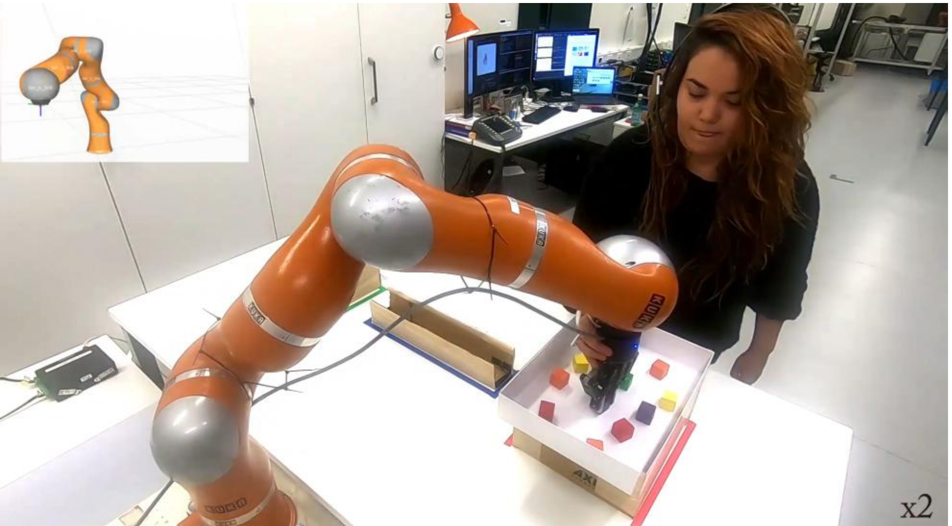
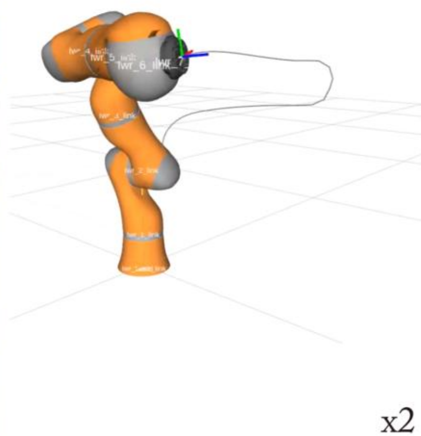
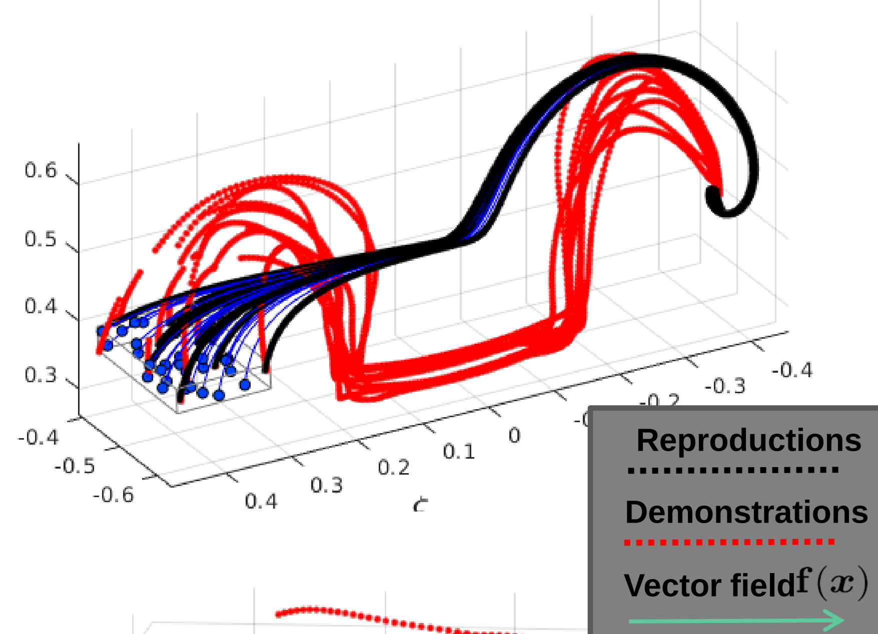
Learning Control Laws

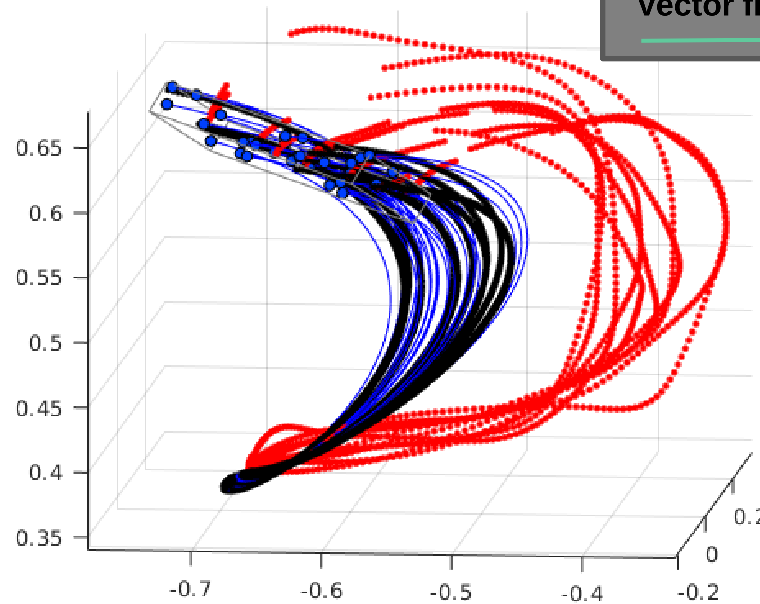
Linear Parameter Varying Dynamical Systems (LPVDS)

SEDS on Highly Non-Linear Trajectories



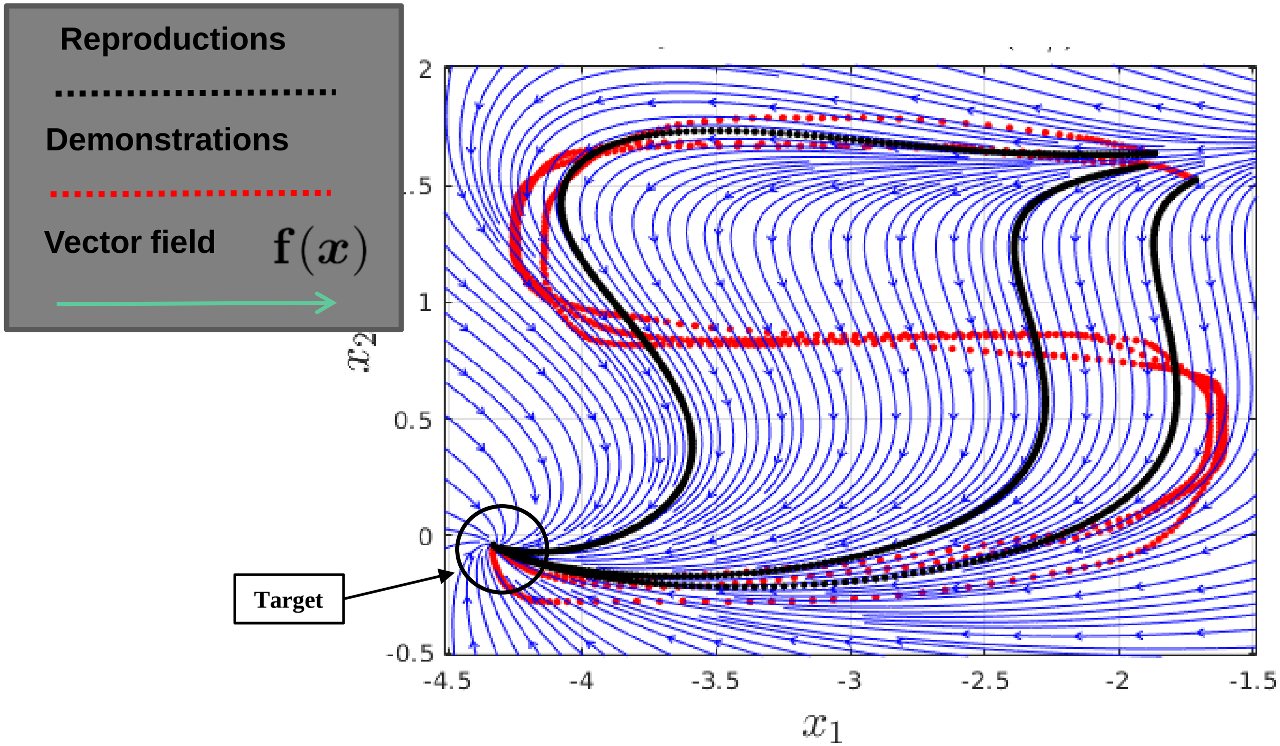






SEDS on Highly Non-Linear Trajectories

$$\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}) = \sum_{k=1}^{K} \gamma_k(\boldsymbol{x})(\mathbf{A}_k \boldsymbol{x} + \boldsymbol{b}_k)$$



- ✓ **Convergence ensured**
- ➢ **Inaccurate Reproduction of highly non-linear motions**

Why?

