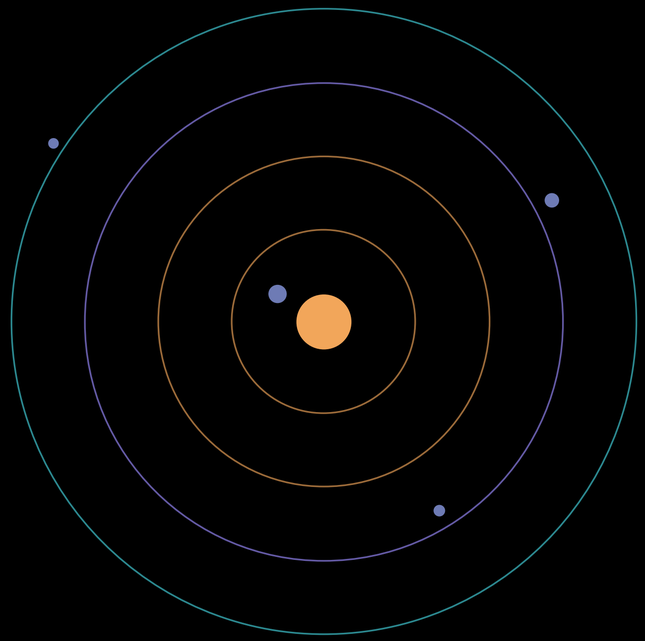
Artificial Intelligence
Institute for Cosmic Origins
Funded by the U.S. National Science Foundation
and the Simons Foundation



Reduced Order Modeling of Parametric Quadratic Chemical Kinetics

Arjun Vijaywargiya

George Biros

TEXAS
The University of Texas at Austin

SIMONS FOUNDATION

NSF

The setting: parametric quadratic dynamical systems

ODEs describing mass-action kinetics

$$\dot{x}(t) = A(p(t)x(t) + B(p(t))(x(t), x(t))$$
$$x(0) = x_0(p(0))$$

- **State** $x(t) \in \mathbb{R}^N$ — chemical abundances of N species
- **Linear part** $A(p)$ — matrix containing unimolecular reaction rates

$k = \alpha\zeta$

$\mathbf{H_2 + H_2 + cr \longrightarrow H_3^+ + H + e^-}$

$k = \alpha\, G_0 \exp(-\gamma A_V)$

$\mathbf{hv + HCO^+ \longrightarrow CO + H}$

The setting: parametric quadratic dynamical systems

ODEs describing mass-action kinetics

$$\dot{x}(t) = A(p(t)x(t) + B(p(t))(x(t), x(t))$$
$$x(0) = x_0(p(0))$$

- **State** $x(t) \in \mathbb{R}^N$ — chemical abundances of N species
- **Linear part** $A(p)$ — matrix containing unimolecular reaction rates
- **Quadratic part** $B(p)$ — tensor containing bimolecular reaction rates
- **Quantity of interest** $q(t) = C\,x(t)$ — a few key species (e.g. CO, e^-)

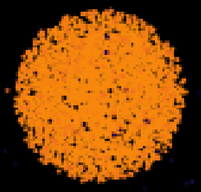
$k = \alpha\,(T/300)^{\beta}\exp(-\gamma/T)\,n_H$

$H_2 + CH_2 \longrightarrow CH_3 + H$

A Full Order Model solve is expensive

FOM: An implicit time-integrator that solves the ODE system.

- FOM is expensive for several reasons:
  - High dimensionality: large number of species and reactions.
    - KIDA gas phase network: 578 species and 8275 reactions.
  - Stiffness: disparate timescale.
    - Need implicit time integrators.
  - Non-linear and parametric: Newton solves.
    - The Jacobian needs to built and factored every time $p(t)$ changes.
  - Large number of parameter trajectories.



What is a reduced-order model?

A linear autoencoder with projected dynamics.

encode U^T decode U

Full state

$x \in \mathbb{R}^N$ (N species)

→

Latent coefficients

$\hat{y} \in \mathbb{R}^r$, $r \ll N$

→

Reconstruction

$\hat{x} = U\hat{y} \approx x$

LATENT DYNAMICS (Galerkin projection)

$$\dot{\hat{y}} = \hat{A}\hat{y} + \hat{B}(\hat{y}, \hat{y})$$

PROJECTED OPERATORS — size r, r×r, r×r×r

$$\hat{A} = U^T A\, U, \qquad \hat{B} = U^T \hat{B}(U\,\cdot, U\,\cdot)$$

POD BASIS — truncated SVD of snapshots

$$\left|\left|\tilde{X} - U_r U_r^T \;\tilde{X}\right|\right|_F^2 = \Sigma_{i>r}\sigma_i^2$$

- **Offline:** **run the expensive model, collect snapshots, build U**
- **Online:** **integrate an r-dimensional ODE; cost independent of N**

vs. autoencoders

Encoding/decoding is via one orthonormal matrix U (POD = optimal linear AE, via SVD)

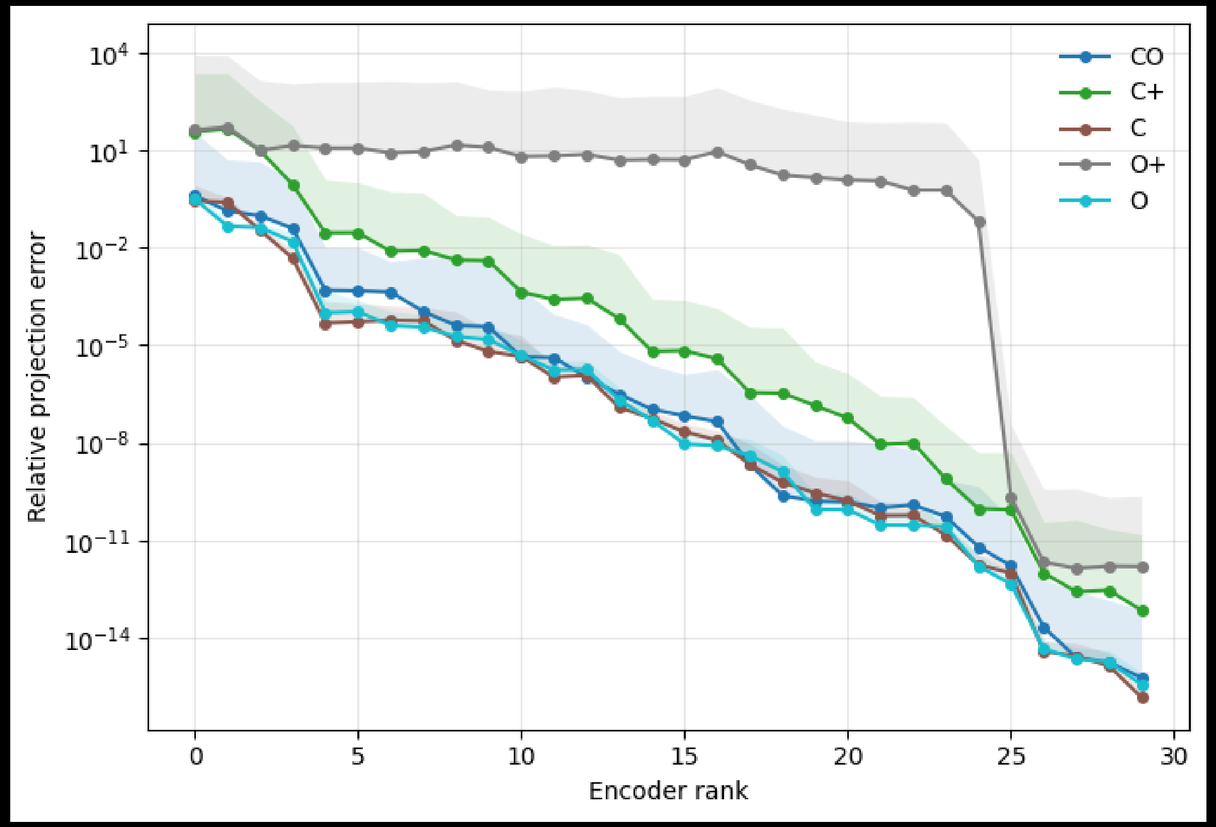
Dynamics evolve in latent space by projecting the ODE itself.

