Conference on Decision and Control CDC2025

Distributed Lyapunov functions for nonlinear network systems

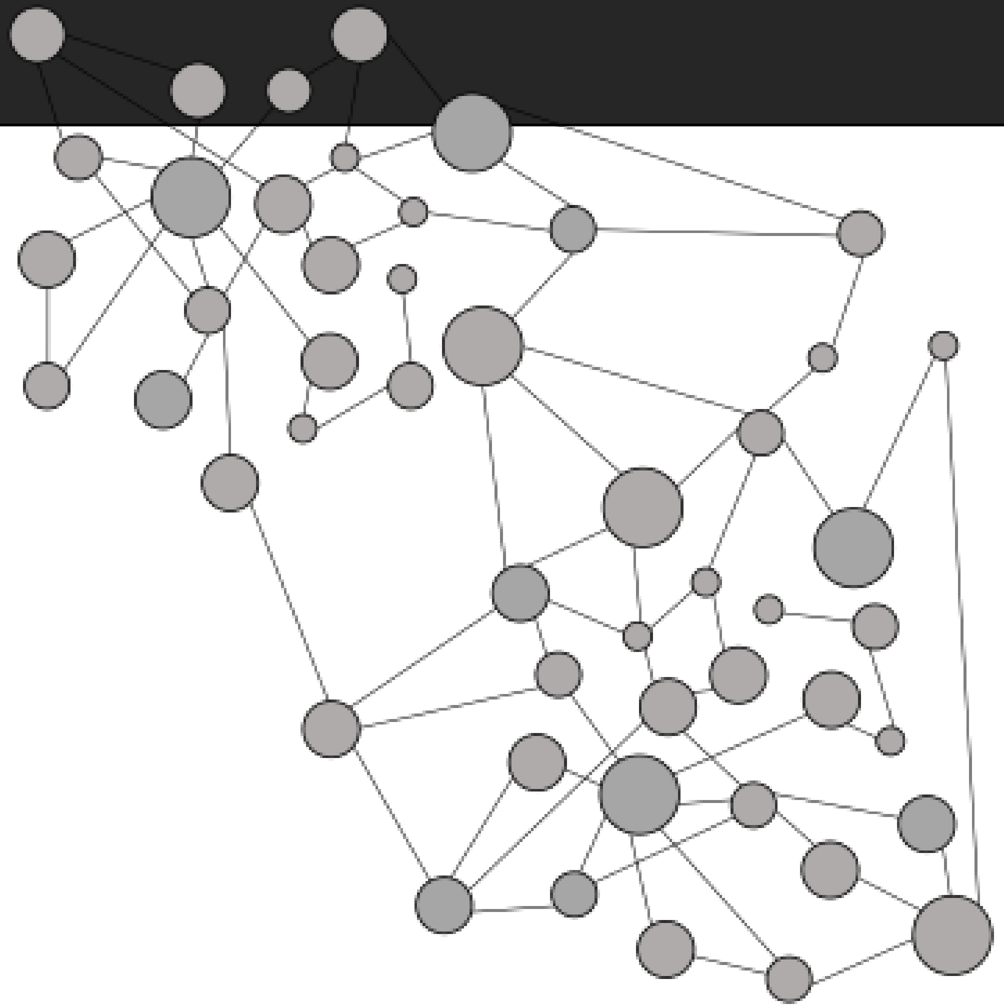
Arthur N. Montanari

Center for Network Dynamics

Northwestern University

December 12, 2025

Nonlinear network



$$\dot{\boldsymbol{x}}_i = \boldsymbol{f}_i(\boldsymbol{x}_i) + \sum_{j=1}^{N} K_{ij}\boldsymbol{g}(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad i = 1, \ldots, N,$$

nodal dynamics (self-dynamics)

coupling term (adjacency matrix K)

Nodes are m-dimensional: $\boldsymbol{x}_i = \begin{bmatrix} x_{i,1} \\ x_{i,2} \\ \vdots \\ x_{i,m} \end{bmatrix} \in \mathcal{S}^m \subseteq \mathbb{R}^m$

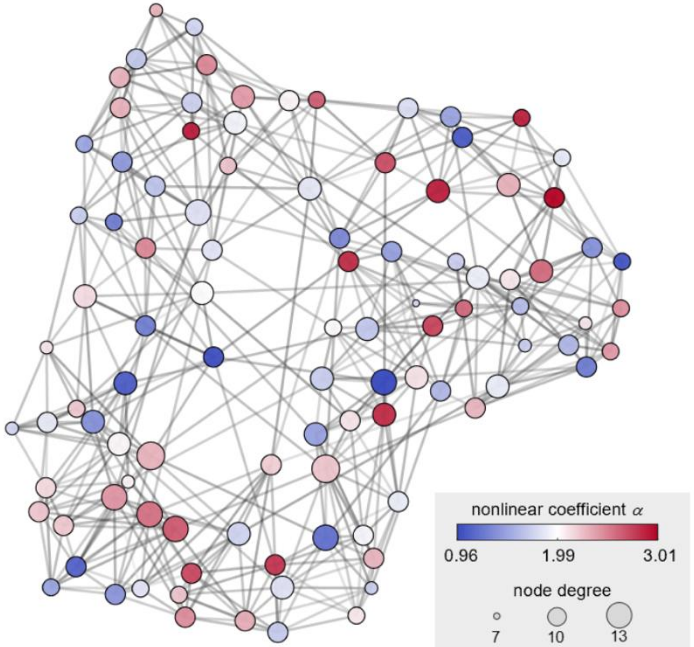
System is n-dimensional: $\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \in \mathcal{S}^n \subseteq \mathbb{R}^n$

Nonlinear network

$$\dot{\boldsymbol{x}}_i = \boldsymbol{f}_i(\boldsymbol{x}_i) + \sum_{j=1}^{N} K_{ij}\boldsymbol{g}(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad i = 1, \ldots, N,$$

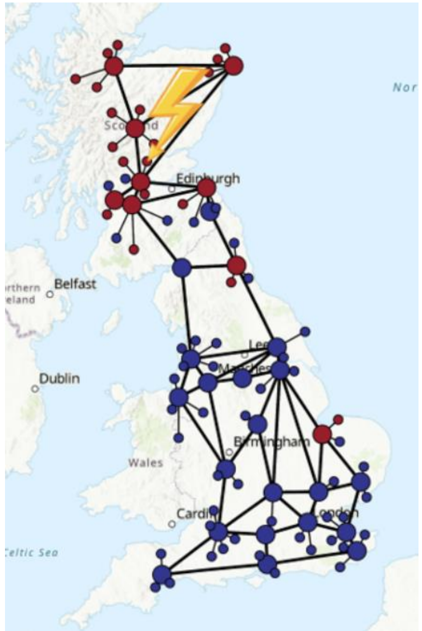
nodal dynamics (self-dynamics)

coupling term (adjacency matrix K)



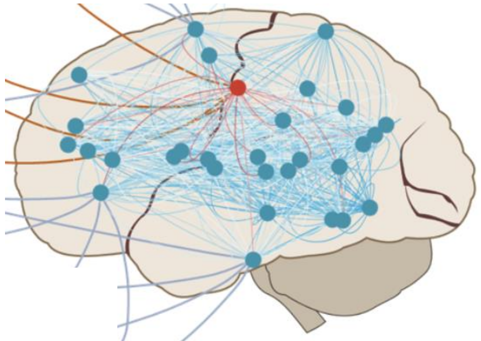
van der Pol oscillators

$$\ddot{x}_i + b_i(1 - x_i^2)\dot{x}_i + x_i = \sum_j A_{ij}(x_j - x_i)$$



power-grid model

$$\frac{2H_i}{\omega_R}\ddot{\phi}_i + \frac{D_i}{\omega_R}\dot{\phi}_i = A_i + \sum_{j=1, j\neq i}^{N} K_{ij}\sin\left(\phi_j - \phi_i + \gamma_{ij}\right)$$

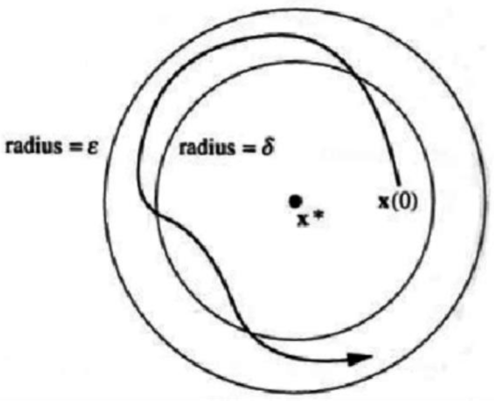


neuronal model (FitzHugh-Nagumo)

$$\dot{v}_i = v_i - \frac{v_i^3}{3} - w_i + \sum_j A_{ij}(v_j - v_i)$$
$$\tau\dot{w}_i = v_i + a - b_i w_i$$

Lyapunov function

$$\dot{\boldsymbol{x}}_i = \boldsymbol{f}_i(\boldsymbol{x}_i) + \sum_{j=1}^{N} K_{ij}\boldsymbol{g}(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad i = 1, \ldots, N,$$

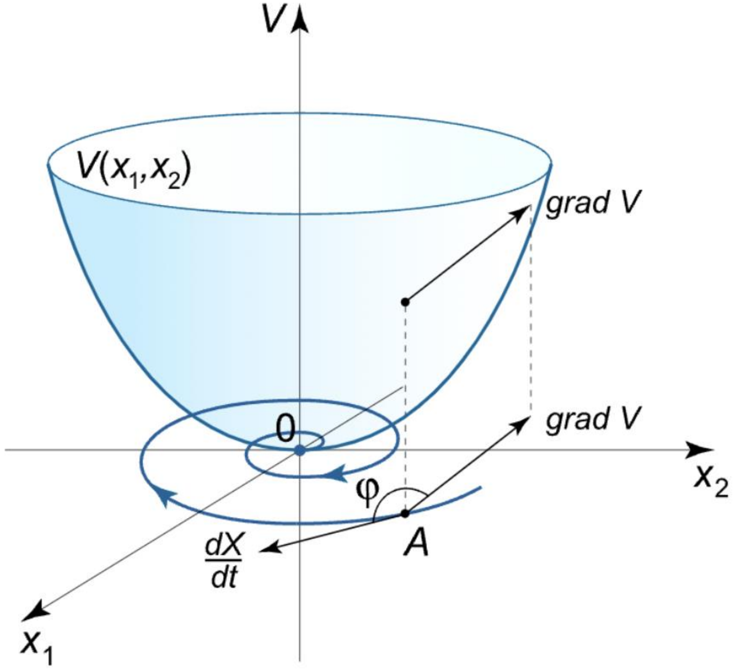


Aleksandr Lyapunov

If $V(\boldsymbol{x}) > 0, \forall \boldsymbol{x} \in \mathcal{D}\backslash\{0\}$, then $\boldsymbol{x}^*$ is Lyapunov stable.
$V(0) = 0$
$\dot{V}(x) \leq 0, \forall \boldsymbol{x} \in \mathcal{D}\backslash\{0\}$

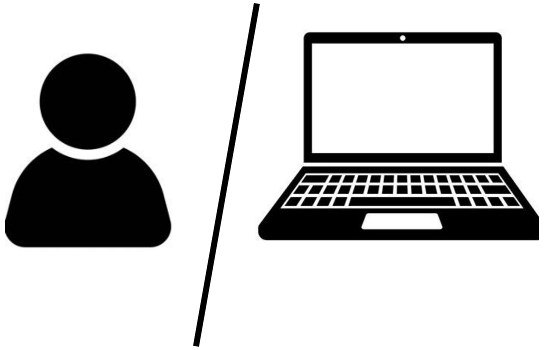
Region of attraction: $\Omega = \{\boldsymbol{x} \in \mathcal{D} : \dot{V}(\boldsymbol{x}) \leq \eta\}$

(positively invariant set)

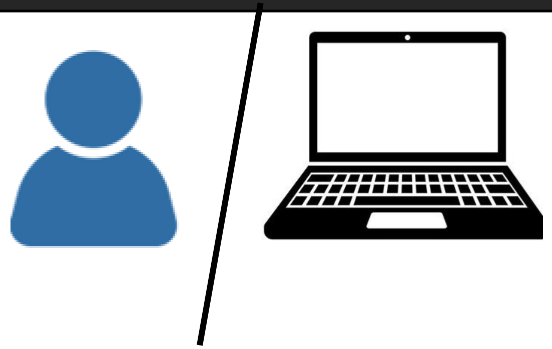


How to construct them?