SEDS on Highly Non-Linear Trajectories

$$\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}) = \sum_{k=1}^{K} \gamma_k(\boldsymbol{x})(\mathbf{A}_k \boldsymbol{x} + \boldsymbol{b}_k)$$

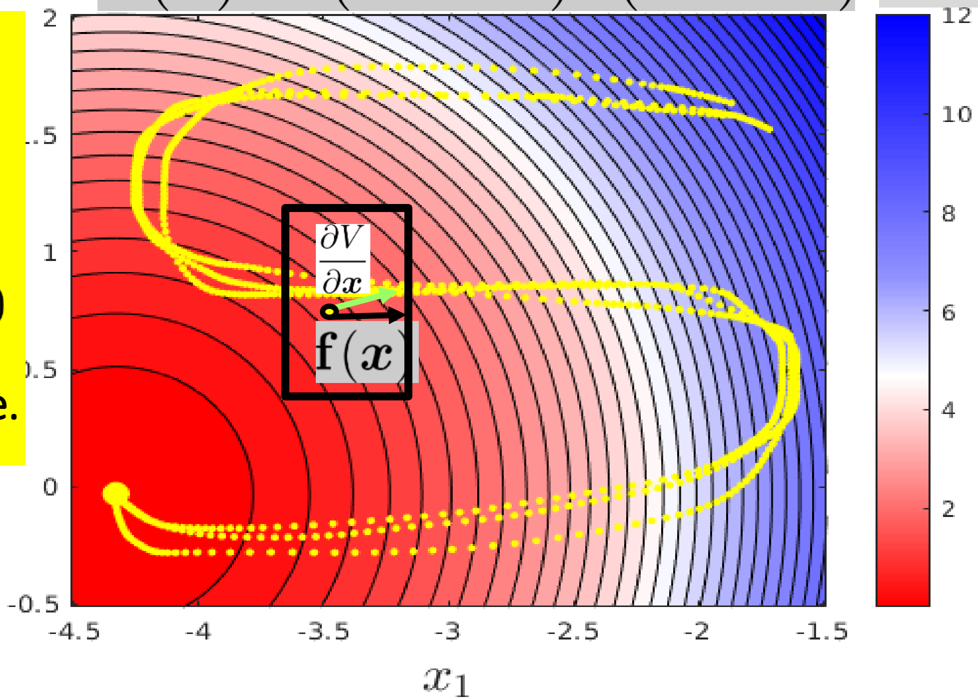
$$V(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}^*)^T (\boldsymbol{x} - \boldsymbol{x}^*)$$

SEDS Lyapunov Function

Highly Non-linear trajectories violate stability condition

$$\dot{V}(\boldsymbol{x}) = \frac{\partial V}{\partial \boldsymbol{x}} \mathbf{f}(\boldsymbol{x}) < 0$$

If V is too conservative.



SEDS on Highly Non-Linear Trajectories

$$\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}) = \sum_{k=1}^{K} \gamma_k(\boldsymbol{x})(\mathbf{A}_k \boldsymbol{x} + \boldsymbol{b}_k)$$

State dependent parameter vector

Linear Time-Invariant (LTI) DS

Stability of LTI can be shown if ∃ a generic Lyapunov Function:

$$V(x) = (x - x^*)^T P (x - x^*), \quad P = P^T, P \succ 0$$

Theorem:

The nonlinear DS above is Globally Asymptotically Stable at x^* if $\exists \; P = P^T, P \succ 0$, with $V(x) = (x - x^*)^T P (x - x^*)$, such that:

$$\begin{cases} (A^k)^T P + P A^k = Q^k, \quad Q^k = (Q^k)^T \\ b^k = -A^k x^* \end{cases} \quad \forall k = 1, \ldots, K$$

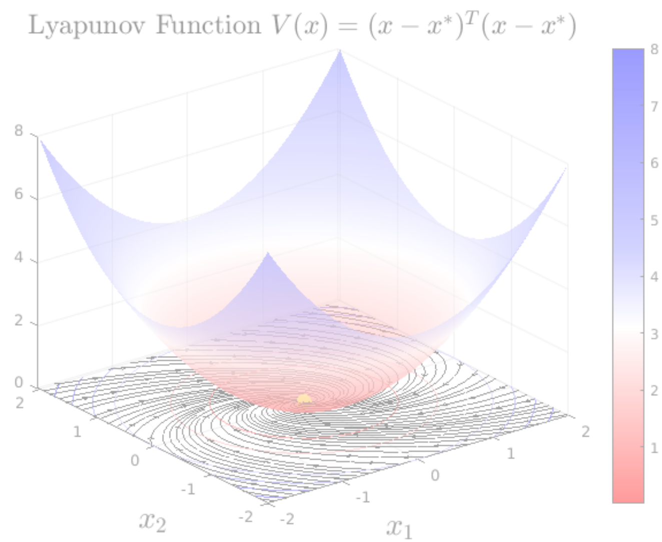
See Theorem 3.3 (Book)

Learning Non-linear DS with GMM's and P-QLF

Goal: Learn the parameters of a non-linear DS with P-QLF

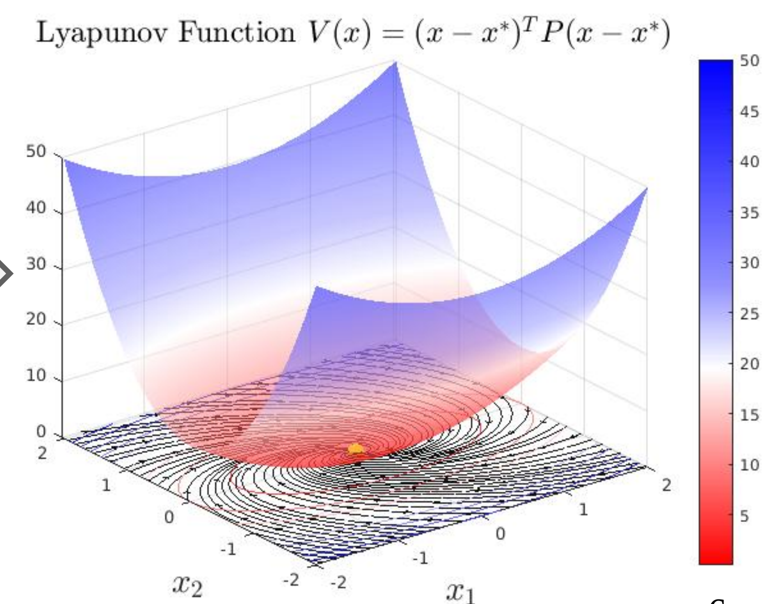
Quadratic Lyapunov Function (QLF)

$$V(x) = (x - x^*)^T (x - x^*)$$



Parameterized Quadratic Lyapunov Function (P-QLF)

$$V(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}^*)^T \mathbf{P} (\boldsymbol{x} - \boldsymbol{x}^*)$$