Solve the physics in the latent space: project the ODE onto span(U) and integrate r equations instead of N.

Parametric ROMs

Several challenges

1. Slow Kolmogorov n-width decay associated with a global basis.
   - Need parameter calibrated local bases.
2. Disparate scales.
   - Need parametric, component wise scaling
3. Operators need to be queried and projected, adding to online cost.
   - Affine expansions, Taylor expansions.



Global basis: Median Projection error vs rank

$$A(\mu) \approx A_1\mu_1 + A_2\mu_2 + A_3\mu_3$$

$$A(\mu) \approx A(\mu_c) + \nabla_\mu A\,(\mu - \mu_c)$$

Piecewise constant approximation

Piecewise-constant p(t) provides a constant-coefficient quadratic system in each interval.

- p(t) is provided on a discrete time grid.
- Freeze p in each interval:

$$\dot{x}_k(s) = A(p_k)x_k + B(p_k)(x_k, x_k), \qquad s \in [0, \Delta t]$$
$$x_k(0) = x_{k-1}(\Delta t),$$

Use a ROM per interval

Encode

$$\hat{y}_k = U_k^T D_k^{-1}(x_k - \bar{x}_k)$$

Latent space dynamics

$$\dot{\hat{y}}_k = \hat{c}_k + \hat{A}_k \hat{y}_k + \hat{B}_k(\hat{y}_k, \hat{y}_k)$$

Decode

$$x_k = D_k U_k \hat{y}_k + \bar{x}_k$$

- p_k denotes the parameter value in $[t_k, t_{k+1})$.
- The basis U_k, diagonal scaling D_k, and mean $\bar{x}_k$ are functions of p_k

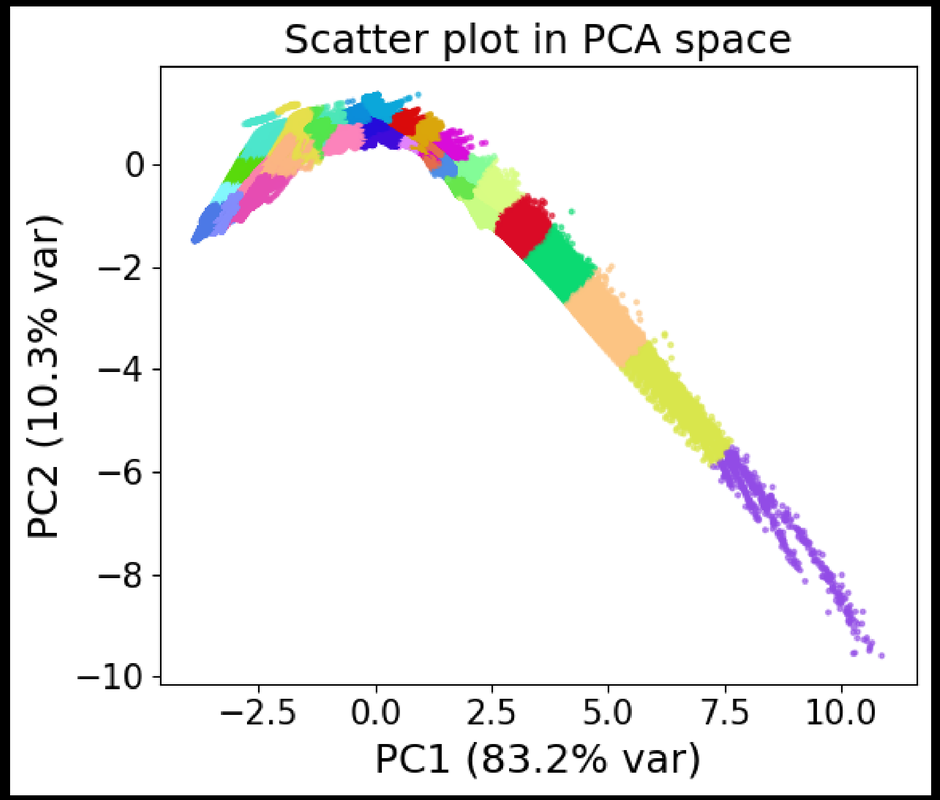
$$U_k \coloneqq U(p_k), \qquad D_k \coloneqq D(p_k), \qquad \bar{x}_k \coloneqq \bar{x}(p_k)$$

- Initial condition for y_k is $y_{k-1}(\Delta t)$.

How do we get the bases, scalings, and means?

Feature based k-means clustering.

- Need snapshot data : $\{p^i(t_j), x^i(t_j)\}$
- For fast singular value decay, need variance concentrated in a few directions.
- Do nearby parameters see similar chemistry?
  - Cluster in parameter space!
  - NO!



$$\dot{x}(t) = A\big(p(t)\big)x(t)$$

Linear system

$$\rightarrow x(t_k) = e^{A_{k-1}\,\Delta t} \,..\, e^{A_0\,\Delta t} x_0$$

History of p

What is a more appropriate feature space?

- Should clustering be done in the joint parameter + state space?
  - High dimensional with noisy features.
    - $\mathbb{R}^{N+N_p}$
- Cluster in the parameter + QoI space.
  - A compromise between parameters only and parameters + state clustering.
- Gather chemical snapshot data X_c matrix per cluster and precompute the ROM objects:
  - Mean $\bar{x}_c$, diagonal scaling D_c, and Basis U_c.