Aleksandr Lyapunov



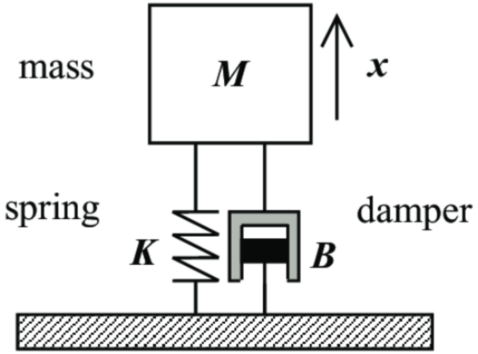
How to construct them?



(quadratic)

$$V(x) = \frac{1}{2}\|x\|^2 \text{ or, more generally, } V(x) = x^T P x$$

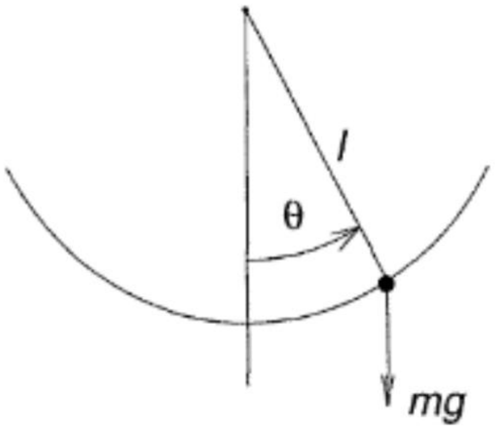
(polynomial)



$$m\ddot{x} + c|\dot{x}|\dot{x} + k_1 x + k_2 x^3 = 0$$

$$V(\mathbf{x}) = \frac{1}{2} m x_2^2 + \frac{1}{2} k_1 x_1^2 + \frac{1}{4} k_2 x_1^4$$

(trigonometric)



$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -a \sin x_1$$

$$V(x) = a(1 - \cos x_1) + \tfrac{1}{2} x_2^2$$

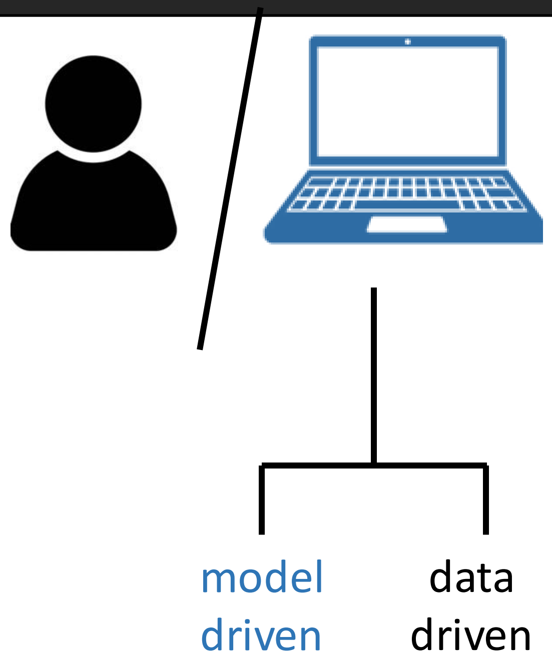
Khalil. *Nonlinear systems* (2002).

(switched systems)

$$V(x) := \max\left\{\min\{x^\top P_1 x, x^\top P_2 x\}, x^\top P_3 x\right\}$$

Rossa, Tanwani, Zaccarian. Max–min Lyapunov functions for switched systems and related differential inclusions. *Automatica* (2020).

How to construct them?

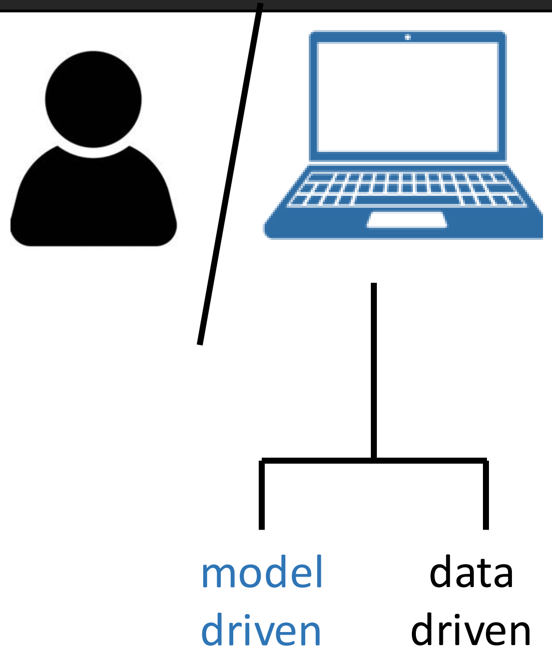


$$\Omega = \{x \in \mathcal{D} \ : \ V(\boldsymbol{x}) \leq \eta\} \qquad \max_{V(x)} \ \eta$$

SOS optimization

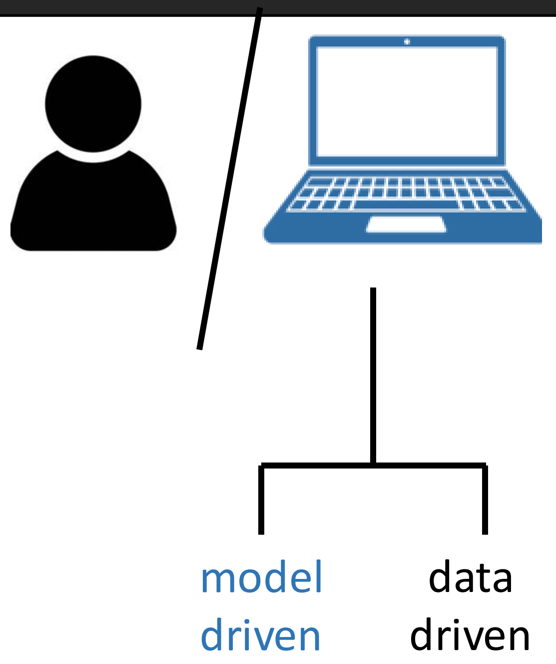
Finding a Lyapunov polynomial function is an NP-hard problem.

Finding a Lyapunov “sum-of-squares” polynomial function can be done efficiently with semidefinite programming (convex optimization).



Papachristodoulou & Prajna. On the construction of Lyapunov functions using the sum of squares decomposition. *IEEE Conference on Decision and Control* (2002).

SOS optimization



Finding a Lyapunov polynomial function is an NP-hard problem.

Finding a Lyapunov "sum-of-squares" polynomial function can be done efficiently with semidefinite programming (convex optimization).

Definition. $p(x) := p(x_1, \ldots, x_n) \in$ SOS if $p(x) = \sum_i h_i^2(x)$.

Example. $p(x_1, x_2) = x^2 - x_1 x_2^2 + x_2^4 + 1 = 0.75(x_1 - x_2^2)^2 + 0.25(x_1 + x_2)^2 + 1^2$.

Semidefinite programming. $p(x) = z(x)^T Q z(x) \in$ SOS iff $\exists Q \geq 0 \quad \equiv \quad Q_0 + \sum_i \lambda_i Q_i \geq 0$

monomials $[x_1, x_2, x_1^2, x_1 x_2, x_2^2, \ldots]$

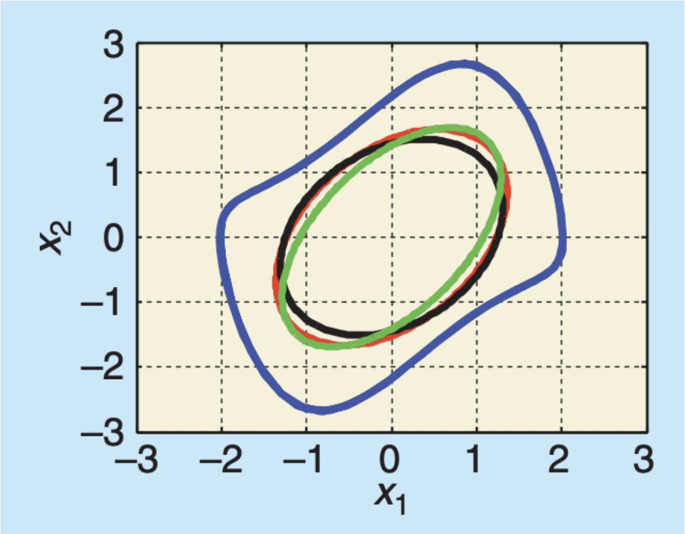
$$\min_{Q \in \mathcal{S}^n} \quad 1$$

s.t. affine function$(Q) \geq 0$

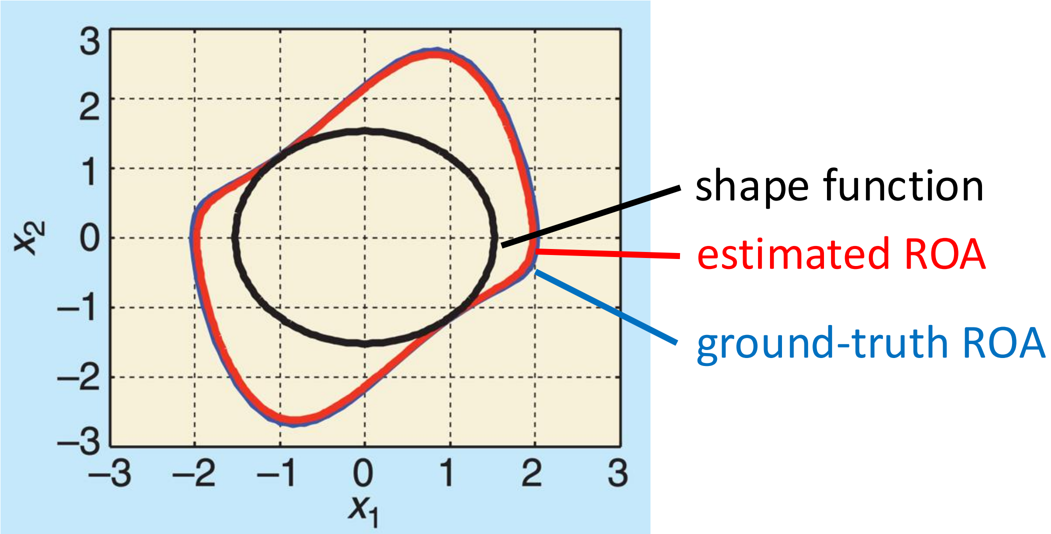
Papachristodoulou & Prajna. On the construction of Lyapunov functions using the sum of squares decomposition. *IEEE Conference on Decision and Control* (2002).