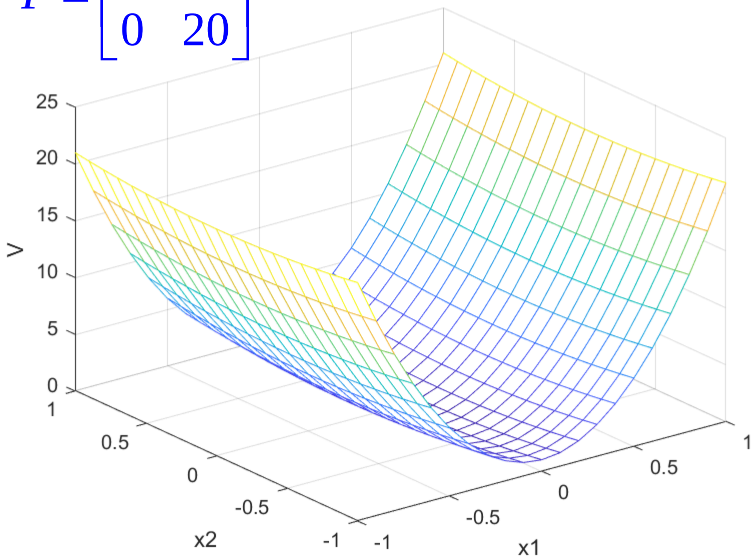


6

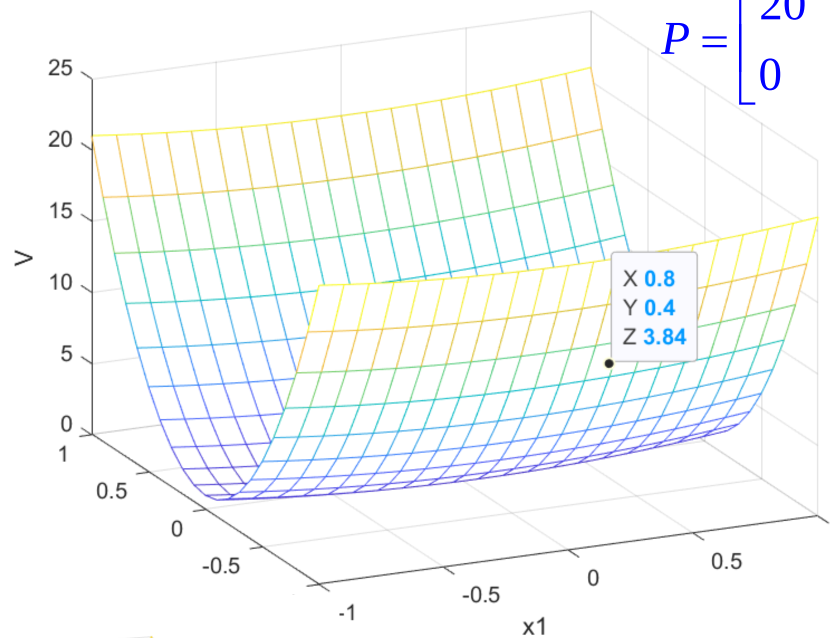
P's effect is of a reshaping of the Lyapunov function

P's effect is of a reshaping of the Lyapunov function

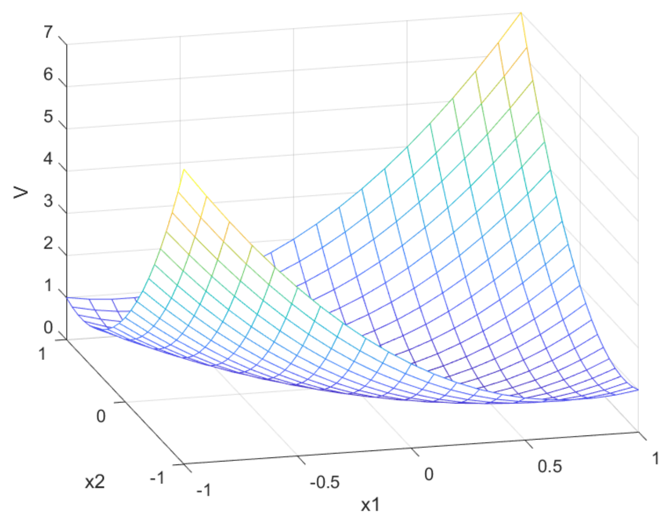
$$P = \begin{bmatrix} 1 & 0 \\ 0 & 20 \end{bmatrix}$$



$$P = \begin{bmatrix} 20 & 0 \\ 0 & 1 \end{bmatrix}$$



$$P = \begin{bmatrix} 2 & 1.5 \\ 1.5 & 2 \end{bmatrix}$$



P-QLF Stability Condition

Parameterized Quadratic Lyapunov Function (P-QLF)

$$V(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{x}^*)^T \mathbf{P} (\boldsymbol{x} - \boldsymbol{x}^*)$$

$$\mathbf{P} = \mathbf{P}^T \succ 0$$

How to ensure $\dot{V}(\boldsymbol{x})$ is always negative?

$$\dot{V}(\boldsymbol{x}) = \frac{\partial V}{\partial \boldsymbol{x}} \mathbf{f}(\boldsymbol{x}) < 0 \longrightarrow \mathbf{A}^T \mathbf{P} + \mathbf{P}\mathbf{A} \prec 0$$

Optimization of P-QLF – 1st formulation

Objective function: Maximum likelihood or Mean-square error

Constraints:

$$\begin{cases} b^k = -A^k x^* \\ \left(A^k\right)^T P + PA^k \prec 0 \end{cases} \quad \forall k = 1,\ldots,K$$

Joint estimation of P and A makes the problem non-convex
Depends on good initial guess for P.

Idea: decouple the problem in two-steps:

1) Estimate the A^k matrices with standard GMM
2) Estimate P in order to enforce the stability constraints

N. Figueroa and A. Billard.. A physically-consistent Bayesian non-parametric Mixture Model for dynamical system learning. CoRL. Oct 2018.

Learning Non-linear DS with GMM's and P-QLF

(Proposed Approach) We decouple the density estimation from the DS parameters

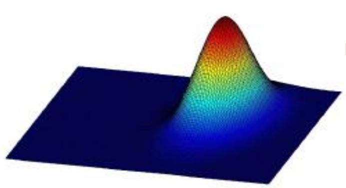
$$\mathbf{f}(\boldsymbol{x}) = \sum_{k=1}^{K} \gamma_k(\boldsymbol{x})(\mathbf{A}_k \boldsymbol{x} + \boldsymbol{b}_k)$$

Step 1: Learn the GMM density solely on position variables

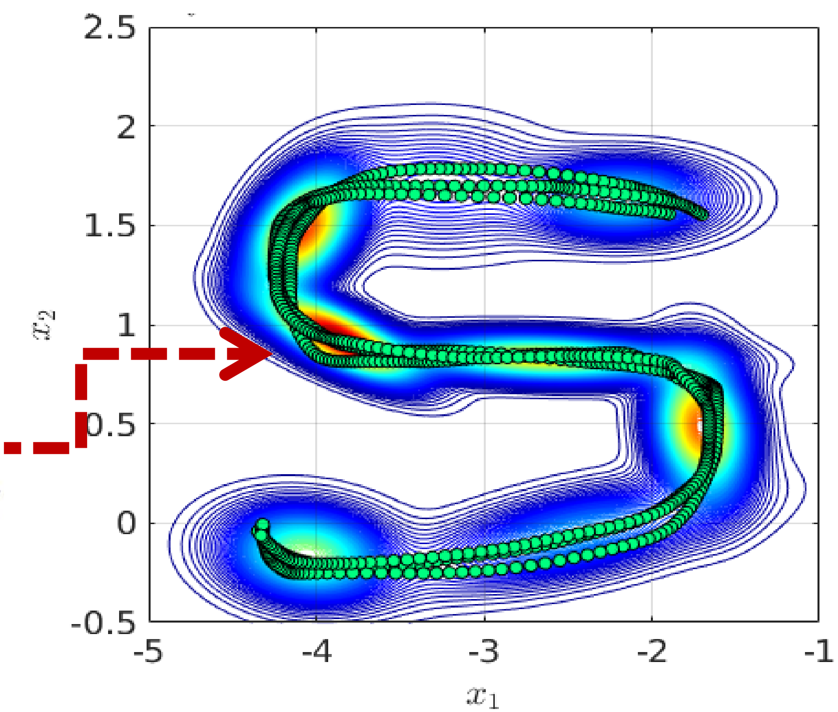
$$p(\boldsymbol{x}|\theta_\gamma) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k)$$
$$\theta_\gamma = \{\pi_k, \boldsymbol{\mu}^k, \boldsymbol{\Sigma}^k\}_{k=1}^{K}$$

$$\gamma_k(\boldsymbol{x}) = \frac{\pi_k p(\boldsymbol{x}|k)}{\sum_j \pi_j p(\boldsymbol{x}|j)}$$
$$\mathbf{A}_k = \boldsymbol{\Sigma}^k_{\boldsymbol{x}\dot{\boldsymbol{x}}}(\boldsymbol{\Sigma}^k_{\boldsymbol{x}})^{-1}$$
$$\boldsymbol{b}_k = \boldsymbol{\mu}^k_{\dot{\boldsymbol{x}}} - \mathbf{A}_k \boldsymbol{\mu}^k_{\boldsymbol{x}}$$

$p(x) \sim N(x; \mu, \Sigma)$



2D projection of a normal distribution



Learning Non-linear DS with GMM's and P-QLF

(Proposed Approach) We decouple the density estimation from the DS parameters

$$\mathbf{f}(\boldsymbol{x}) = \sum_{k=1}^{K} \gamma_k(\boldsymbol{x})(\mathbf{A}_k \boldsymbol{x} + \boldsymbol{b}_k)$$

Step 2: Estimate DS parameters via non-convex Semi-Definite Programming

$$\min_{\Theta_f} J(\Theta_f) = \min_{\Theta_f} \sum_{i=1}^{M} \left\| f(x^i) - \dot{x}^i \right\|$$

