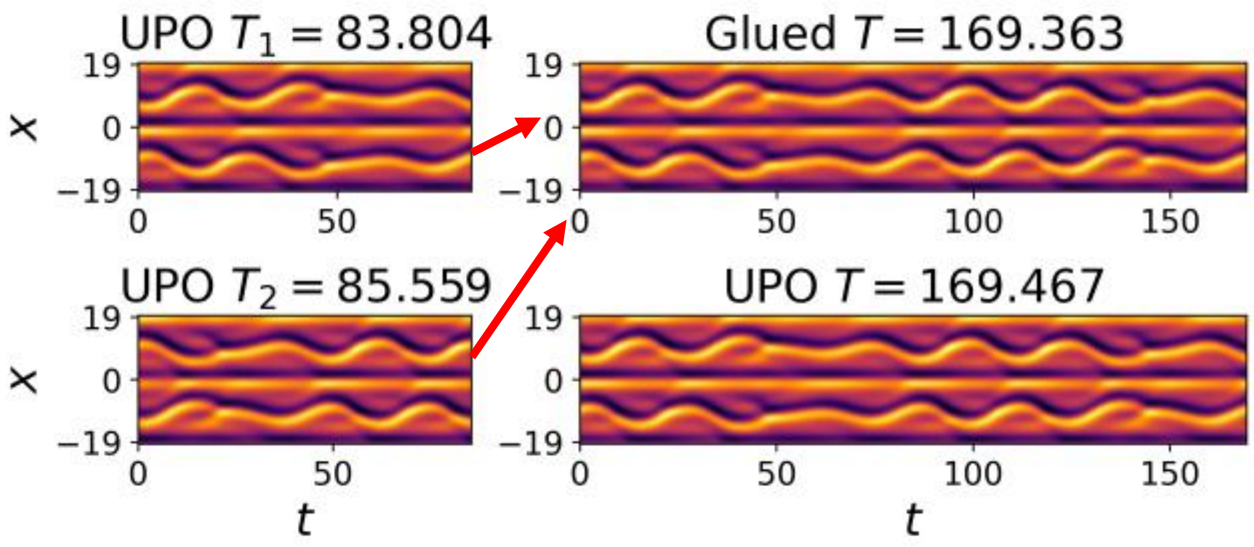
Type	Count	Percentage	p
24.91	15	2.1	1
25.37	4	0.6	1
50.37	3	0.4	2
52.04	2	0.3	2
53.13	3	0.4	2
57.23	23	3.3	2
57.63	20	2.9	2
75.28	35	5.0	3
75.72	18	2.6	3
75.94	16	2.3	3
76.62	40	5.7	3
76.85	30	4.3	3
76.95	38	5.4	3
77.37	16	2.3	3
85.54	1	0.1	3
No convergence	385	55.0	
Fixed points	51	7.3	