$$\bar{x}_c = \frac{1}{N_c} X_c 1$$

$$[d_c]_\ell = \sqrt{(\text{standard deviation of row } \ell)}$$

$$D_c = \text{Diag}(d_c)$$

$$\widetilde{X_c} = D_c^{-1}(X_c - \overline{x_c} 1^T)$$

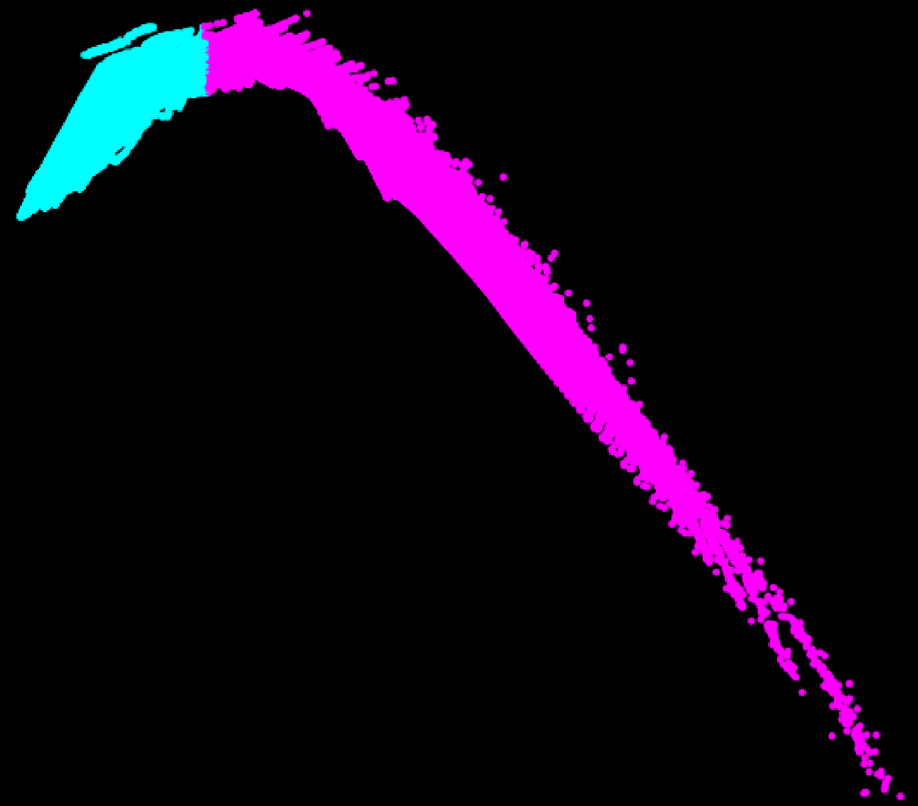
$$U_c = \Phi[:, :r], \qquad \widetilde{X_c} = \Phi \Sigma \Psi^T$$

All precomputed!

Adaptive clustering

Split a cluster if QoI error is large.

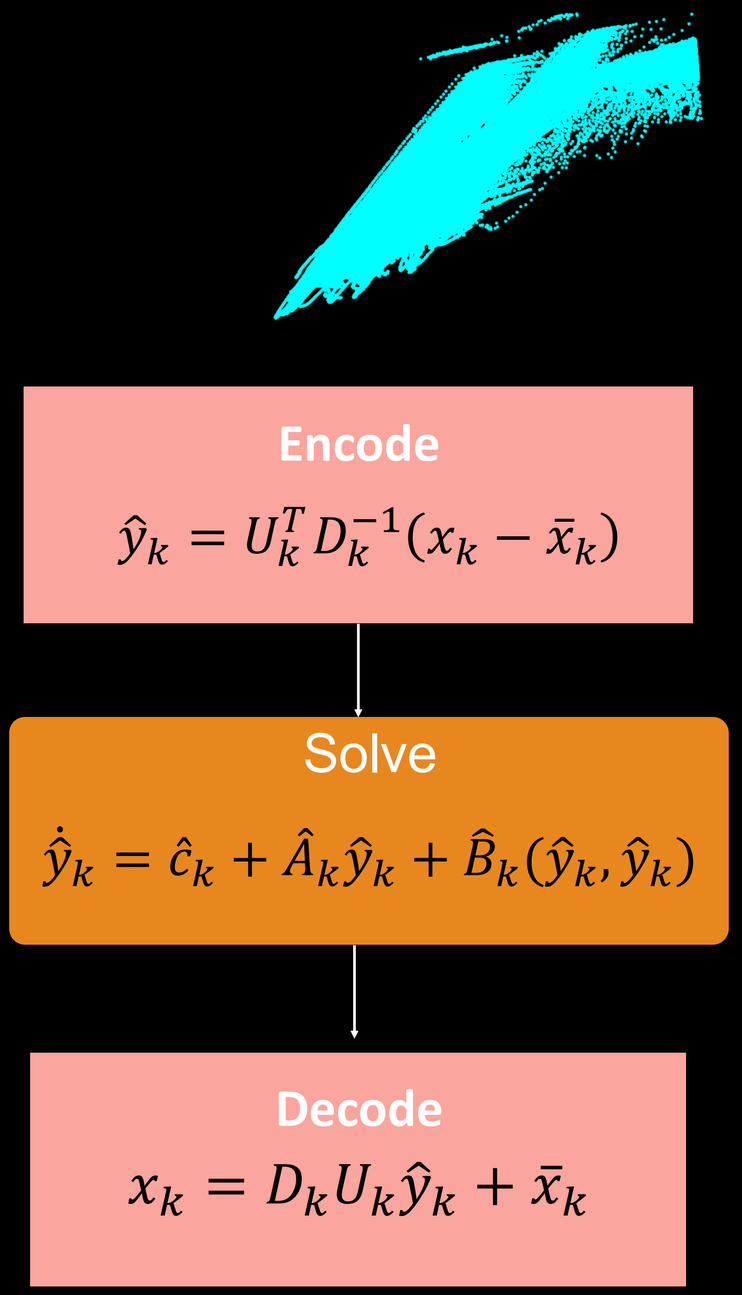
1. Start with a base clustering.



Adaptive clustering

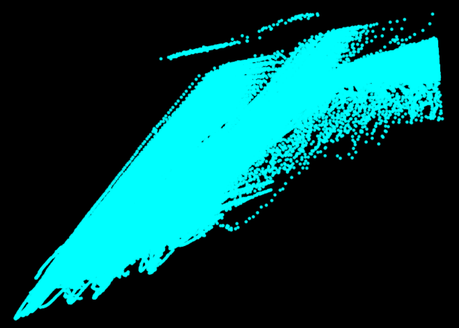
Split a cluster if QoI error is large.

1. Start with a base clustering.
2. For each cluster, compute one step solution using the local ROM.



Adaptive clustering

Split a cluster if QoI error is large.



1. Start with a base clustering.
2. For each cluster, compute one step solution using the local ROM.
3. Measure QoI error.

$$\hat{q} = C\hat{x}$$

$$e^i = \max_{1 \le l \le N\_q} \frac{\left|\left|q_l^i - \hat{q}_l^i\right|\right|_2}{\left|\left|q_l^i\right|\right|_2}$$

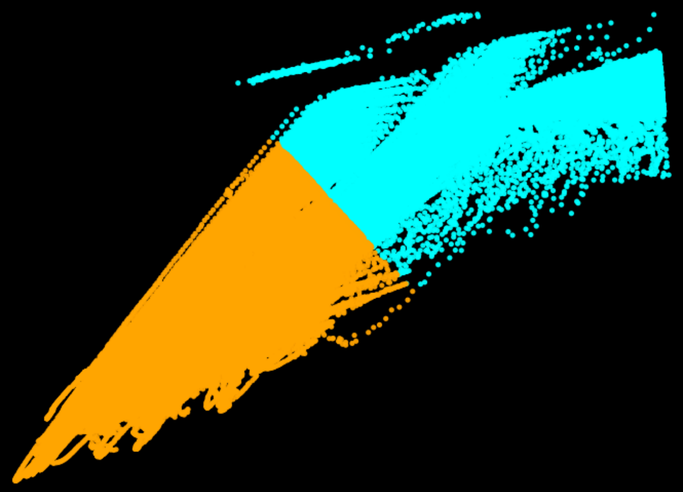
$$E_{cluster} = \frac{1}{|cluster|} \sum_i e^i$$