MSE Loss

$$\Theta_f = \{A_k, b_k\}_{k=1}^{K}$$

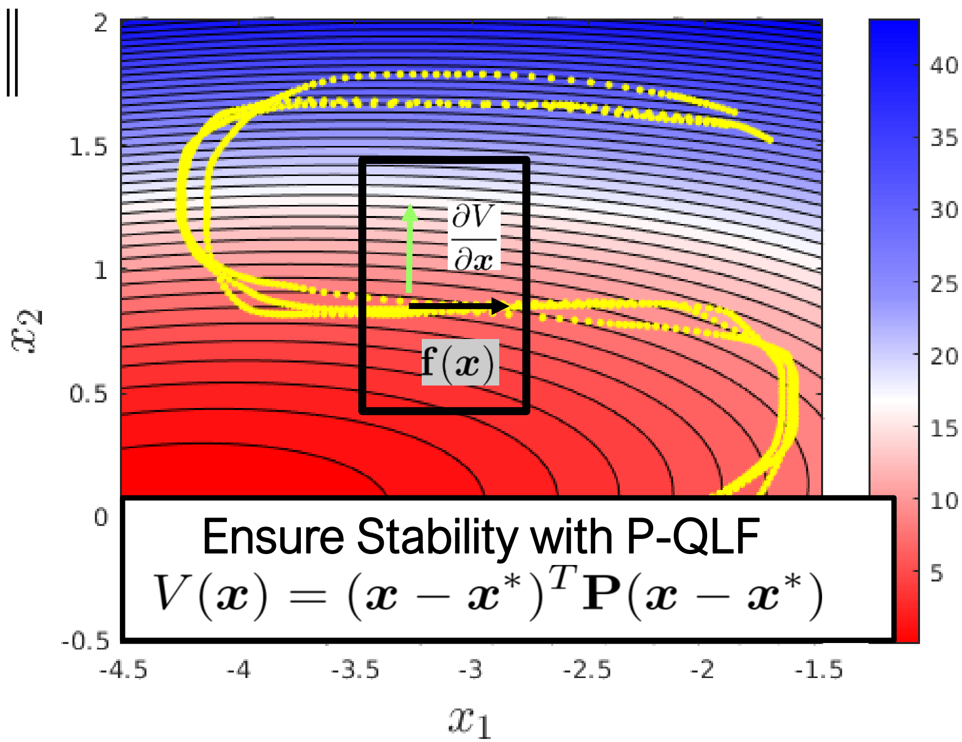
Stability Constraints

$$(\mathbf{A}_k)^T \mathbf{P} + \mathbf{P}\mathbf{A}_k \prec 0$$

$$\boldsymbol{b}_k = -\mathbf{A}_k \boldsymbol{x}^*$$

$$\mathbf{P} = \mathbf{P}^T \succ 0$$

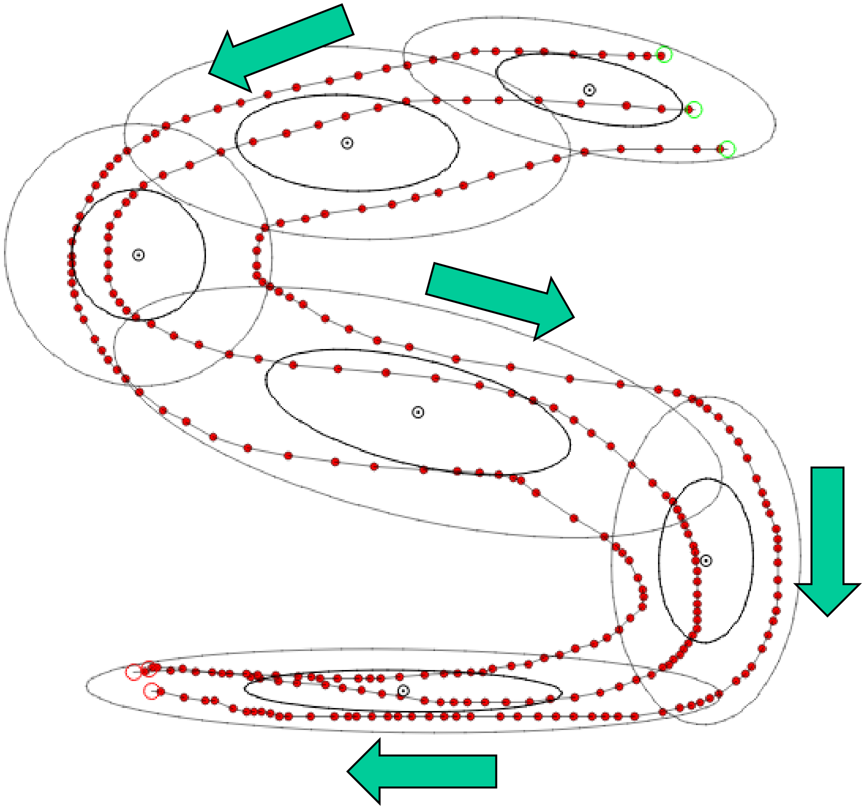
$$\forall\ k = 1, \ldots, K$$



Learning Non-linear DS with GMM's and P-QLF

(Caveat) Since the density estimation is decoupled, DS reproduction accuracy relies on whether the mixture of Gaussians fits well the dynamics of the data.

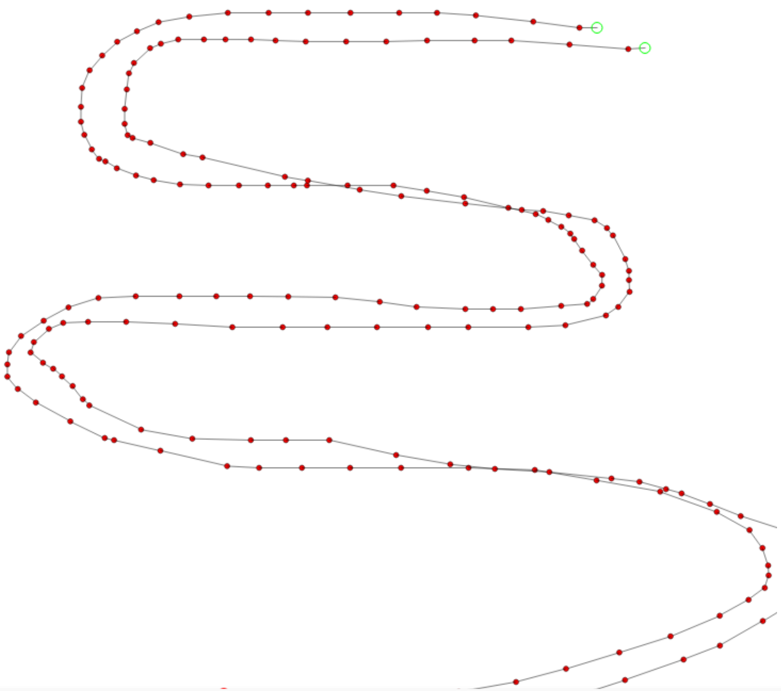
Aligns well with direction of curvature

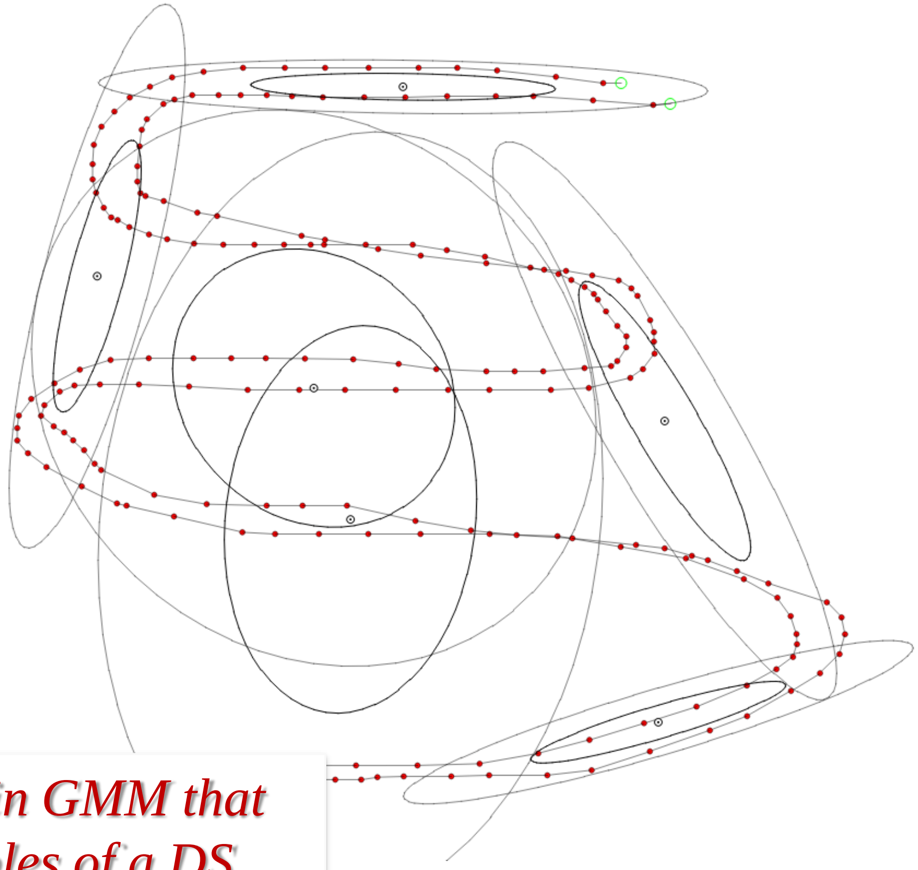


Learning Non-linear DS with GMM's and P-QLF

(Caveat) Since the density estimation is decoupled, DS reproduction accuracy relies on whether the mixture of Gaussians fits well the dynamics of the data.