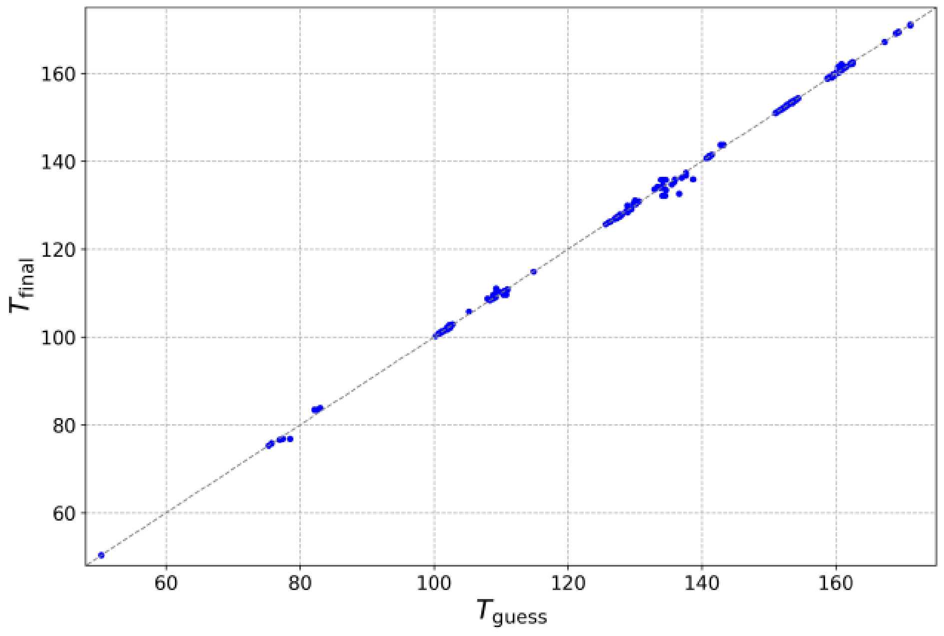
38%

Guessed period 75

p = 1 reappear when the loop converges to a triple periodic orbit

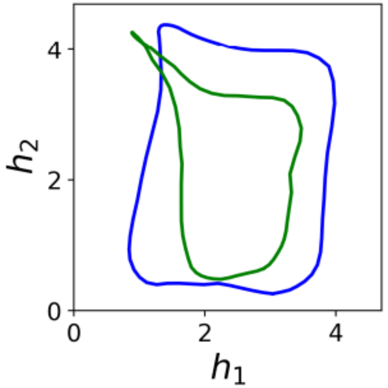
Result – Loops Glue Guessing (L = 39)





$T_{guess} = T_1 + T_2$

Result – Loops Glue Guessing (L = 39)



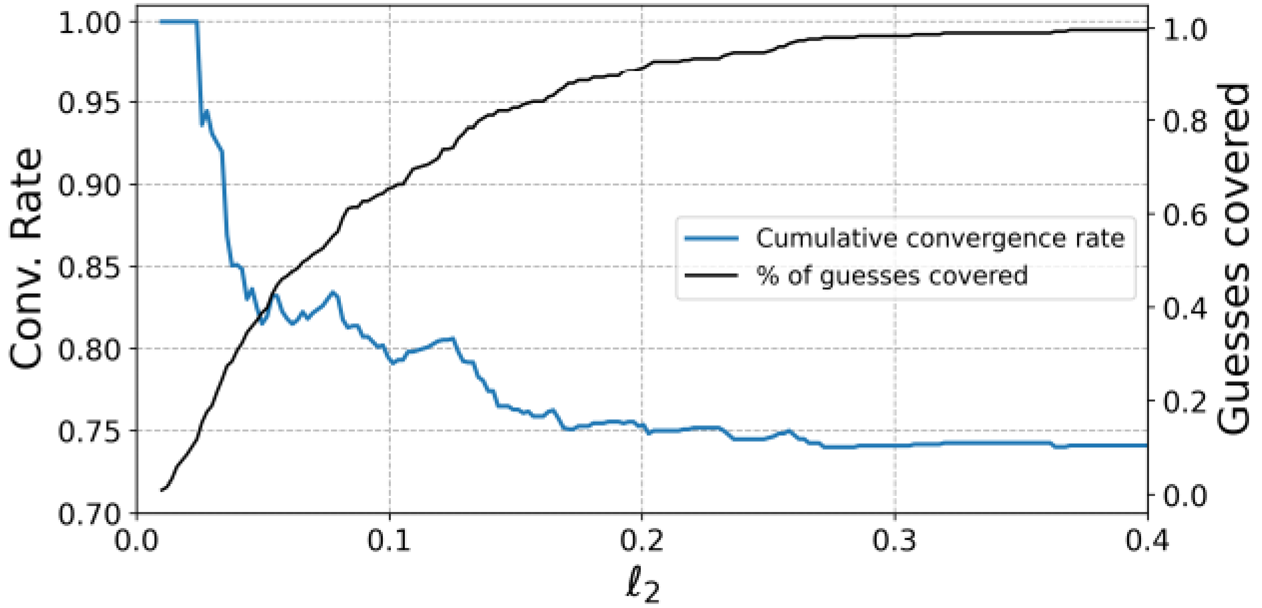


FIG. 12. Left axis: Cumulative convergence rate of glued guesses with distance of closest passage in the latent space less than ℓ_2 (blue). Right axis: percentage of guesses with distance of closest passage in the latent space less than ℓ_2 (black).