Adaptive clustering

Split a cluster if QoI error is large.

$$\text{If } E_{cluster} > \text{Tolerance}$$

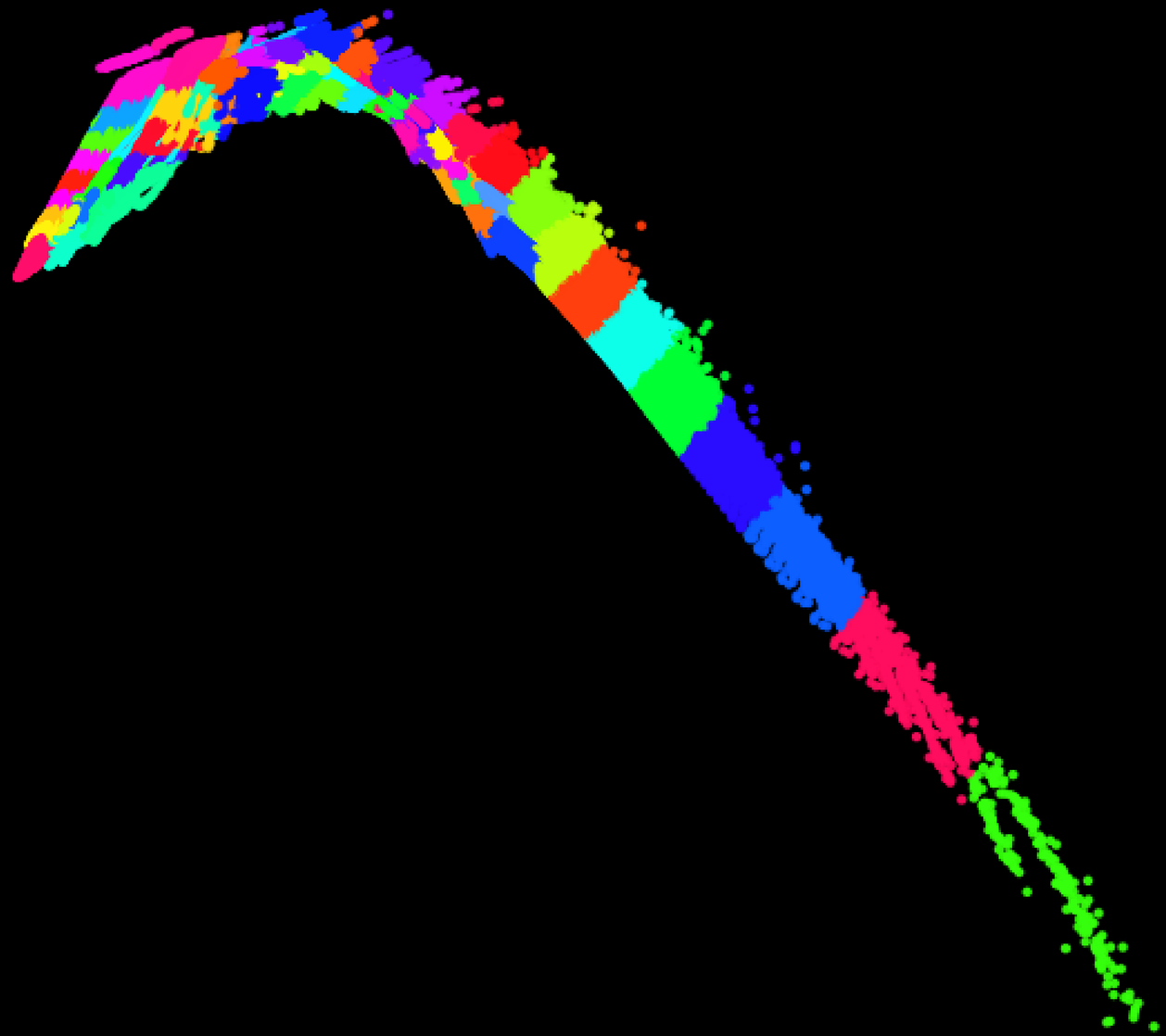
1. Start with a base clustering.
2. For each cluster, compute one step solution using the local ROM.
3. Measure QoI error.
4. If the error is large, split the cluster into two by applying k-means with k=2.



Adaptive clustering

Split a cluster if QoI error is large.

1. Start with a base clustering.
2. For each cluster, compute one step solution using the local ROM.
3. Measure QoI error.
4. If the error is large, split the cluster into two.
5. Repeat until all clusters satisfy the error tolerance or max clusters reached.



Inference

- Input: $\{p(t_k)\}_k$ QoI selector C, and $x_0 = x(p(t_0))$
- For each k,

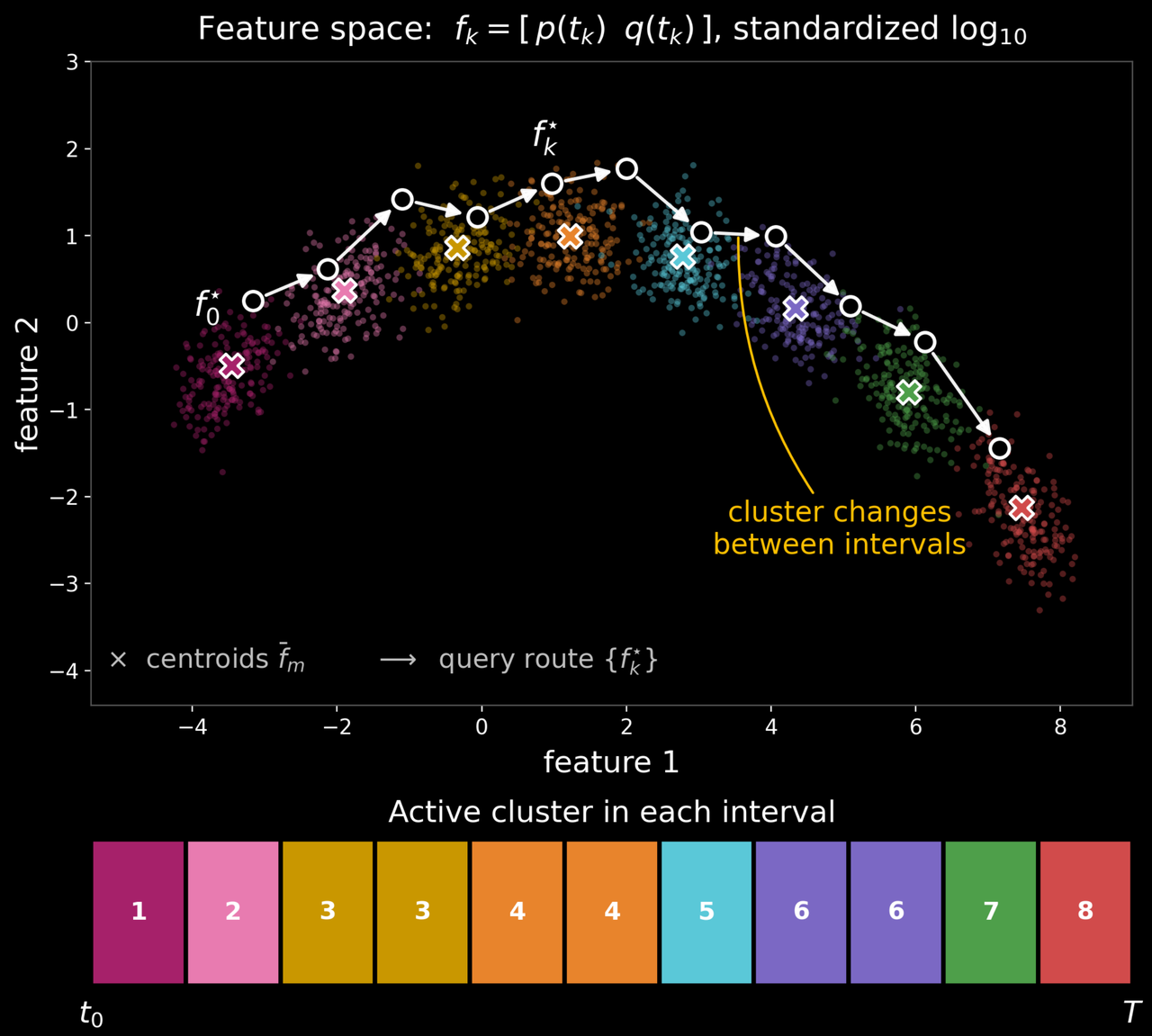
1. Route to the appropriate cluster (based on distance to the centroid)