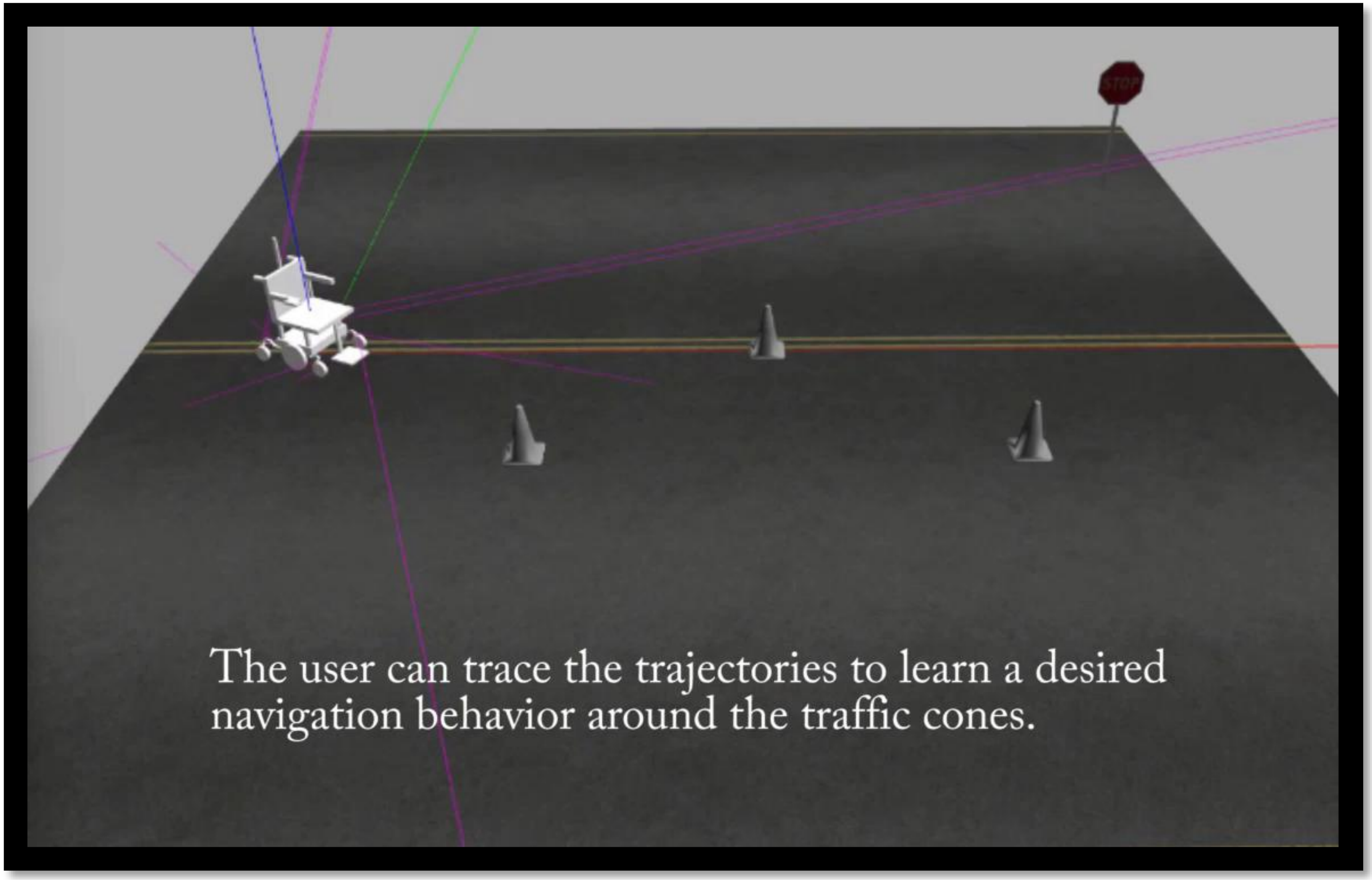
Not always the case, especially as nonlinearity of dataset increases





→ Need to devise a new procedure to train GMM that is informed by the fact that data are samples of a DS.

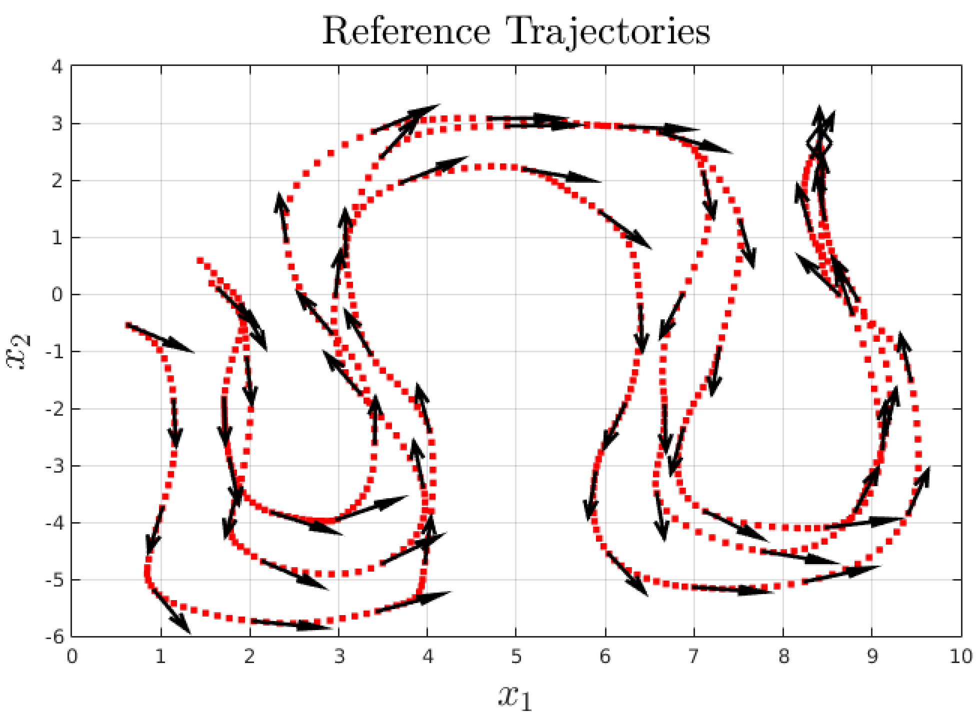
Learning Non-linear DS with GMM's and P-QLF