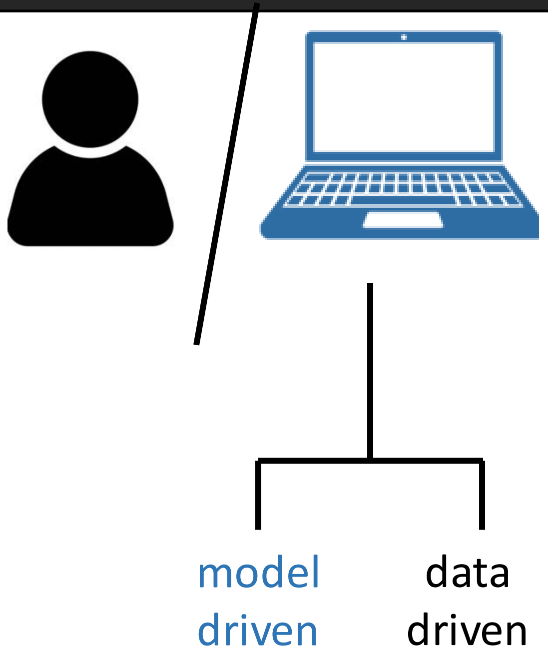
SOS optimization

quadratic Lyapunov functions



degree 6 SOS Lyapunov functions





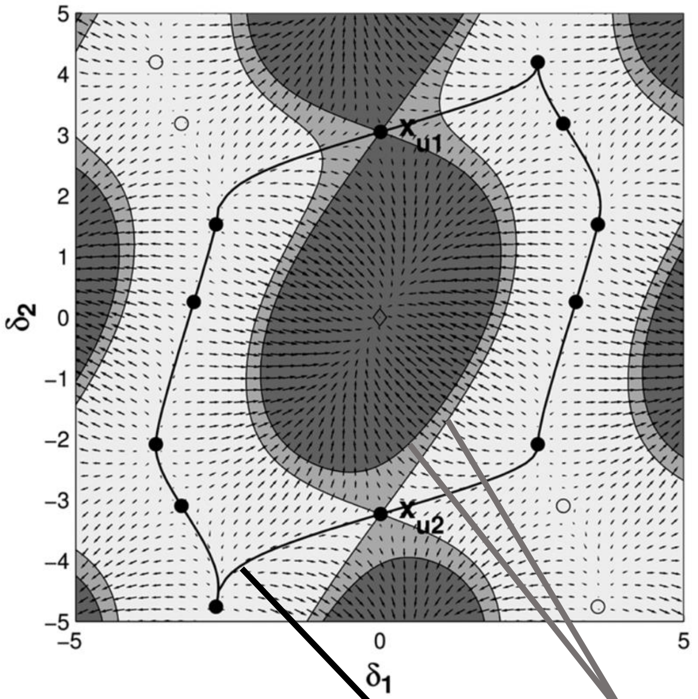
Packard, Topcu, Seiler Jr, Balas. Help on SOS. *IEEE Control Systems Magazine* (2010).

$$\max_{V(x)} \eta$$

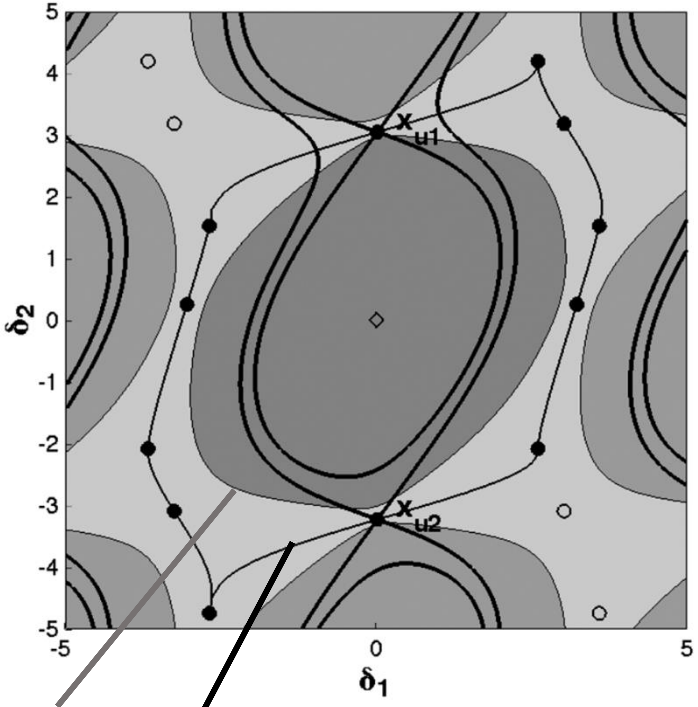
$$\Omega = \{x \in \mathcal{D} \ : \ V(\boldsymbol{x}) \leq \eta\}$$

SOS optimization

analytical Lyapunov functions

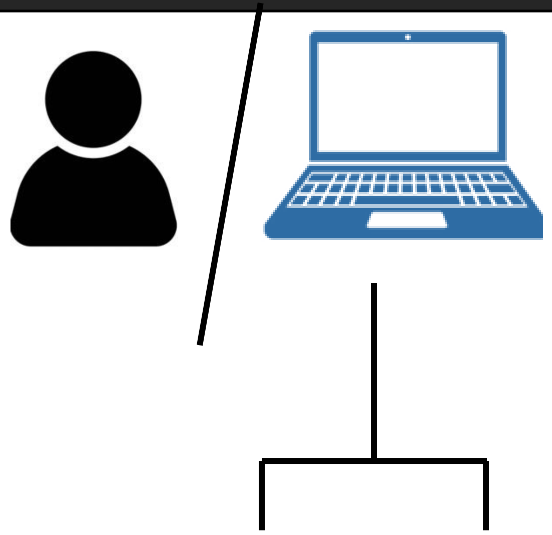


degree 2 SOS Lyapunov functions



estimated ROA

ground-truth ROA



model driven

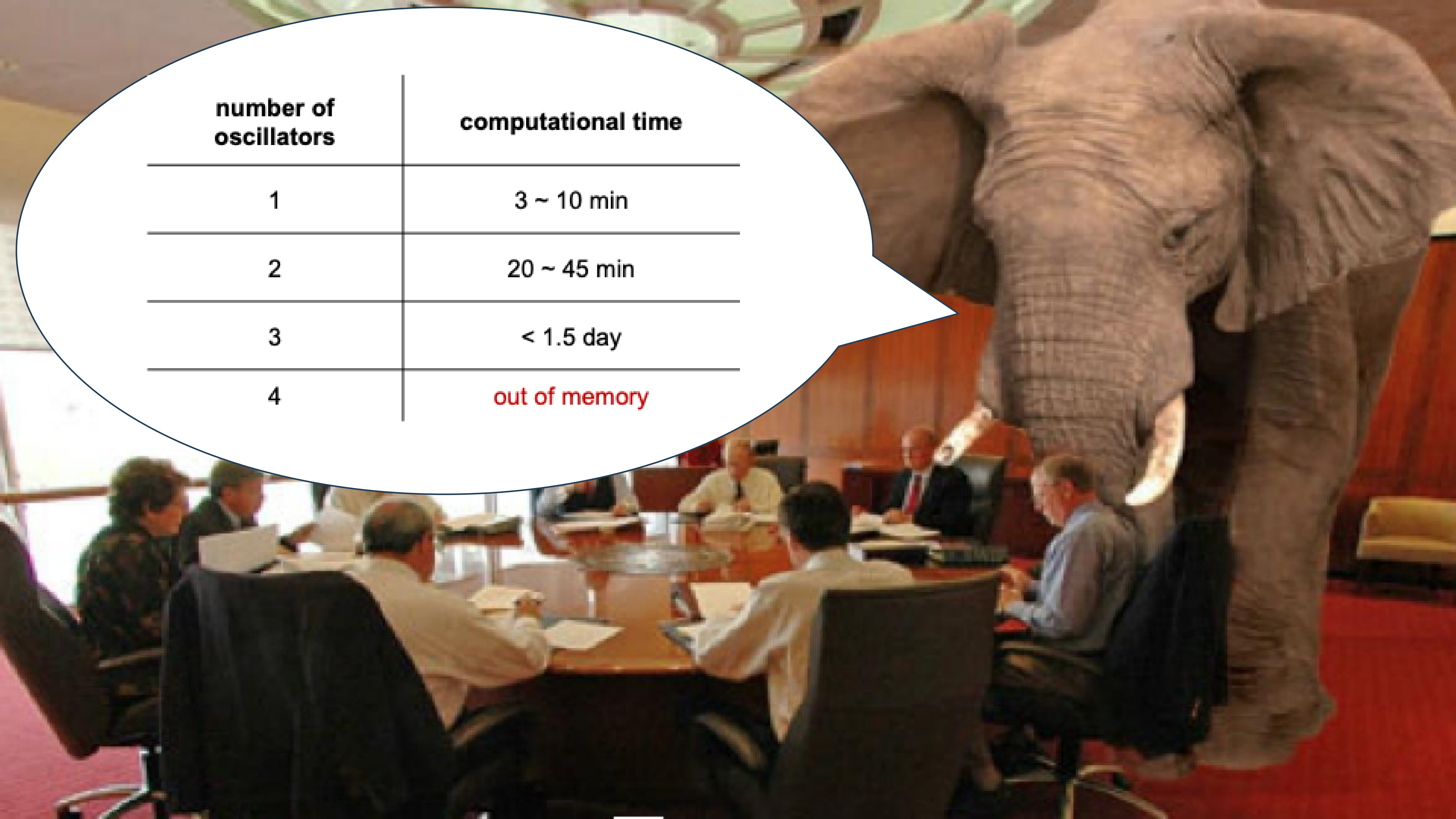
data driven

$$\max_{V(x)} \ \eta$$

$$
\begin{aligned}
V(x) = {} & 0.0030\sin(x_1) - 0.00008x_4 - 0.2683\cos(x_1) \\
& - 0.2649\cos(x_3) - 0.0030x_2 + 0.0044\sin(x_3) \\
& - 0.2377\cos(x_1)\cos(x_3) + 0.0008\cos(x_1)\sin(x_1) \\
& + 0.0047\cos(x_1)\sin(x_3) - 0.0037\cos(x_3)\sin(x_1) \\
& - 0.0092\cos(x_3)\sin(x_3) - 0.1588\sin(x_1)\sin(x_3) \\
& - 0.0109\cos(x_1)^2 + 0.0203\cos(x_3)^2 - 0.0004x_2x_4 \\
& - 0.0016x_2\cos(x_1) + 0.0047x_2\cos(x_3) \\
& + 0.0011x_4\cos(x_1) - 0.0010x_4\cos(x_3) \\
& + 0.0579x_2\sin(x_1) + 0.0219x_2\sin(x_3) \\
& + 0.0195x_4\sin(x_1) + 0.0972x_4\sin(x_3) \\
& + 0.1461x_2^2 + 0.1703x_4^2 + 0.7614.
\end{aligned}
$$

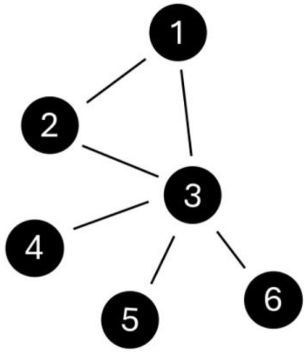
Anghel, Milano, Papachristodoulou. Algorithmic construction of Lyapunov functions for power system stability analysis. *IEEE Transactions on Circuits and Systems I* (2013).