$$m_\star = \min\{m : m \in \arg\, min_{1\le m\le M} \mathrm{Dist}(f_k, \overline{f_m})\}$$

$$f_j = [q(t_k) \quad p(t_k)]$$

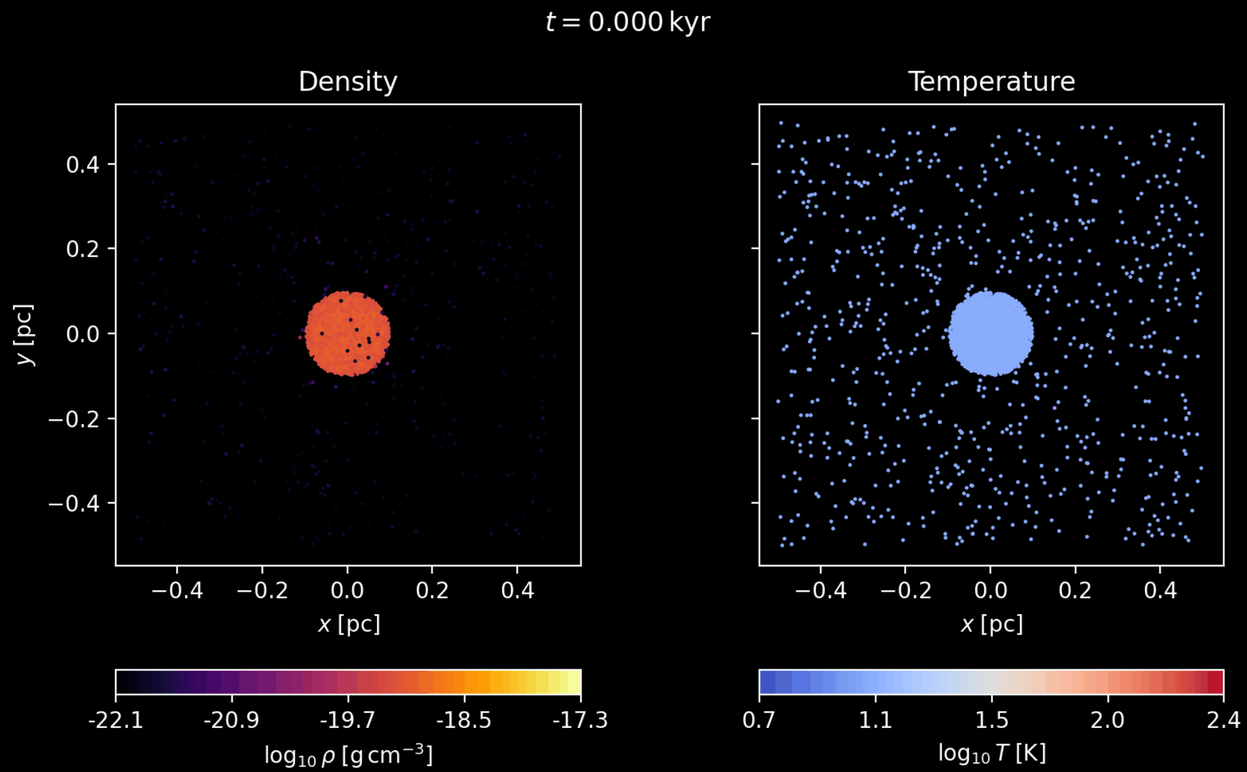
2. Query operators $A(p(t_k))$ and $B(p(t_k))$
3. Using $D_k, \overline{x_k}, U_k, c_k$, build and integrate the local ROM in $[t_k, t_{k+1})$ with the initial conditior

$$y_k(0) = U_k^T D_k^{-1}(\bar{x}_{k-1} - \bar{x}_k\} + U_k^T D_k^{-1} D_{k-1} U_{k-1} y_{k-1}(t_k)$$



Results

Hydrodynamic simulation: collapse of cold, 6 $M_\odot$ spherical cloud under self-gravity