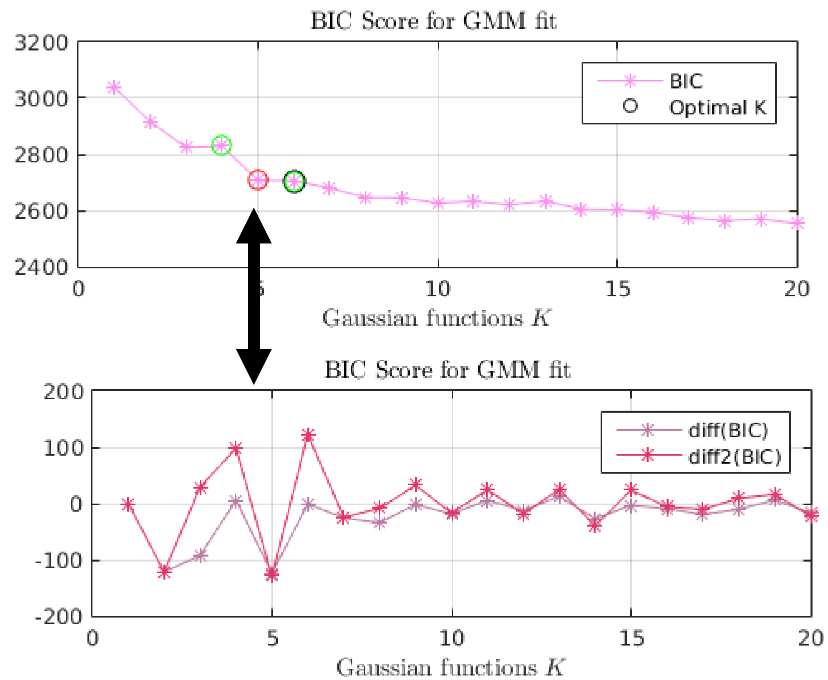
Example: Navigating Around Obstacles

Fit with traditional GMM training

Use classic EM estimation to fit the Gauss functions



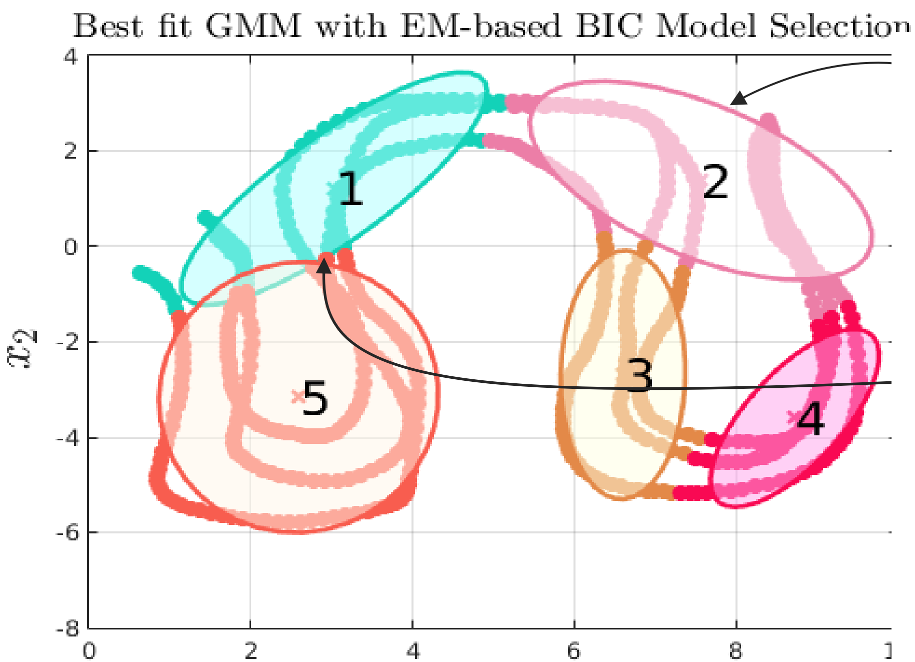
Use Bayesian Information Criterion (BIC) to determine optimal number of Gauss functions.

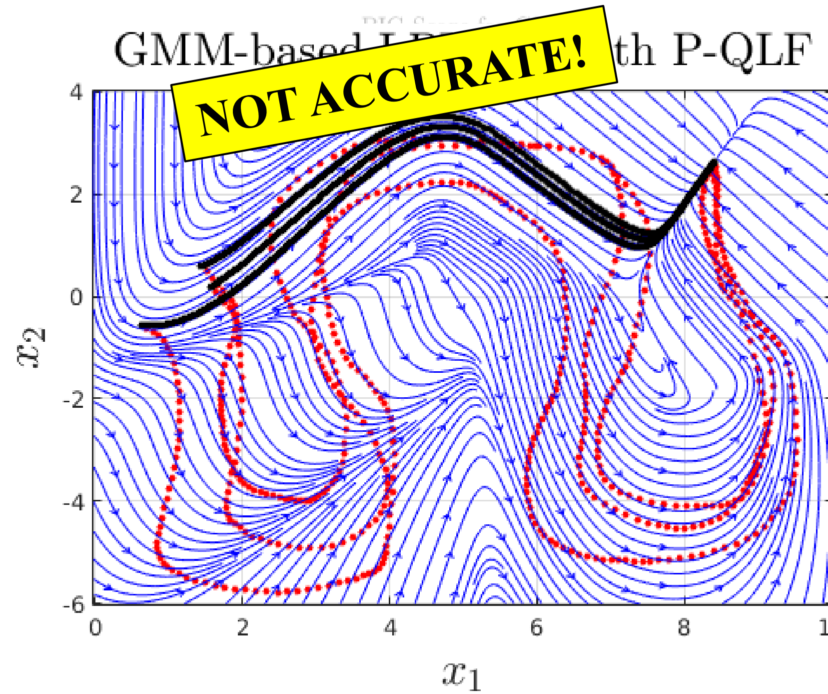


Repeat with different initial conditions and compare the fits.

Result from traditional GMM fit

Use classic EM estimation to fit the Gauss functions



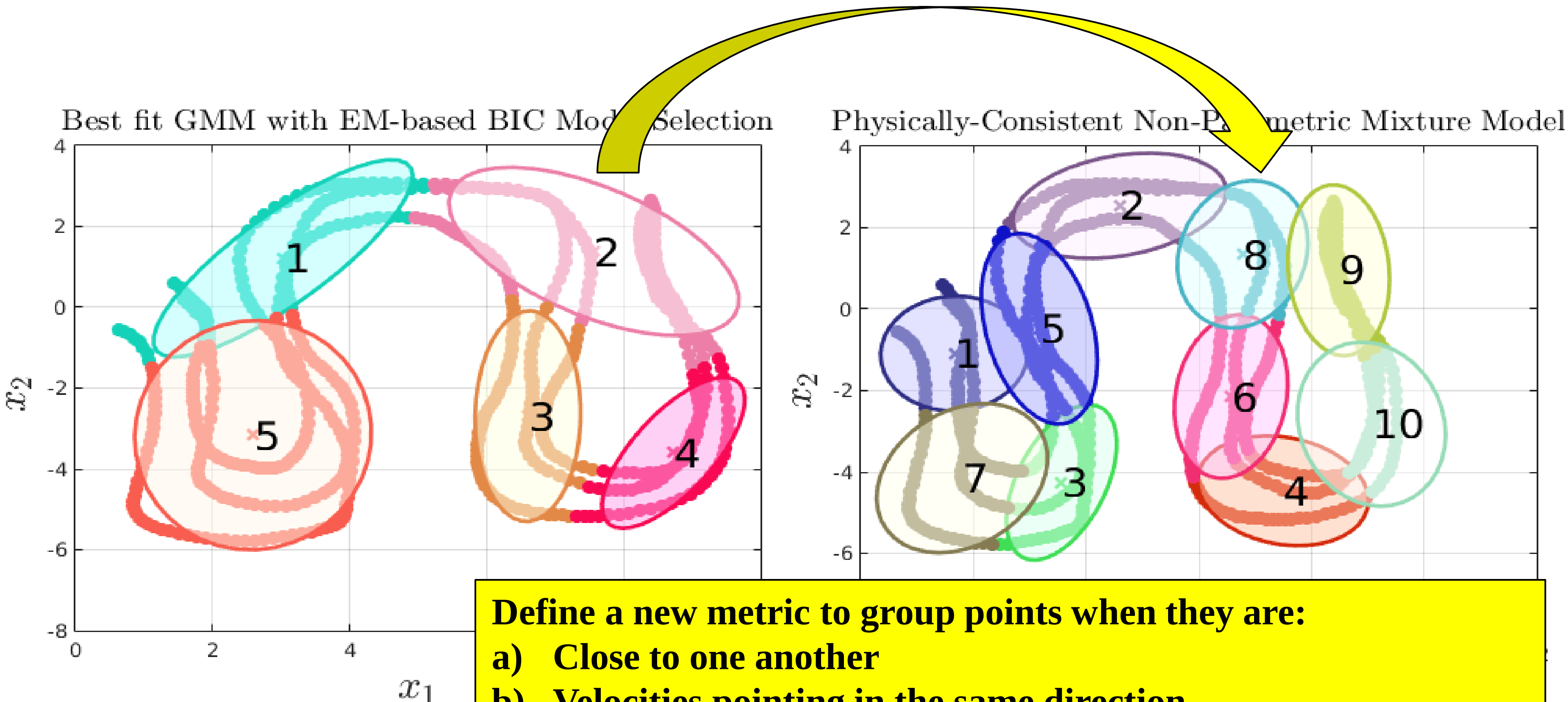


NOT PHYSICALLY CONSISTENT!