What problem??



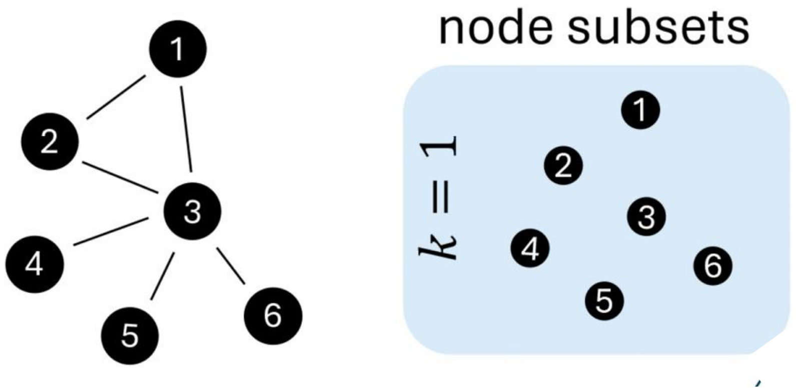
number of oscillators	computational time
1	3 ~ 10 min
2	20 ~ 45 min
3	< 1.5 day
4	out of memory

Distributed approach



$$\dot{\boldsymbol{x}}_i = \boldsymbol{f}_i(\boldsymbol{x}_i) + \sum_{j=1}^{N} K_{ij}\boldsymbol{g}(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad i = 1, \ldots, N,$$

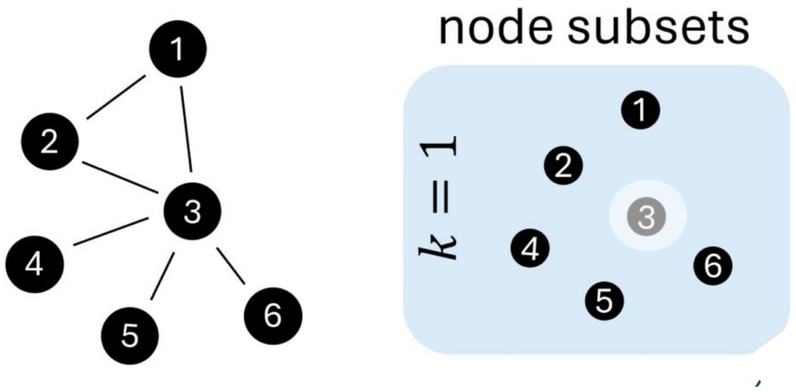
Distributed approach



For each subset $\mathcal{V}_p$, we are going to build a partial function $V_p : \mathcal{S}_p \mapsto \mathbb{R}$ defined on the constrained subspace

$$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : \boldsymbol{x}_j = 0, \;\; \forall j \notin \mathcal{V}_p\}$$

Distributed approach

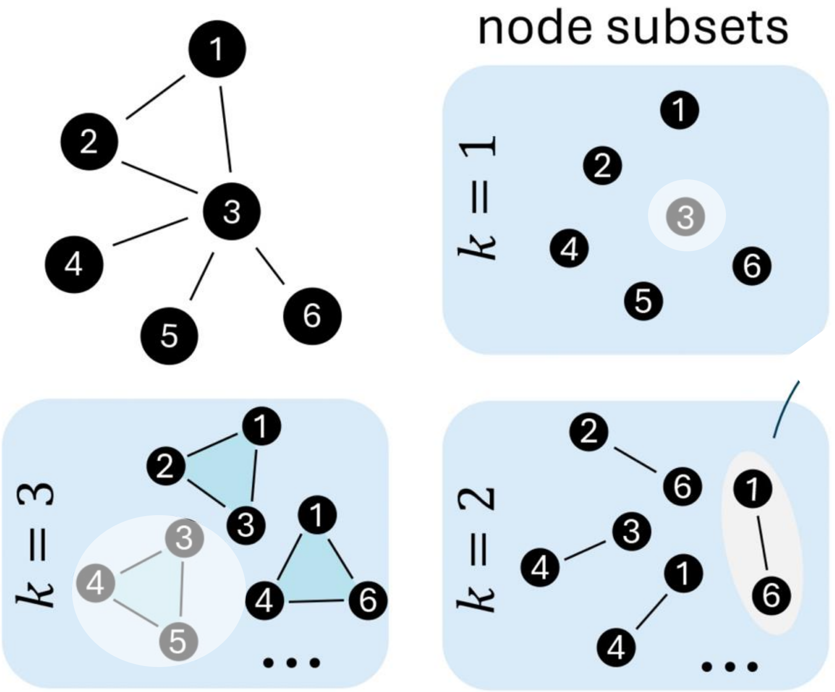


For each subset $\mathcal{V}_p$, we are going to build a partial function $V_p : \mathcal{S}_p \mapsto \mathbb{R}$ defined on the constrained subspace

$$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : \boldsymbol{x}_j = 0, \;\; \forall j \notin \mathcal{V}_p\}$$

$$\mathcal{V}_p = \{3\},$$
$$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : x_1 = x_2 = x_4 = x_5 = x_6 = 0\}$$

Distributed approach



For each subset $\mathcal{V}_p$, we are going to build a partial function $V_p : \mathcal{S}_p \mapsto \mathbb{R}$ defined on the constrained subspace

$$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : \boldsymbol{x}_j = 0, \ \forall j \notin \mathcal{V}_p\}$$

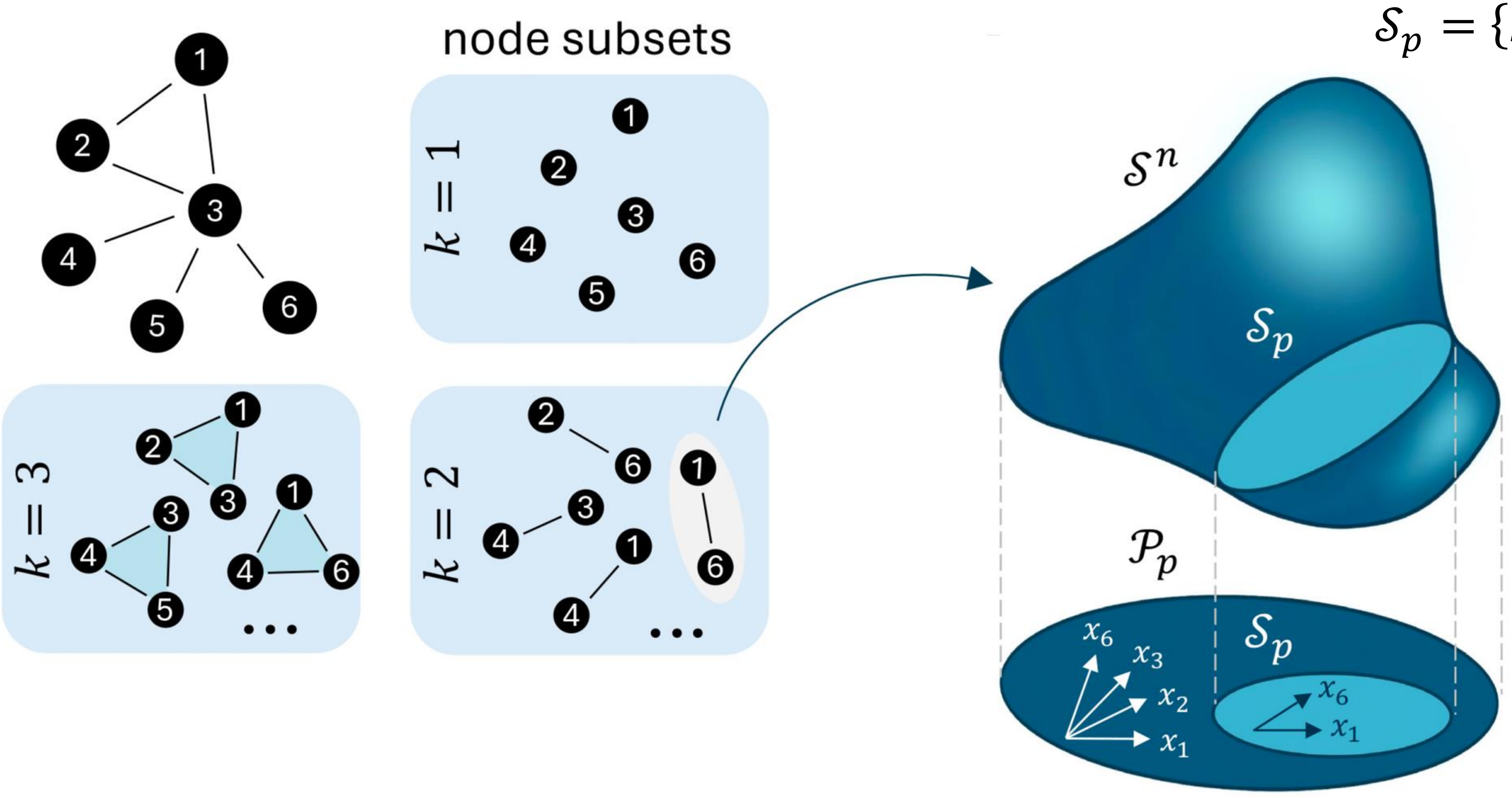
$$\mathcal{V}_p = \{3\},$$
$$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : x_1 = x_2 = x_4 = x_5 = x_6 = 0\}$$

$$\mathcal{V}_p = \{1,6\},$$
$$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : x_2 = x_3 = x_4 = x_5 = 0\}$$

$$\mathcal{V}_p = \{3,4,5\},$$
$$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : x_1 = x_2 = x_6 = 0\}$$