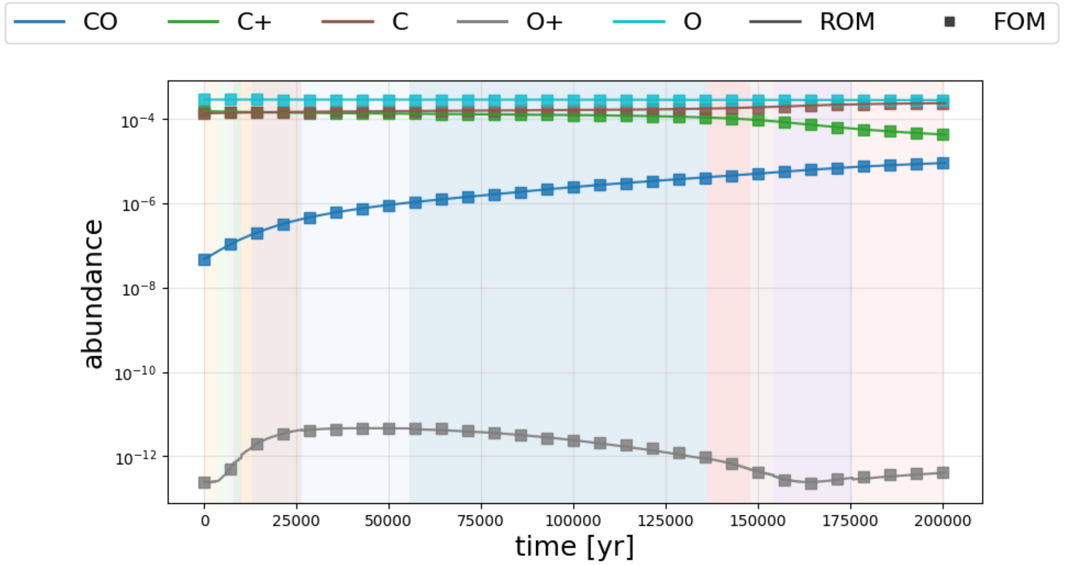


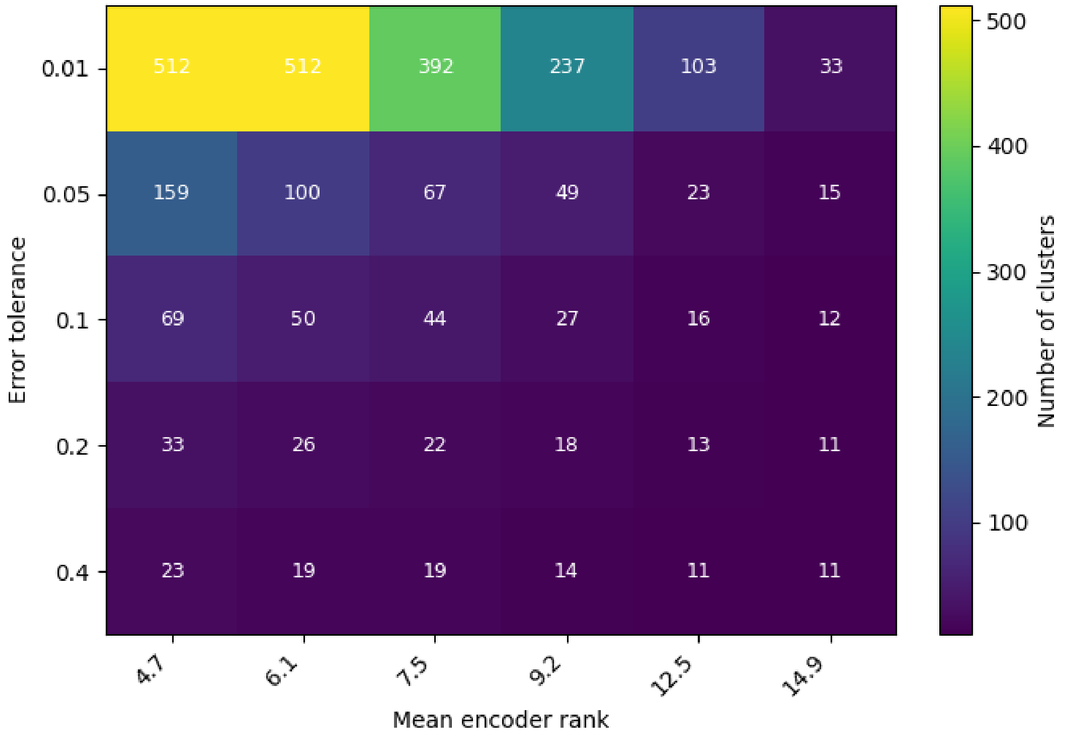
1. Nelson chemical network
   - 14 species and 23 reactions.
   - QoIs: CO and e-
   - Rates depend on 4 parameters: $[n_H, T, A_v, G_0]$
2. OSU_09_2008 chemical network
   - 29 species and 224 reactions.
   - QoIs: CO, C+, C, O+, O.
   - Rates depend only on n_H and T.

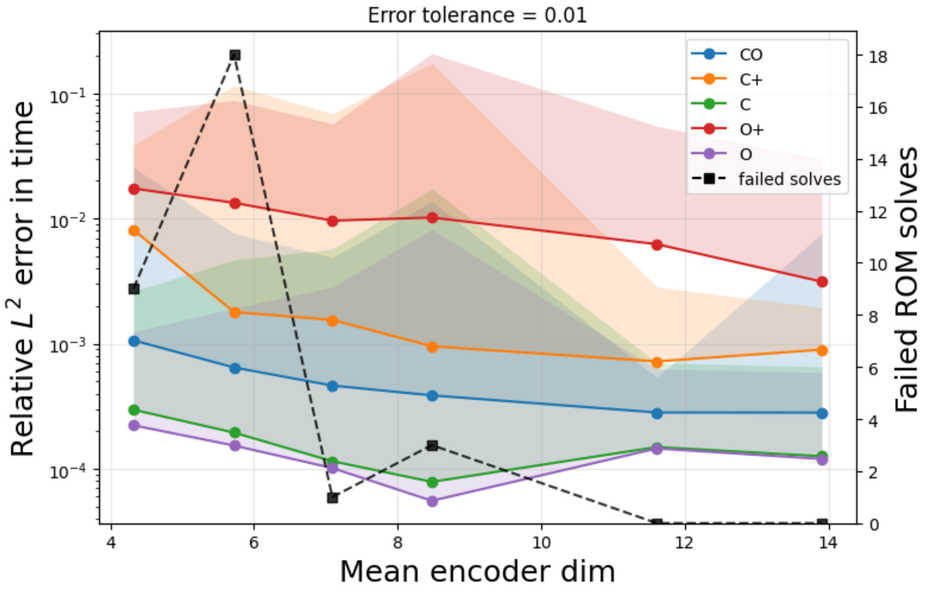
Training set: 2820 samples of p(t)

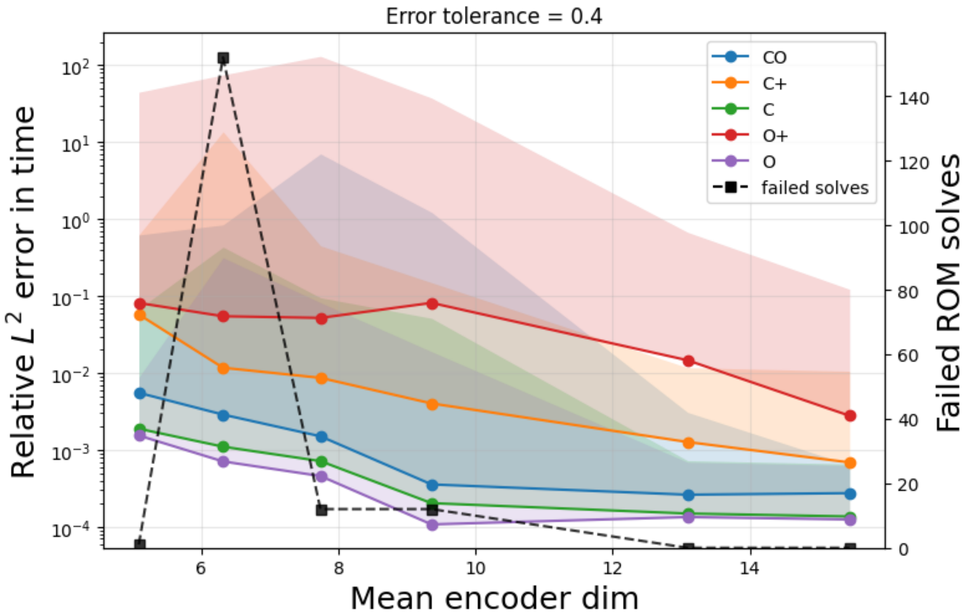
Testing set: 1410 samples of p(t)