DO NOT FOLLOW ORDERING COMING FROM VELOCITY FLOW

Physically-Consistent GMM

IDEA: ALIGN GAUSS FUNCTION WITH VELOCITY FLOW



Define a new metric to group points when they are:

a) Close to one another

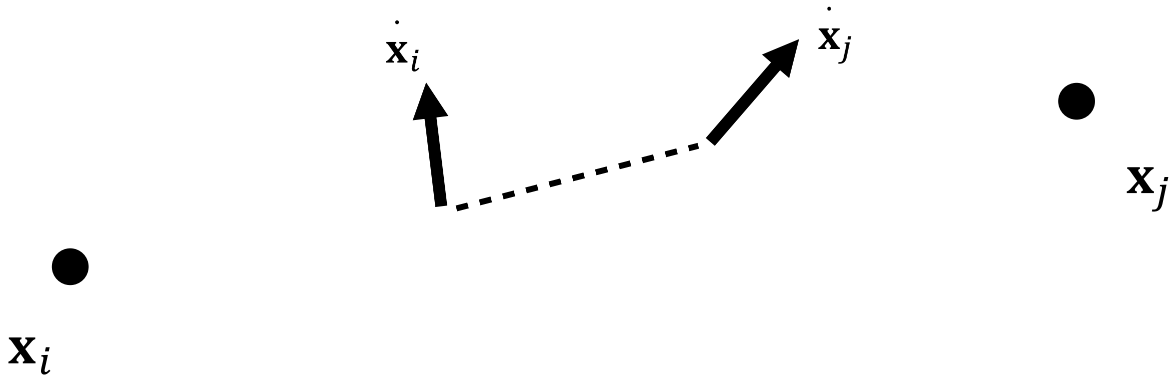
b) Velocities pointing in the same direction

Physically-Consistent GMM

Introduce a new metric

$$\Delta_{ij}(x^i, x^j, \dot{x}^i, \dot{x}^j) = \underbrace{\left(1 + \frac{(\dot{x}^i)^T \dot{x}^j}{||\dot{x}^i|| \, ||\dot{x}^j||}\right)}_{\text{Directionality}} \underbrace{\exp\left(-l ||x^i - x^j||^2\right)}_{\text{Locality}} .$$

$\gg 0$ $\quad$ $\gg 0$



Use this metric to assign datapoints to a Gauss function.

Bayesian Nonparametric Mixture Model

- **Bayesian**: Bayesian treatment of GMM training

→ No need to fix number of Gauss functions.

→ It learns both the GMM parameters and the number of these parameters required for an optimal fit of the data.

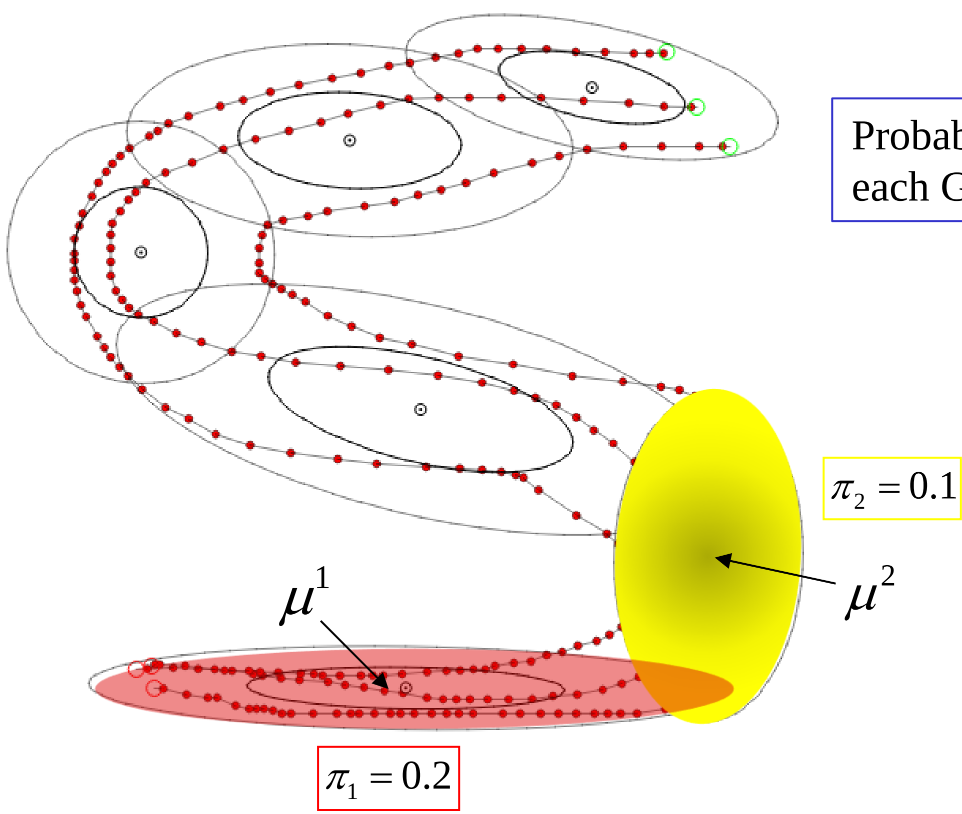
- ***Non-parametric*****:** Does NOT mean methods with “no parameters”, rather models whose complexity (# of states, # Gaussians) is inferred from the data.
    - Number of parameters grows with sample size.
    - Infinite-dimensional parameter space!

Recall: GMM Clustering Assignment

(see Machine Learning I course on clustering with GMM)

Likelihood of the mixture of K Gaussians: $L\left(\Theta=\left\{\pi_k,\mu^k,\Sigma^k\right\}_{k=1}^{K};x\right)=\sum_{k=1}^{K}\pi_k\cdot p\left(x;\mu^k,\Sigma^k\right)$

- π_k: Probability associated to each Gauss function
- μ^k: Center of Gauss function
- Σ^k: Length and orientation of distribution



The number of clusters K is a hyperparameter, sometimes difficult to determine.

$\rightarrow$ *Bayesian* treatment of GMM

Bayesian Nonparametric Mixture Model

1: Set priors on model parameters

Normal-inverse-Wishart distribution

See supplement on moodle on conjugate Bayesian analysis of the Gaussian distribution

$$\Theta = \left\{ \pi_k, \mu^k, \Sigma^k \right\}_{k=1}^{K}$$

Dirichlet Prior

The number of Gauss function is unknown and infinite,

$\Rightarrow K \rightarrow \infty$

The Dirichlet Process is used as a non-parametric prior on the mixing coefficients π.

It removes the need to specify K.

2: Use *Bayesian inference* to estimate the parameters.

See Book's Annexes B.3.2-3.3 for details

E-M Traditional GMM

Set the number of clusters K

Soft Assignment:

r_i^k: responsibility of cluster k for point x^i

$$r_i^k = \frac{\pi_k p\left(x^i; \mu^k, \sigma_j^k\right)}{\sum_{k'} \pi_k p\left(x^i; \mu^{k'}, \sigma_j^{k'}\right)}$$

Determine automatically cluster assignment, through maximum likelihood

Draw cluster assignment according to norm-2 distance.

Bayesian Nonparametric Mixture Model

Number of clusters could be infinite and is drawn from a distribution.

Hard assigment, we denote:

$z_i = k$ as the assignment of point x^i to cluster k

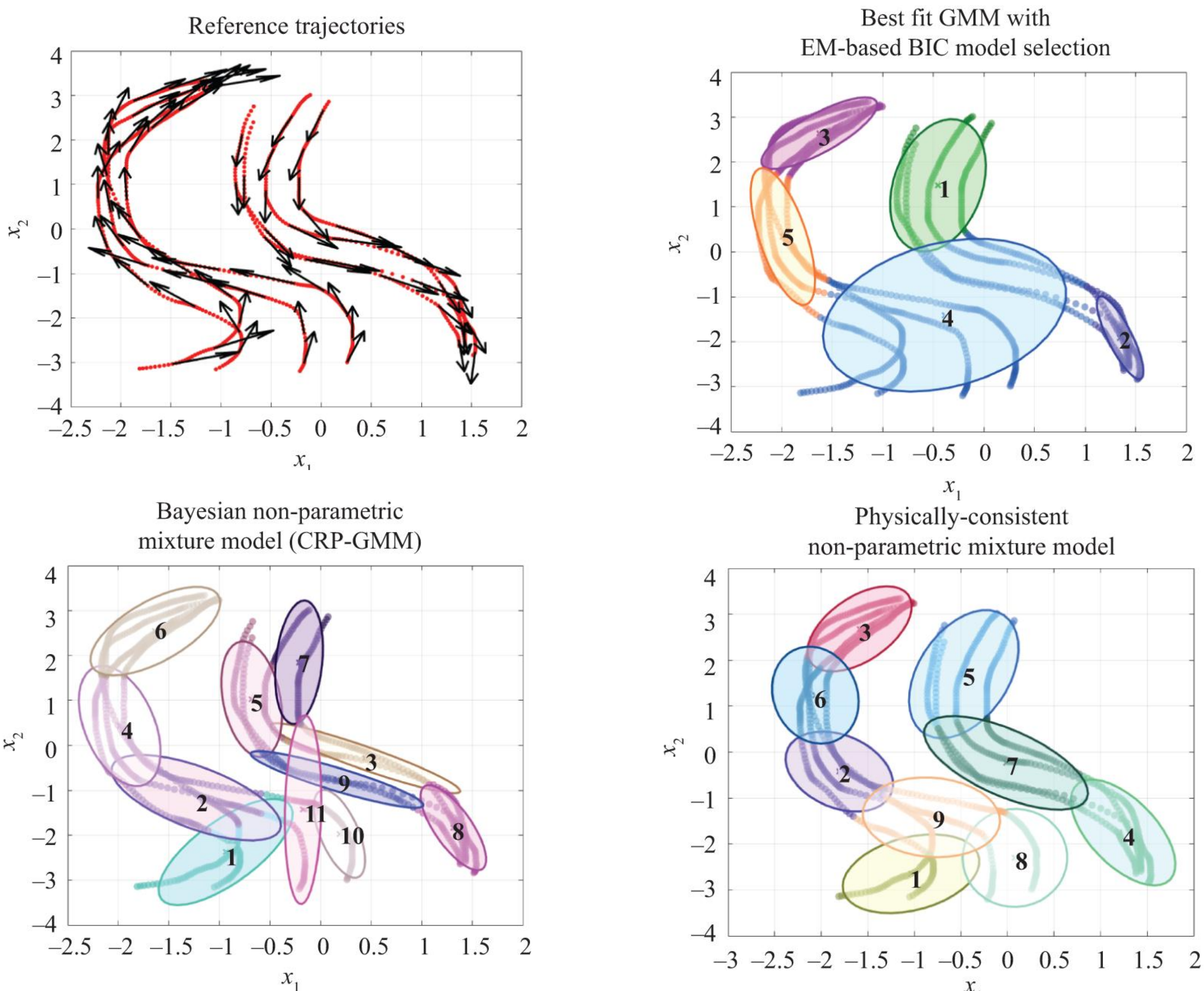
Determine automatically number of clusters and cluster assignment, through maximum likelihood.