Distributed approach

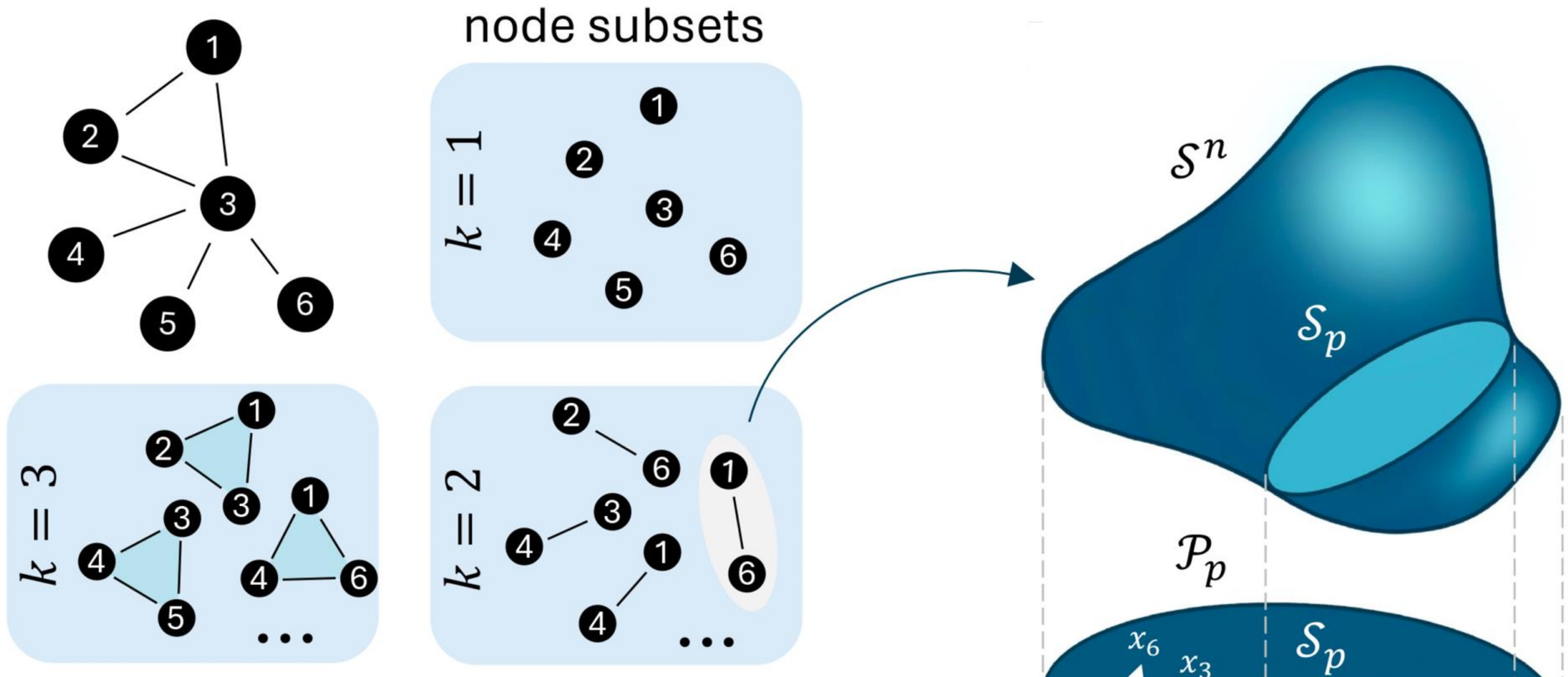
$\mathcal{V}_p = \{1,6\},$
$\mathcal{S}_p = \{\boldsymbol{x} \in \mathcal{S}^n : x_2 = x_3 = x_4 = x_5 = 0\}$



partial function $V_p : \mathcal{S}_p \mapsto \mathbb{R}$

$$V_p\left(\left(P_p \otimes I_m\right)\boldsymbol{x}\right)$$

Distributed Lyapunov function



$$\tilde{V}_c = w_i \, \textcircled{i} + \ldots + w_{ij} \, (\textcircled{i}-\textcircled{j}) + \ldots + w_{ijl} \, (\textcircled{i}\textcircled{j}\textcircled{l}) + \ldots$$

$$= \sum_{p=1}^{L} w_p V_p$$

Main result

If $\tilde{V}_c(x) > 0, \forall x \in \mathcal{D} \backslash \{0\},$ **then the composite function** $\widetilde{V}_c = \sum_{p=1}^{L} w_p V_p$ **is a Lyapunov function.**

$\tilde{V}_c(0) = 0$

$\dot{\tilde{V}}_c(x) \leq 0, \forall x \in \mathcal{D} \backslash \{0\}$

Main result

If $\tilde{V}_c(x) > 0, \forall x \in \mathcal{D}\backslash\{0\}$, **then the composite function** $\widetilde{V}_c = \sum_{p=1}^{L} w_p V_p$ **is a Lyapunov function.**
$\tilde{V}_c(0) = 0$
$\dot{\tilde{V}}_c(x) \leq 0, \forall x \in \mathcal{D}\backslash\{0\}$

Conditions I and II: If V_p is an SOS polynomial, then it is positive definite and so will be $\tilde{V}_c$ for nonnegative weights w_p.

Main result

If $\tilde{V}_c(x) > 0, \forall x \in \mathcal{D}\backslash\{0\}$, **then the composite function** $\tilde{V}_c = \sum_{p=1}^{L} w_p V_p$ **is a Lyapunov function.**
$\tilde{V}_c(0) = 0$
$\dot{\tilde{V}}_c(x) \le 0, \forall x \in \mathcal{D}\backslash\{0\}$

Conditions I and II: If V_p is an SOS polynomial, then it is positive definite and so will be $\tilde{V}_c$ for nonnegative weights w_p.

Main result: If all partial functions V_p satisfy the relationship

$$\begin{bmatrix} \dot{V}_1 \\ \vdots \\ \dot{V}_L \end{bmatrix} \le (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + B\tilde{V}_c \le 0$$

... then all partial functions V_p are also Lyapunov functions

(i) $A_{pr} - B_{pr} > 0$ for all $r = 1, \ldots, N$ and $r \neq p$,
(ii) $\sum_{r=1}^{L} (A_{pr} - B_{pr}) \le 0$,
(iii) $\sum_{r=1}^{L} B_{pr} \ge 0$,

Main result

If $\tilde{V}_c(x) > 0, \forall x \in \mathcal{D}\backslash\{0\}$, **then the composite function** $\tilde{V}_c = \sum_{p=1}^{L} w_p V_p$ **is a Lyapunov function.**

$\tilde{V}_c(0) = 0$

$\dot{\tilde{V}}_c(x) \leq 0, \forall x \in \mathcal{D}\backslash\{0\}$

Conditions I and II: If V_p is an SOS polynomial, then it is positive definite and so will be $\tilde{V}_c$ for nonnegative weights w_p.

Use SOS optimization to find each of the partial functions in parallel.

Main result: If all partial functions V_p satisfy the relationship

$$\begin{bmatrix} \dot{V}_1 \\ \vdots \\ \dot{V}_L \end{bmatrix} \leq (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + B\tilde{V}_c \leq 0$$

... then all partial functions V_p are also Lyapunov functions

(i) $A_{pr} - B_{pr} > 0$ for all $r = 1, \ldots, N$ and $r \neq p$,

(ii) $\sum_{r=1}^{L} (A_{pr} - B_{pr}) \leq 0$,

(iii) $\sum_{r=1}^{L} B_{pr} \geq 0$,

Main result

If $\tilde{V}_c(x) > 0, \forall x \in \mathcal{D}\backslash\{0\},$
$\tilde{V}_c(0) = 0$
$\dot{\tilde{V}}_c(x) \leq 0, \forall x \in \mathcal{D}\backslash\{0\}$

then the composite function $\tilde{V}_c = \sum_{p=1}^{L} w_p V_p$ **is a Lyapunov function.**

Conditions I and II: If V_p is an SOS polynomial, then it is positive definite and so will be $\tilde{V}_c$ for nonnegative weights w_p.

Use SOS optimization to find each of the partial functions in parallel.

Main result: If all partial functions V_p satisfy the relationship

$$\begin{bmatrix} \dot{V}_1 \\ \vdots \\ \dot{V}_L \end{bmatrix} \leq (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + B\tilde{V}_c \leq 0$$

... then all partial functions V_p are also Lyapunov functions

Condition III: If $\boldsymbol{w}$ is a probability vector, then

$$\dot{\tilde{V}}_c \leq \boldsymbol{w}^T (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + V_c\, \boldsymbol{w}^T B \mathbf{1}_L < 0$$

... and the composite function $\tilde{V}_c$ is a Lyapunov function.

Main result

If $\tilde{V}_c(x) > 0, \forall x \in \mathcal{D}\backslash\{0\}$,
$\tilde{V}_c(0) = 0$
$\dot{\tilde{V}}_c(x) \leq 0, \forall x \in \mathcal{D}\backslash\{0\}$

then the composite function $\tilde{V}_c = \sum_{p=1}^{L} w_p V_p$ **is a Lyapunov function.**

Conditions I and II: If V_p is an SOS polynomial, then it is positive definite and so will be $\tilde{V}_c$ for nonnegative weights w_p.

Use SOS optimization to find each of the partial functions in parallel.

Main result: If all partial functions V_p satisfy the relationship

$$\begin{bmatrix} \dot{V}_1 \\ \vdots \\ \dot{V}_L \end{bmatrix} \leq (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + B\tilde{V}_c \leq 0$$

... then all partial functions V_p are also Lyapunov functions

Condition III: If $\boldsymbol{w}$ is a probability vector, then

$$\dot{\tilde{V}}_c \leq \boldsymbol{w}^T (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + V_c\, \boldsymbol{w}^T B \mathbf{1}_L < 0$$

... and the composite function $\tilde{V}_c$ is a Lyapunov function.

Solve LMI problem to find $(A, B, \boldsymbol{w})$.

Main result

If $\tilde{V}_c(x) > 0, \forall x \in \mathcal{D}\backslash\{0\}$, **then the composite function** $\tilde{V}_c = \sum_{p=1}^{L} w_p V_p$ **is a Lyapunov function.**

$\tilde{V}_c(0) = 0$

$\dot{\tilde{V}}_c(x) \le 0, \forall x \in \mathcal{D}\backslash\{0\}$

Conditions I and II: If V_p is an SOS polynomial, then it is positive definite and so will be $\tilde{V}_c$ for nonnegative weights w_p.

Use SOS optimization to find each of the partial functions in parallel.

Main result: If all partial functions V_p satisfy the relationship

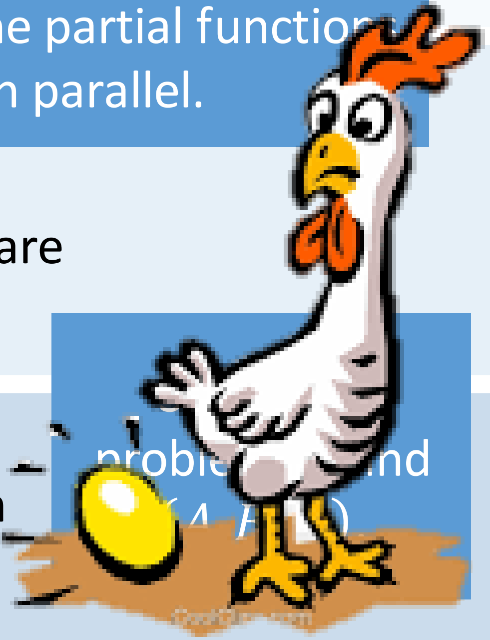
$$\begin{bmatrix} \dot{V}_1 \\ \vdots \\ \dot{V}_L \end{bmatrix} \le (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + B\tilde{V}_c \le 0$$