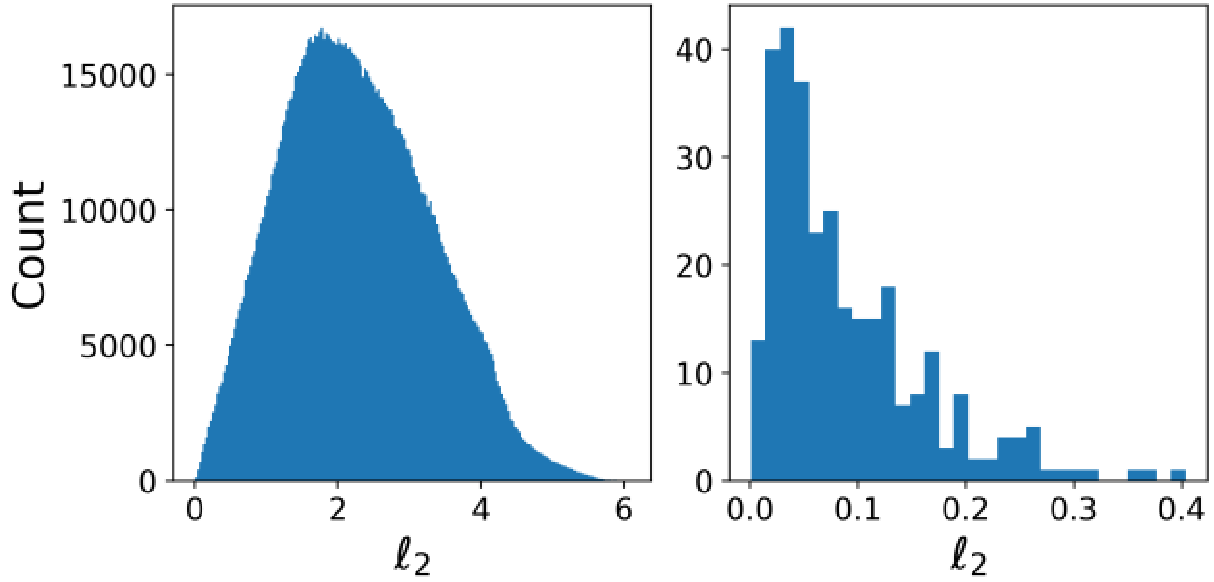
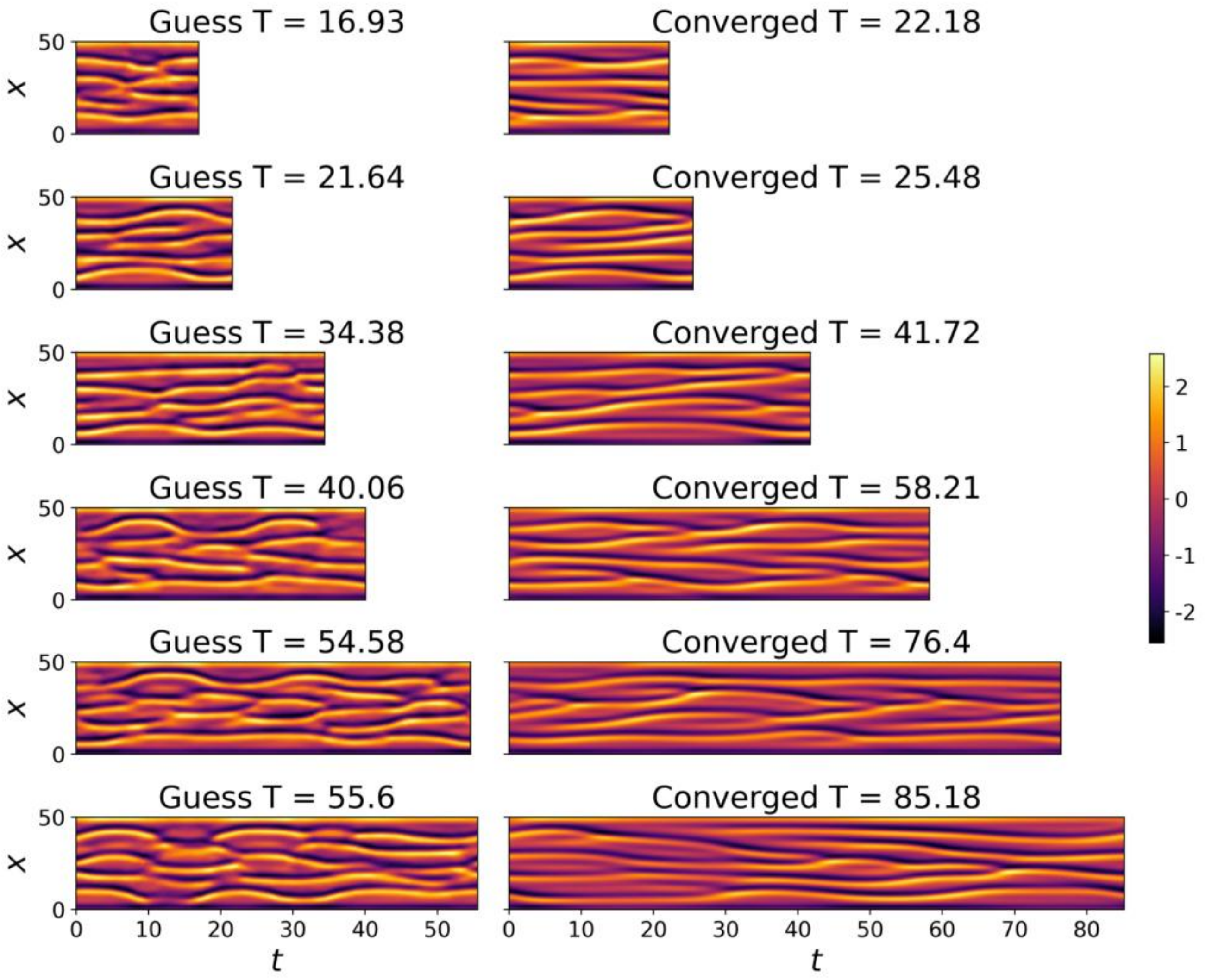