OSU_09_2008

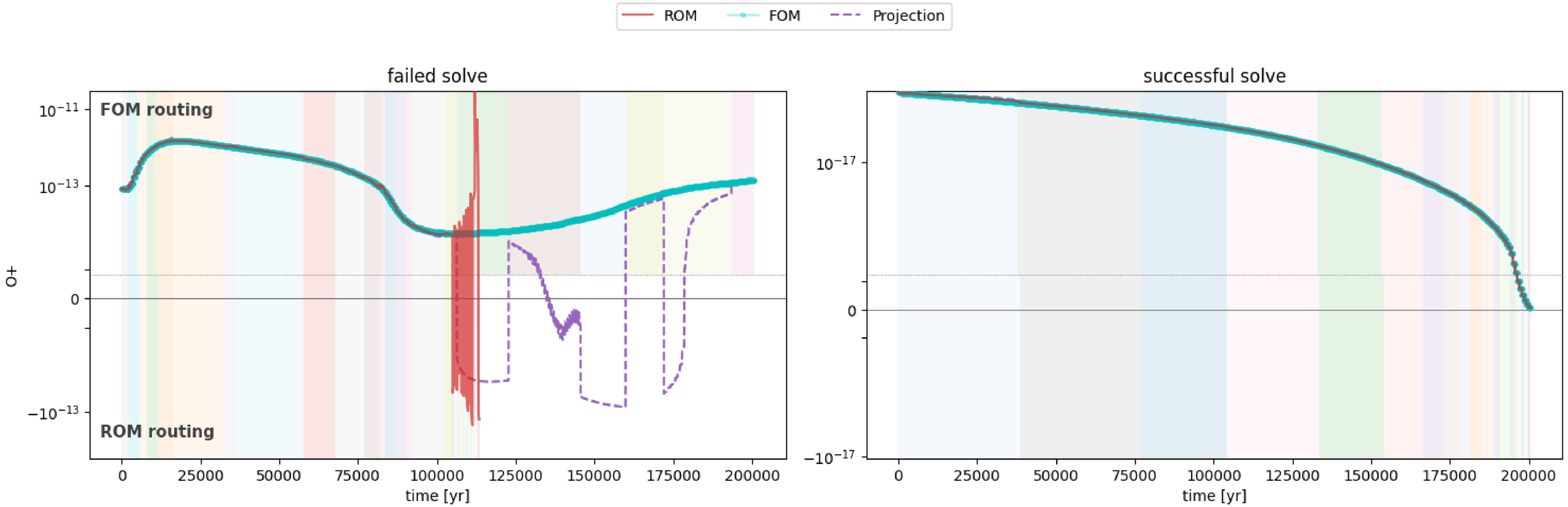

CO
C+
C
O+
O
ROM
FOM
abundance
time [yr]


Error tolerance
Mean encoder rank
Number of clusters
0.01
0.05
0.1
0.2
0.4
4.7
6.1
7.5
9.2
12.5
14.9
512 512 392 237 103 33
159 100 67 49 23 15
69 50 44 27 16 12
33 26 22 18 13 11
23 19 19 14 11 11


Error tolerance = 0.01
Relative L^2 error in time
Failed ROM solves
Mean encoder dim
CO
C+
C
O+
O
failed solves

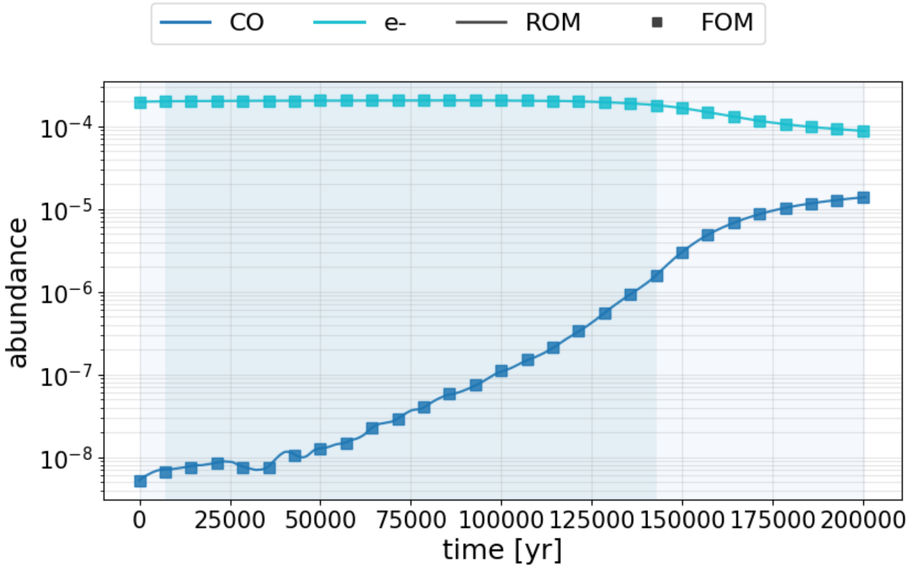

Error tolerance = 0.4
Relative L^2 error in time
Failed ROM solves
Mean encoder dim
CO
C+
C
O+
O
failed solves

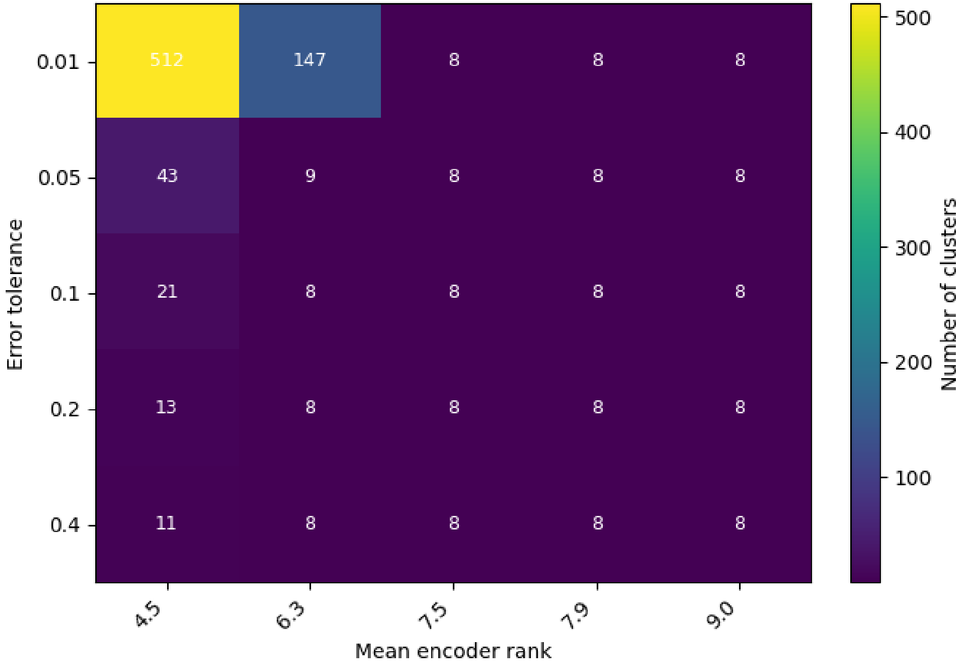
When does a solve fail?