... then all partial functions V_p are also Lyapunov functions

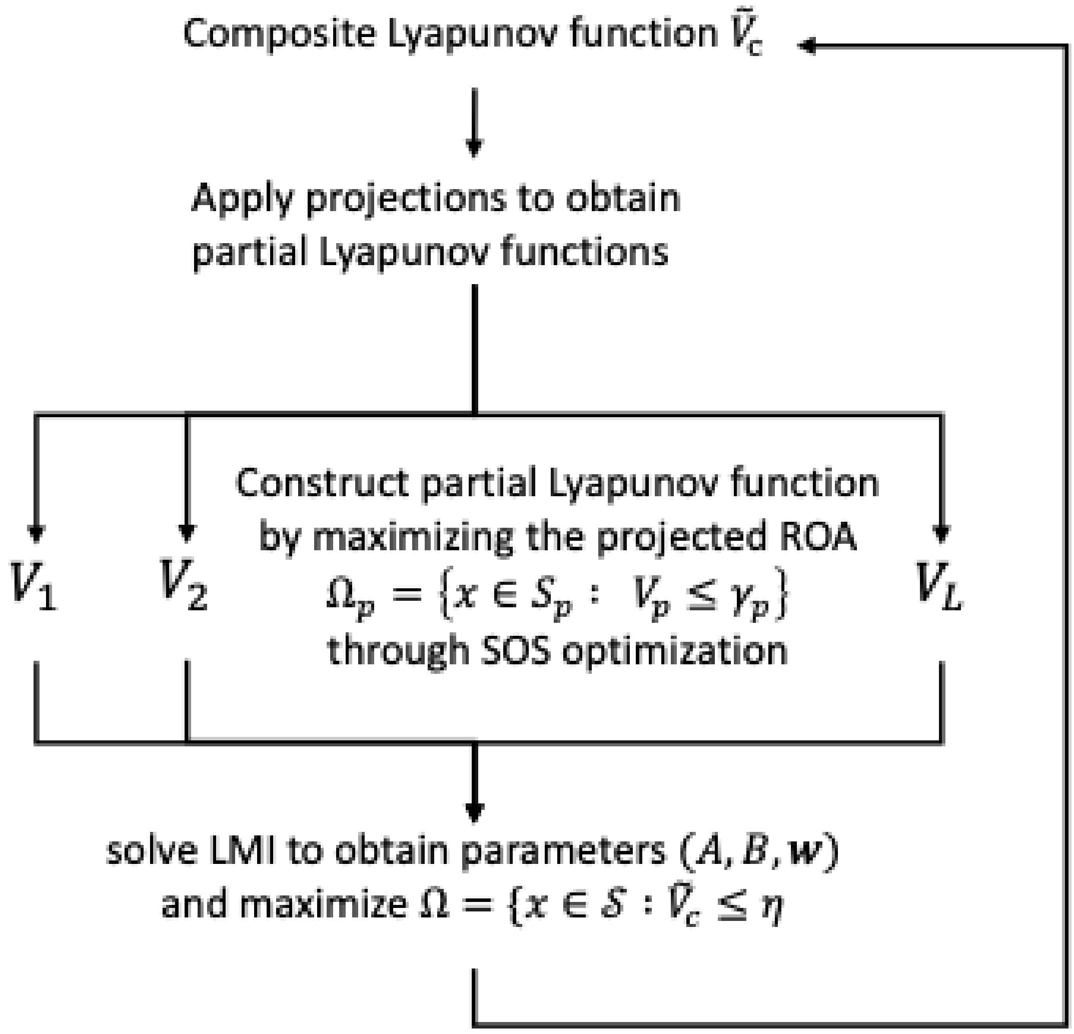
Condition III: If $\boldsymbol{w}$ is a probability vector, then

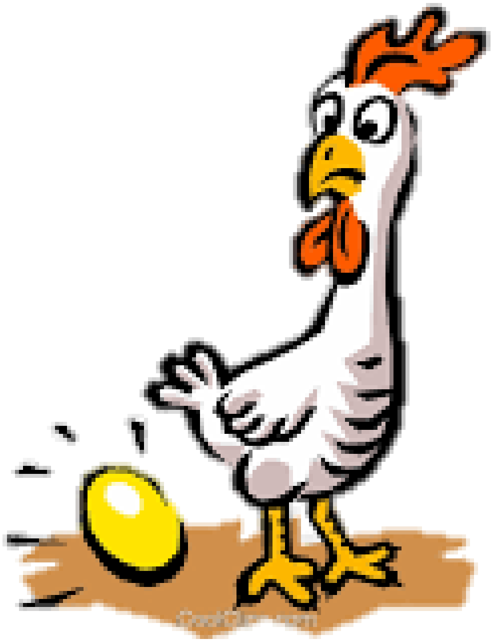
$$\dot{\tilde{V}}_c \le \boldsymbol{w}^T (A - B) \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} + V_c\, \boldsymbol{w}^T B \mathbf{1}_L < 0$$

... and the composite function $\tilde{V}_c$ is a Lyapunov function.



Optimization procedure



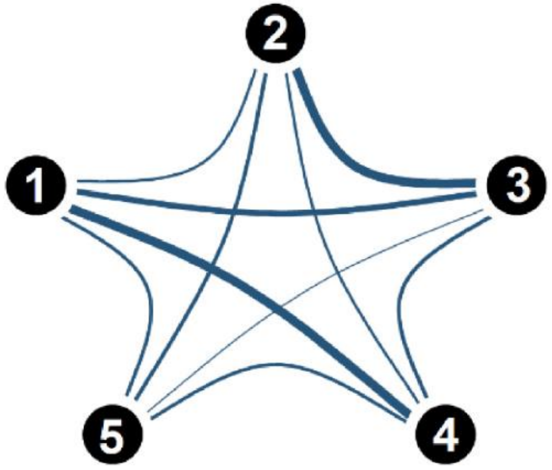


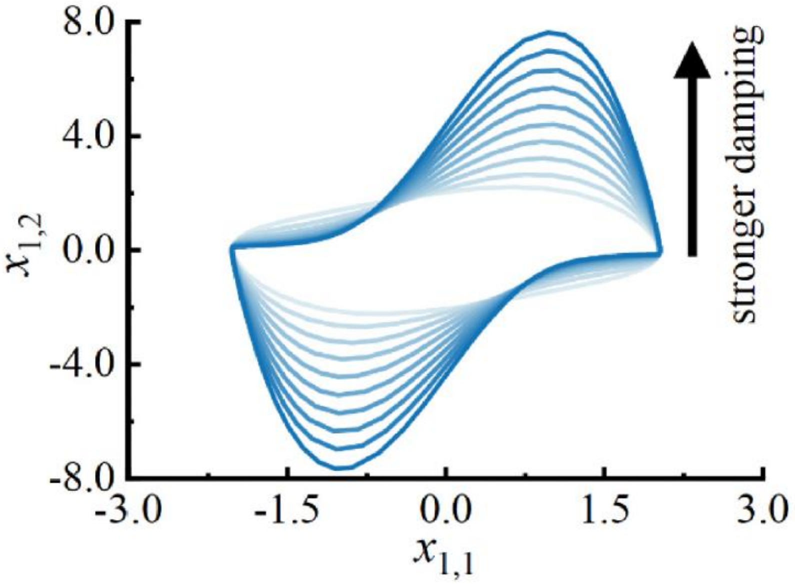
Proof of concept

van der Pol oscillators

$$\dot{x}_{i,1} = -x_{i,2},$$

$$\dot{x}_{i,2} = \alpha_i\left(x_{i,1}^2 - 1\right)\, x_{i,2} + x_{i,1} + \frac{1}{N}\sum_{j=1}^{N} K_{ij}\left(x_{j,1} - x_{i,1}\right)$$



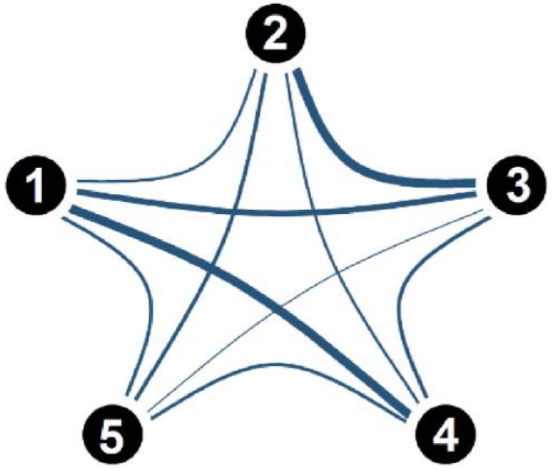


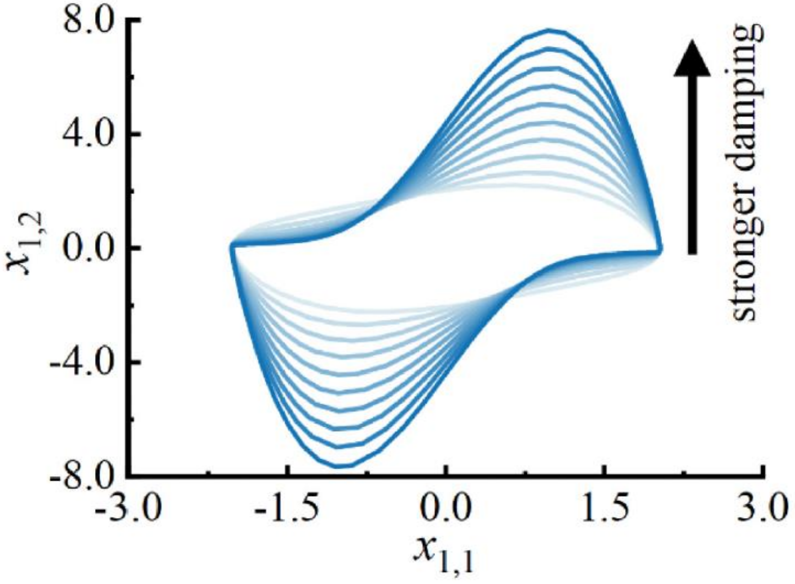
Proof of concept

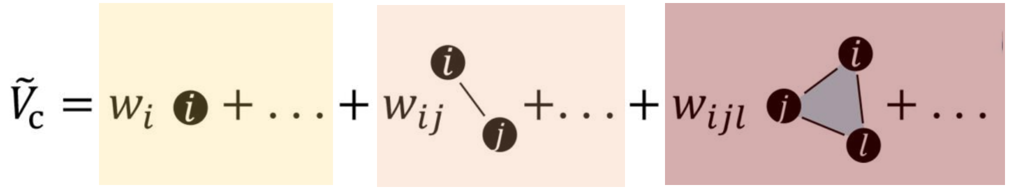
van der Pol oscillators

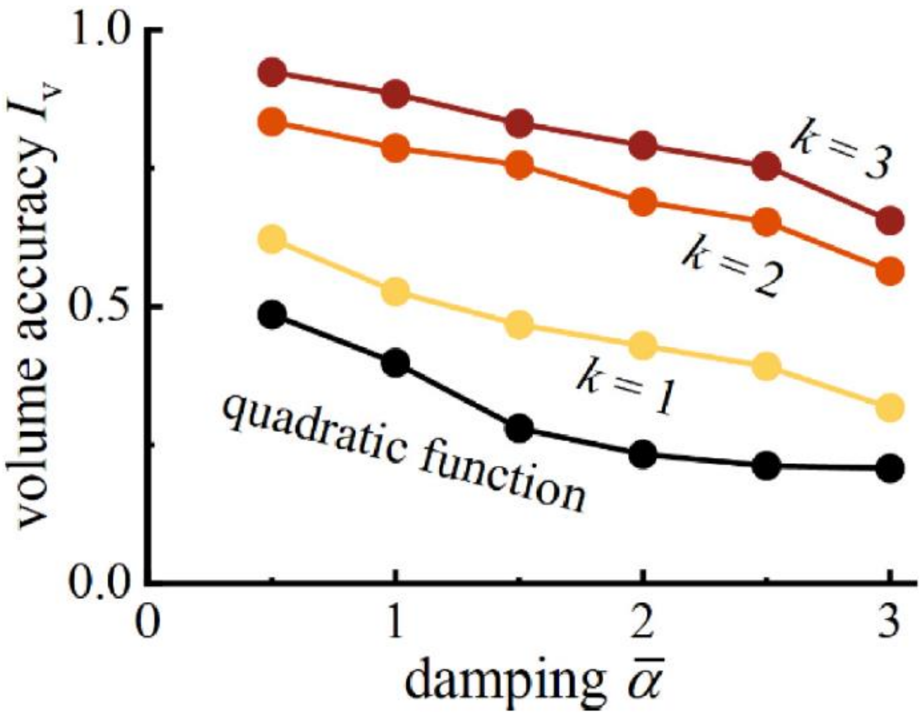
$$\dot{x}_{i,1} = -x_{i,2},$$

$$\dot{x}_{i,2} = \alpha_i \left(x_{i,1}^2 - 1\right) x_{i,2} + x_{i,1} + \frac{1}{N} \sum_{j=1}^{N} K_{ij} \left(x_{j,1} - x_{i,1}\right)$$