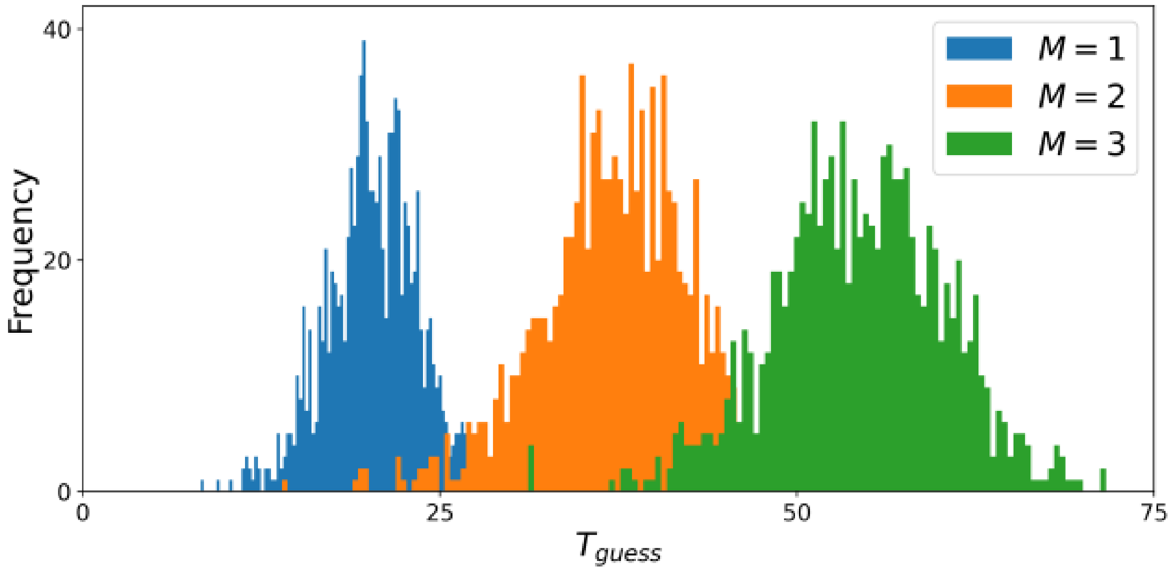