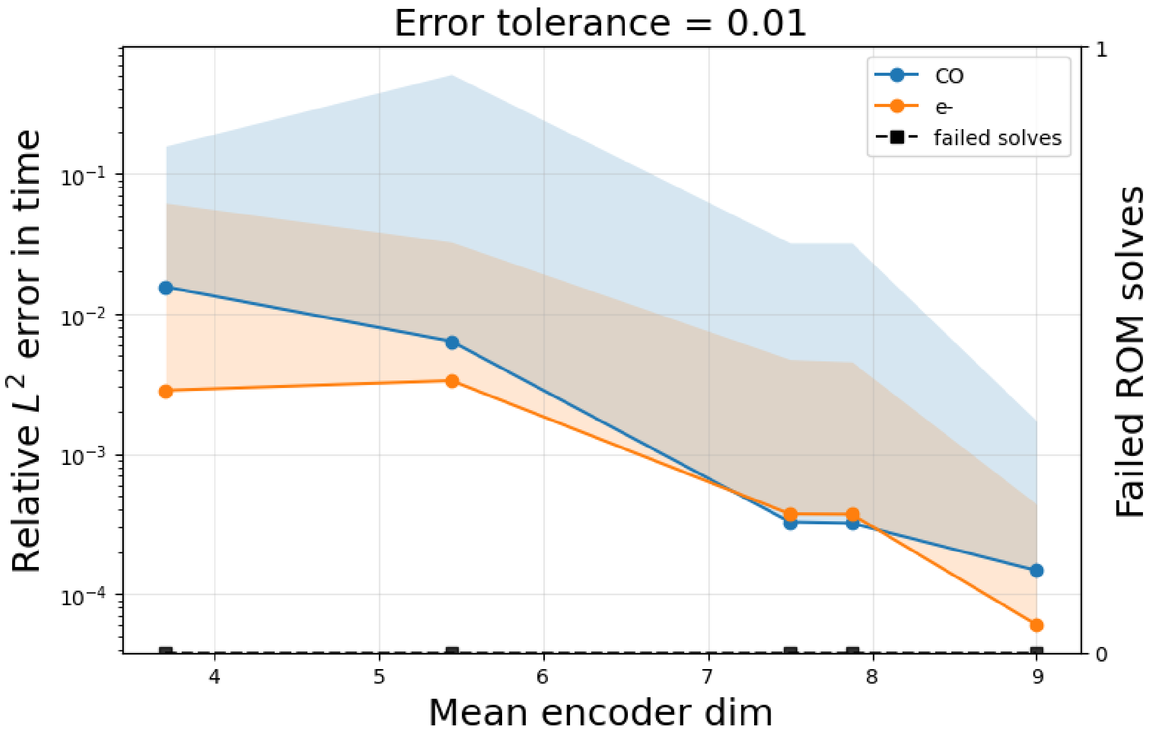


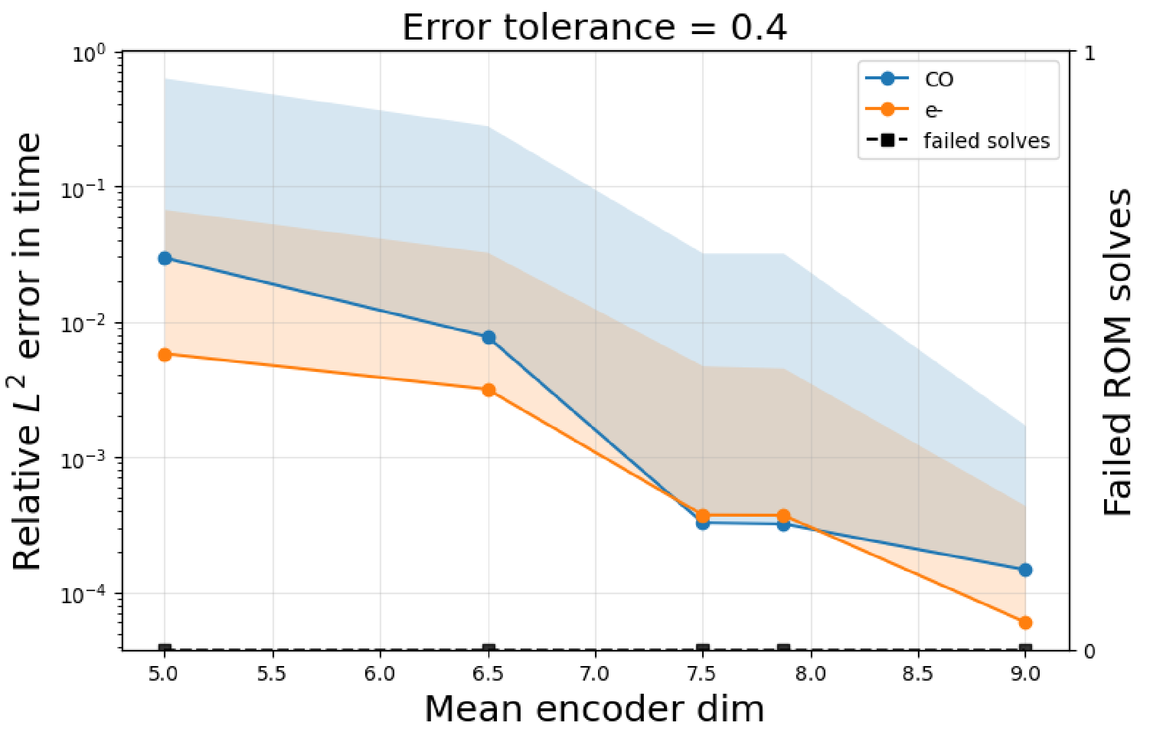
When the projected physical abundance is negative, the trajectory get routed to the wrong cluster.

Nelson


CO
e-
ROM
FOM
abundance
time [yr]


Error tolerance
Mean encoder rank
Number of clusters
0.01
0.05
0.1
0.2
0.4
4.5
6.3
7.5
7.9
9.0
512
147
43
9
21
13
11


Error tolerance = 0.01
CO
e-
failed solves
Relative L^2 error in time
Failed ROM solves
Mean encoder dim


Error tolerance = 0.4
CO
e-
failed solves
Relative L^2 error in time
Failed ROM solves
Mean encoder dim

Thanks!

Questions?

References:

- T. Grassi, F. Nauman, J. P. Ramsey, S. Bovino, G. Picogna, and B. Ercolano, *Reducing the complexity of chemical networks via interpretable autoencoders*, Astron. Astrophys., 668 (2022), A139, https://doi.org/10.1051/0004-6361/202039956.

- R. P. Nelson and W. D. Langer, On the stability and evolution of isolated Bok globules, Astrophys. J., 524 (1999), pp. 923–946, https://doi.org/10.1086/307823.

- M. R. Malik, B. J. Isaac, A. Coussement, P. J. Smith, and A. Parente, Principal component analysis coupled with nonlinear regression for chemistry reduction, Combust. Flame, 187 (2018), pp. 30–41, https://doi.org/10.1016/j.combustflame.2017.08.012.

Next steps

- Apply to the KIDA gas phase network: 578 species and 8275 reaction (ongoing).
  - Requires large basis sizes for successful ROM solves.
  - Operator assembly is expensive.
- Taylor approximation + precomputing operators.

$$A(\mu) \approx A(\mu_c) + (\nabla_\mu A)\,(\mu - \mu_c)$$

$$\mathrm{B}(\mu) \approx B(\mu_c) + (\nabla_\mu B)\,(\mu - \mu_c)$$