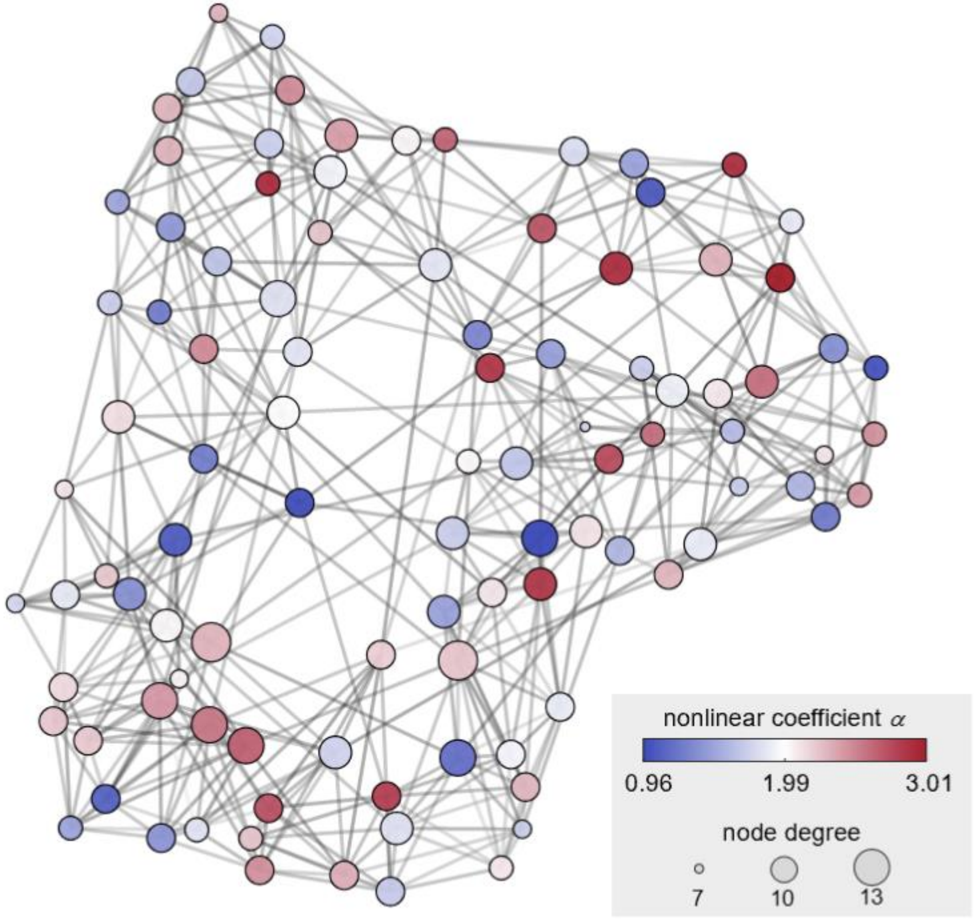


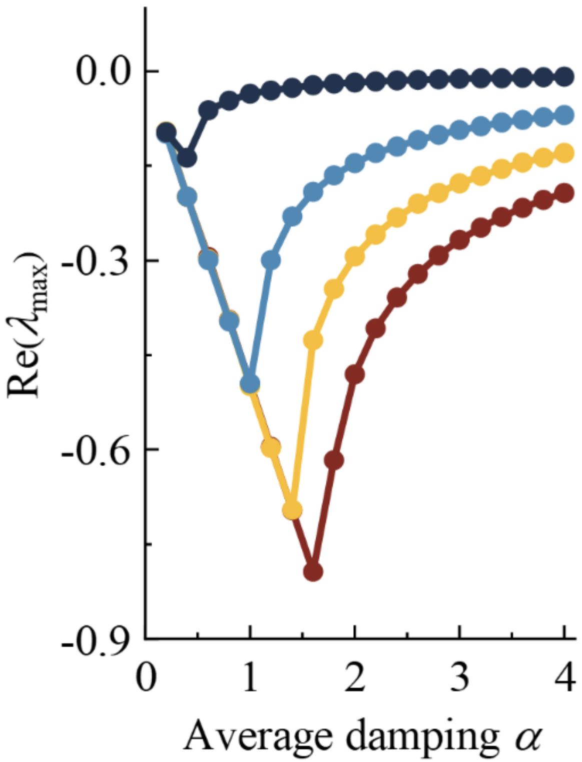


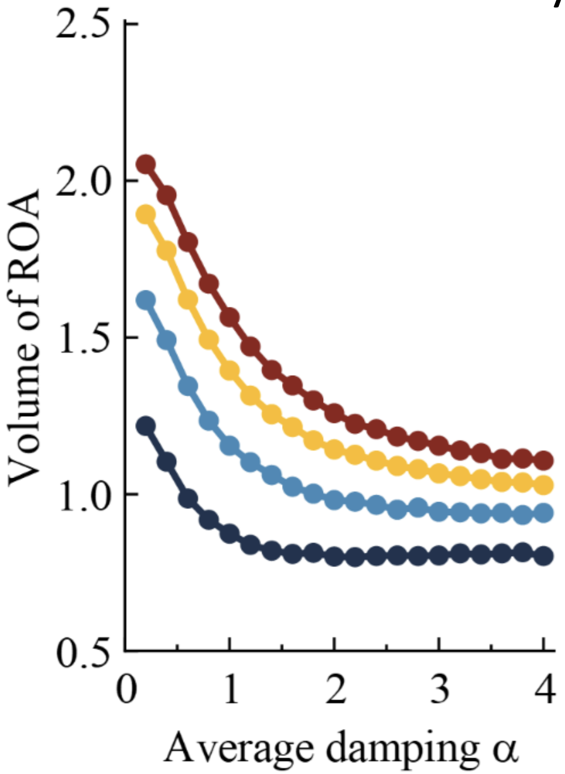




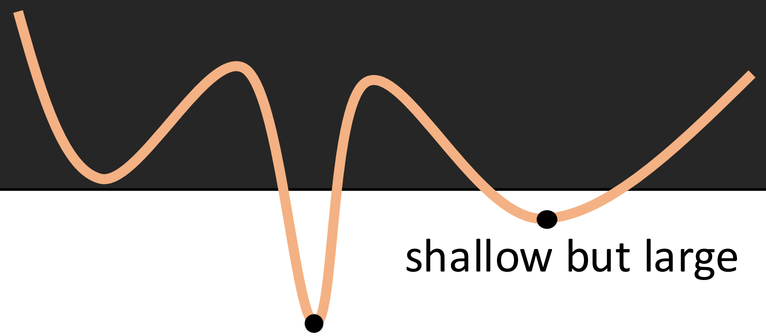
Large-scale networks

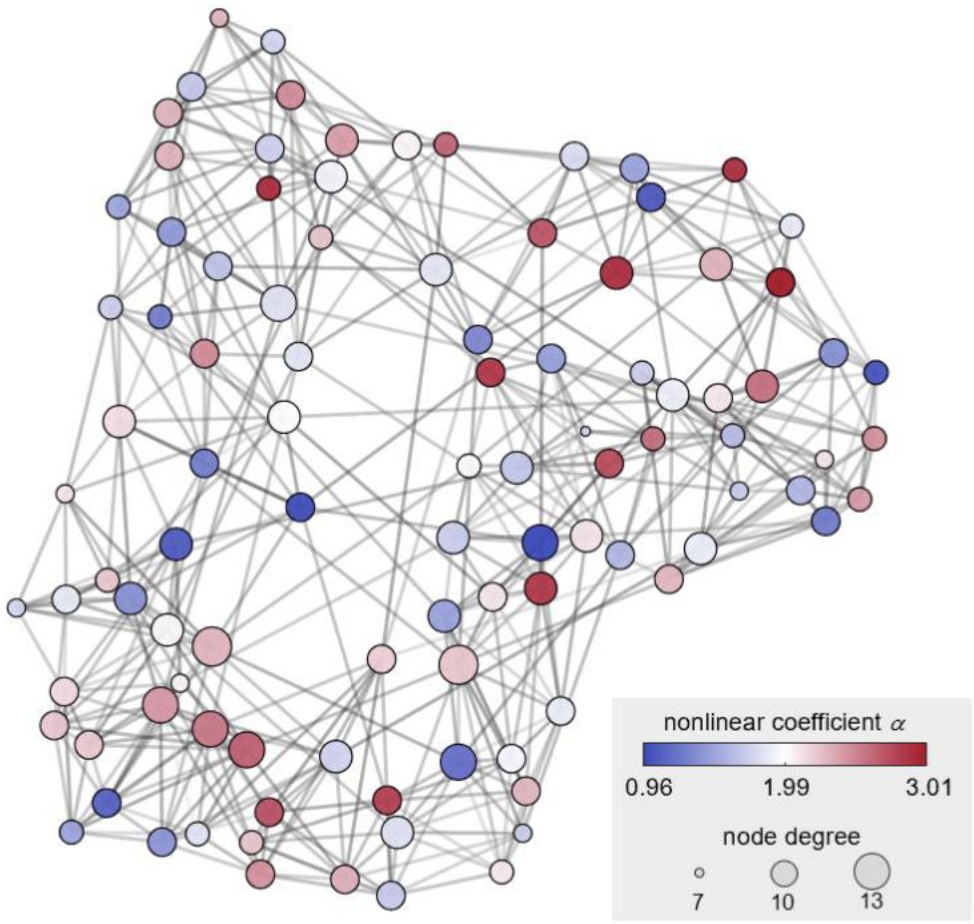


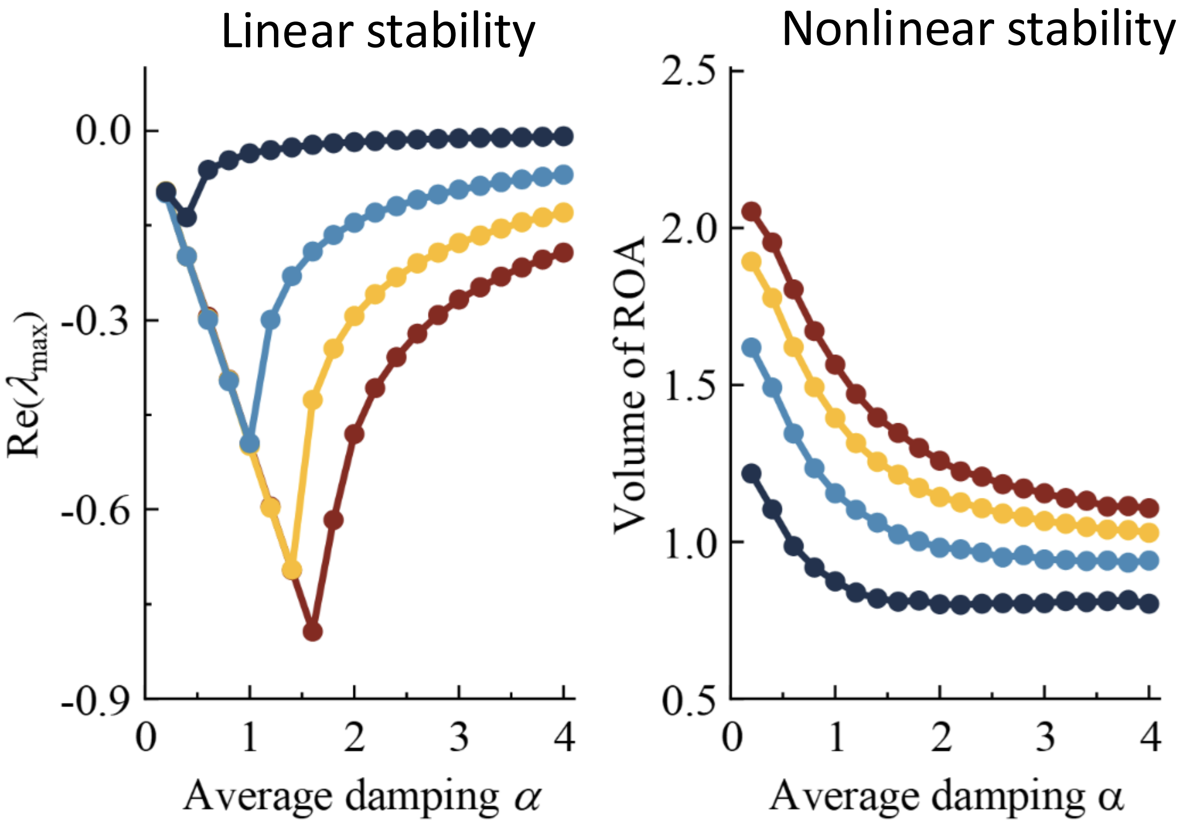




Stability trade-offs

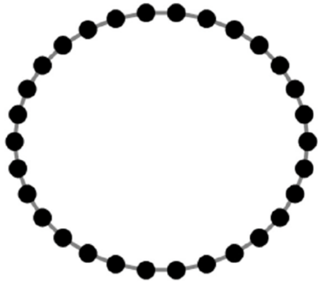






Oscillator networks

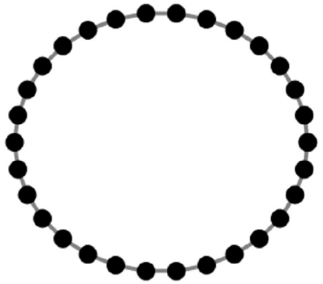
oscillator Ising machines



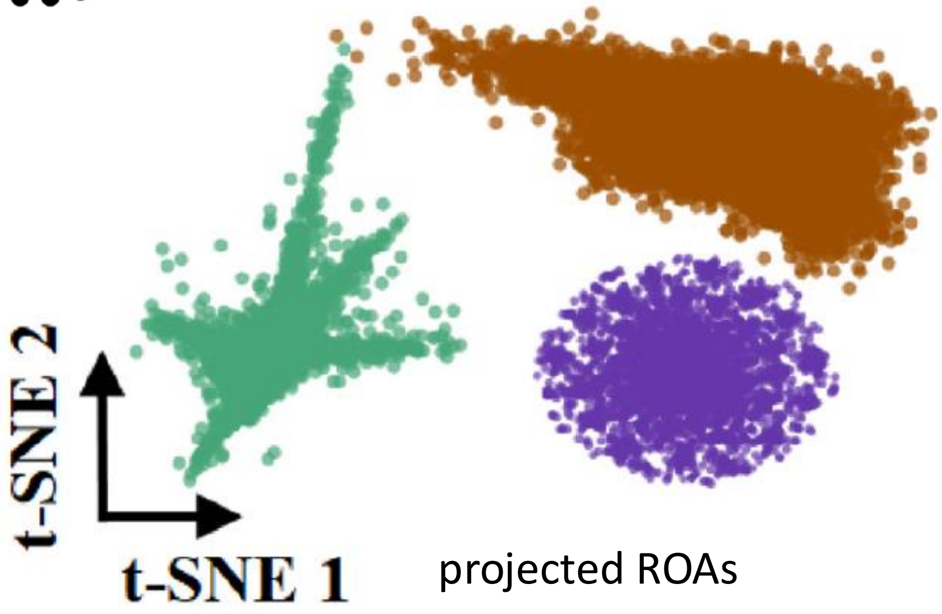
$$\dot{\phi}_i = \sum_{j=1}^{N} K_{ij} \sin\left(x_j - x_i\right) - \mu \sin(2\phi_i)$$

Estimating complex ROA shapes

oscillator Ising machines



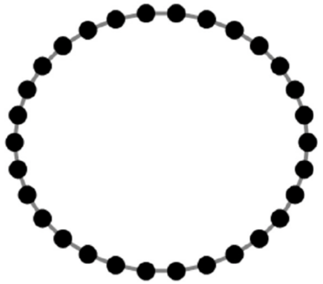