FIG. 7. Left: distribution of random ℓ_2 distances of a long time-series in latent space. Right: distribution of ℓ_2 distances between points of closest passage between UPOs with periods $T < 100$.

$$I, J = \arg\min_{i,j} ||\boldsymbol{L}_1^{(i)} - \boldsymbol{L}_2^{(j)}||_2 \quad \ell_2 = ||\boldsymbol{L}_1^{(I)} - \boldsymbol{L}_2^{(J)}||_2$$

Result – Loops Guessing (L = 100)



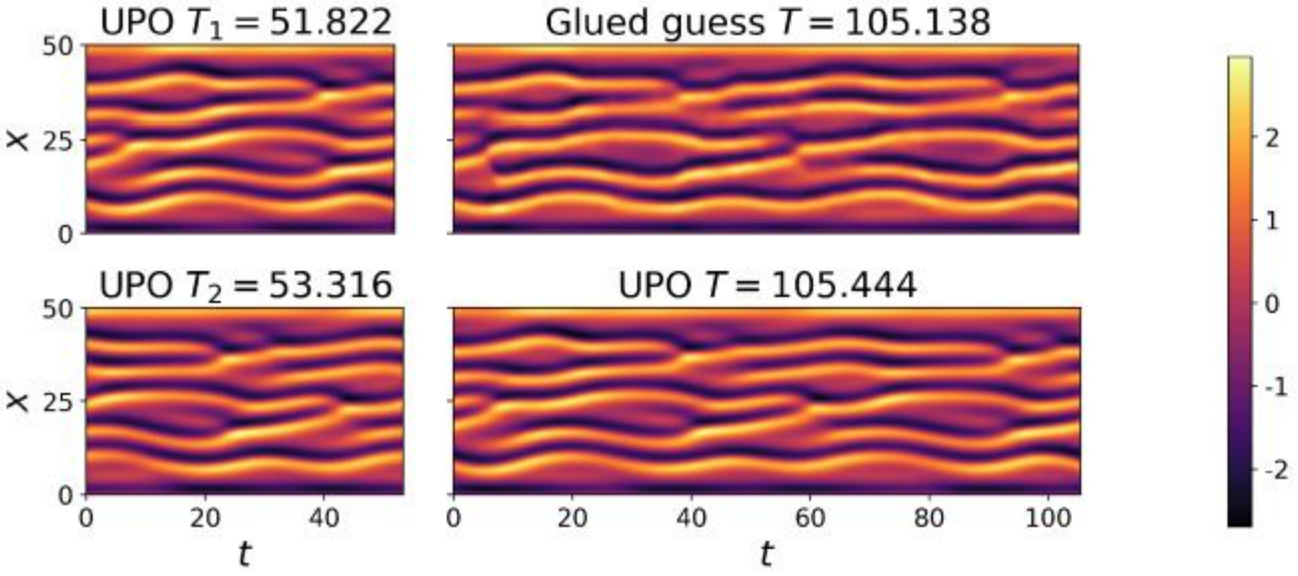


Result – Loops Guessing (L = 100)

M	1	2	3
Guesses	1,000	1,000	1,000
Fixed points	13	0	0
No convergence	834	951	989
UPOs	153	49	11

TABLE V. Summary of the main UPO searches at $L = 100$ for $M = 1, 2$ and 3. The success rate clearly drops as M increases, which may be due to multiple factors, such as the crudeness of the guess definition or stopping the convergence too early.

Result – Loops Glue Guessing (L = 100)



- **Search A**: Guesses where the initial two orbits are close in the latent space, with $\ell_2 < 0.1$, and where $T_1 + T_2 < 125$. This gives a total of 877 guesses.

- **Search B**: A random selection of 1,000 glued guesses among those with $T_1 + T_2 < 125$.

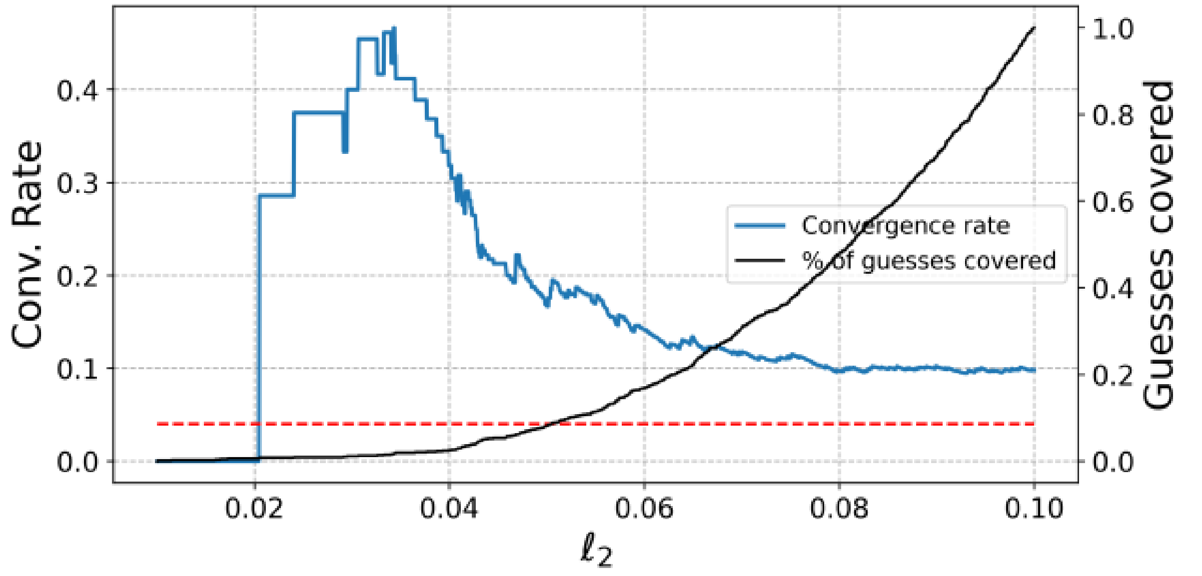


FIG. 20. Left axis: Cumulative convergence rate of glued guesses in search A with distance of closest passage in the latent space less than ℓ_2 (blue). Right axis: percentage of guesses in search A with distance of closest passage in the latent space less than ℓ_2 (black). Red dashed: Convergence rate of search B. The convergence rate for closer orbits is noticeably larger than for random orbits.

Conclusion

- It introduces a new method for generating initial guesses for UPOs by randomly drawing loops in the low-dimensional latent space defined by an autoencoder.
- The convergence rate performs well in low-dimensional chaos and in hyperchaos for small-M.
- The gluing of UPOs is successful and points towards a hierarchy of UPOs where longer UPOs shadow shorter ones.