$$\dot{\phi}_i = \sum_{j=1}^{N} K_{ij} \sin\left(x_j - x_i\right) - \mu \sin(2\phi_i)$$



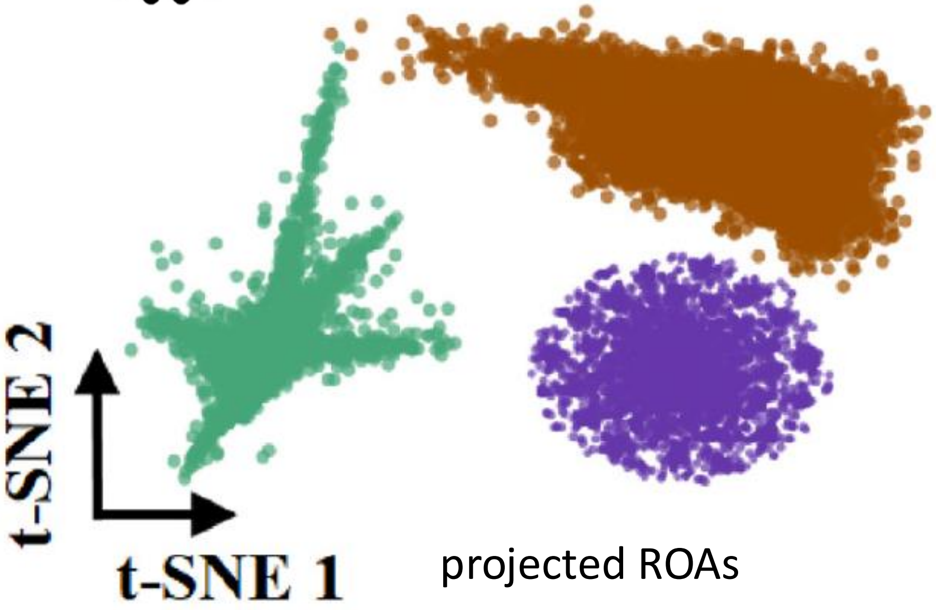
projected ROAs

Estimating complex ROA shapes

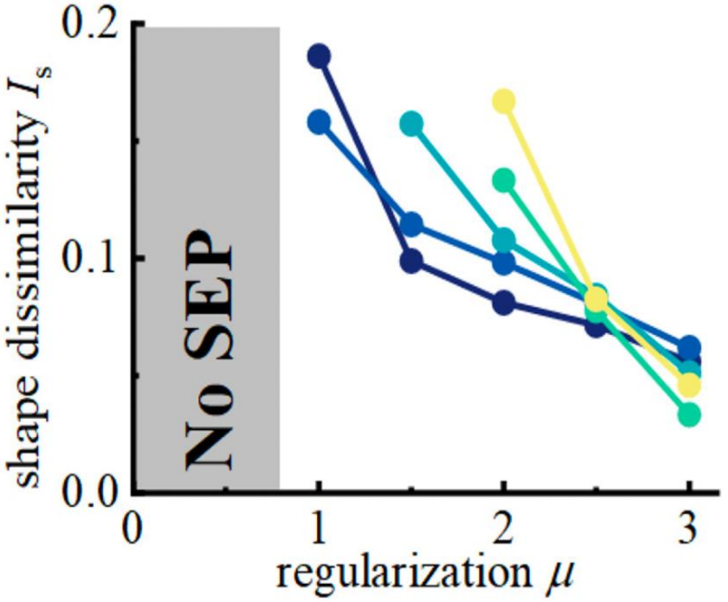
oscillator Ising machines

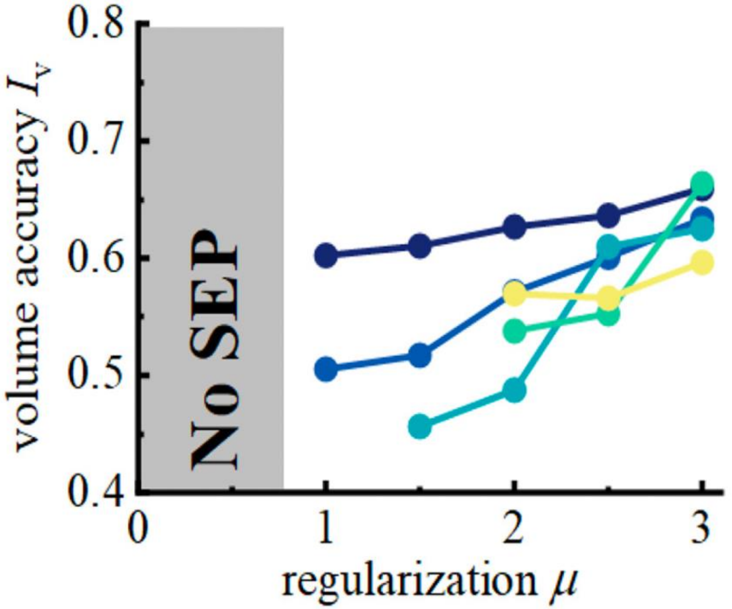


$$\dot{\phi}_i = \sum_{j=1}^{N} K_{ij} \sin(x_j - x_i) - \mu \sin(2\phi_i)$$



projected ROAs





Acknowledgements

IEEE CONTROL SYSTEMS LETTERS, VOL. 9, 2025

Distributed Lyapunov Functions for Nonlinear Networks

Yiming Wang, Arthur N. Montanari, and Adilson E. Motter, *Senior Member, IEEE*

Slides and references are available at my website:

www.montanariarthur.com