Draw cluster assignment according to how close datapoints are under new metric:

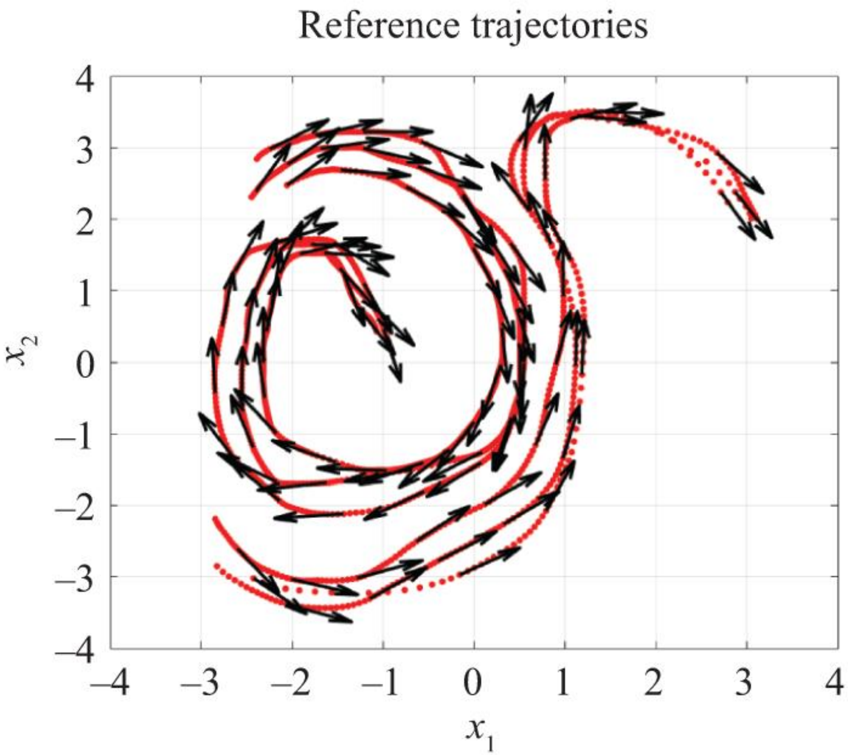
$$\Delta_{ij}(x^i, x^j, \dot{x}^i, \dot{x}^j) = \underbrace{\left(1 + \frac{(\dot{x}^i)^T \dot{x}^j}{||\dot{x}^i||||\dot{x}^j||}\right)}_{\text{Directionality}} \underbrace{\exp\left(-l||x^i - x^j||^2\right)}_{\text{Locality}}.$$

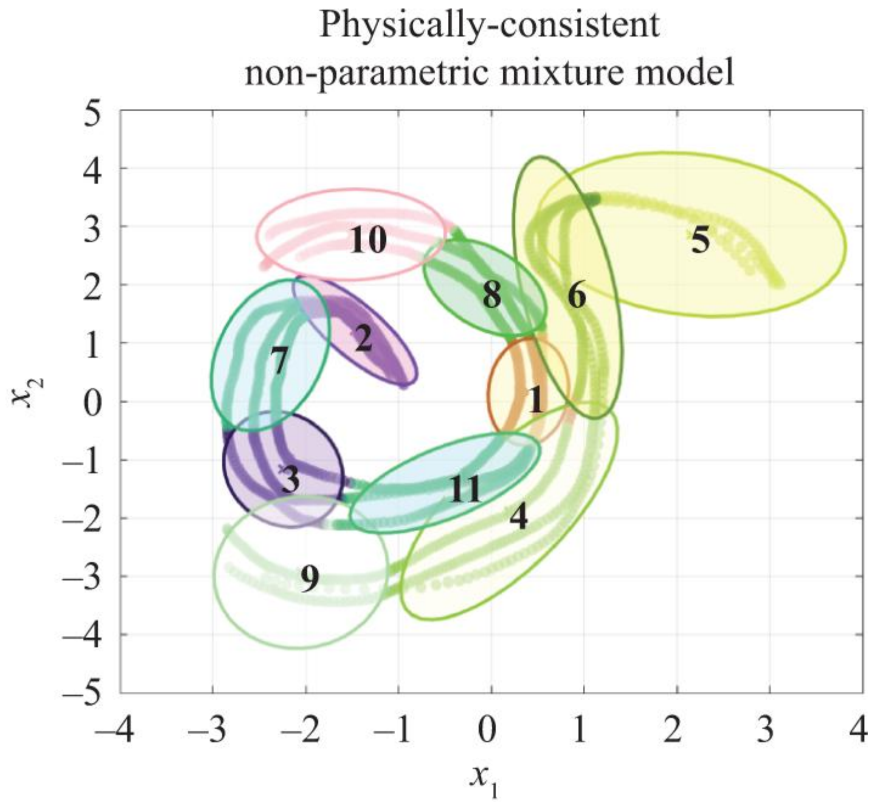
EPFL

Examples: Physically-Consistent GMM

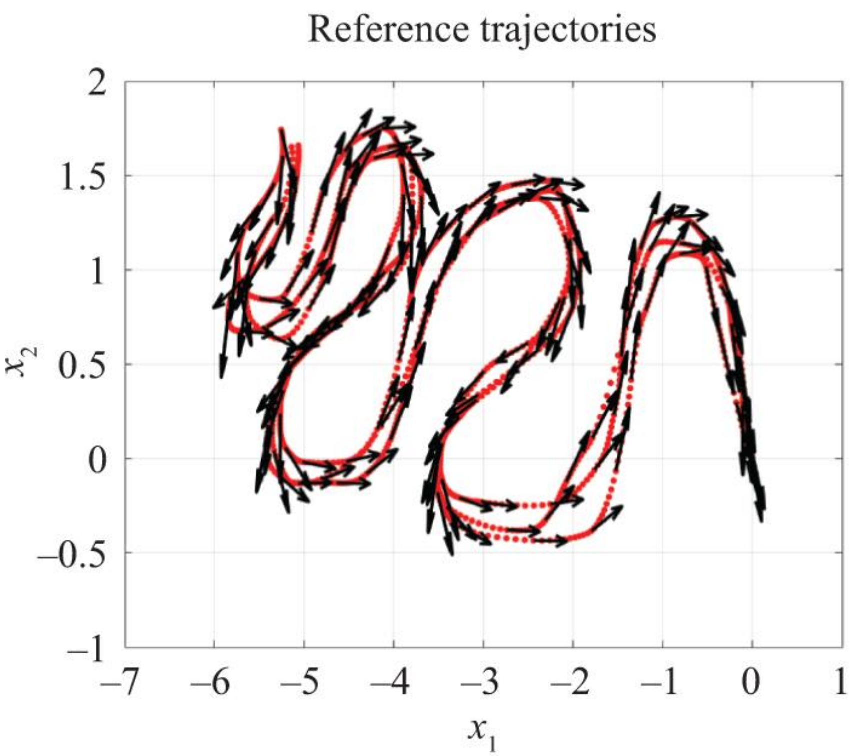


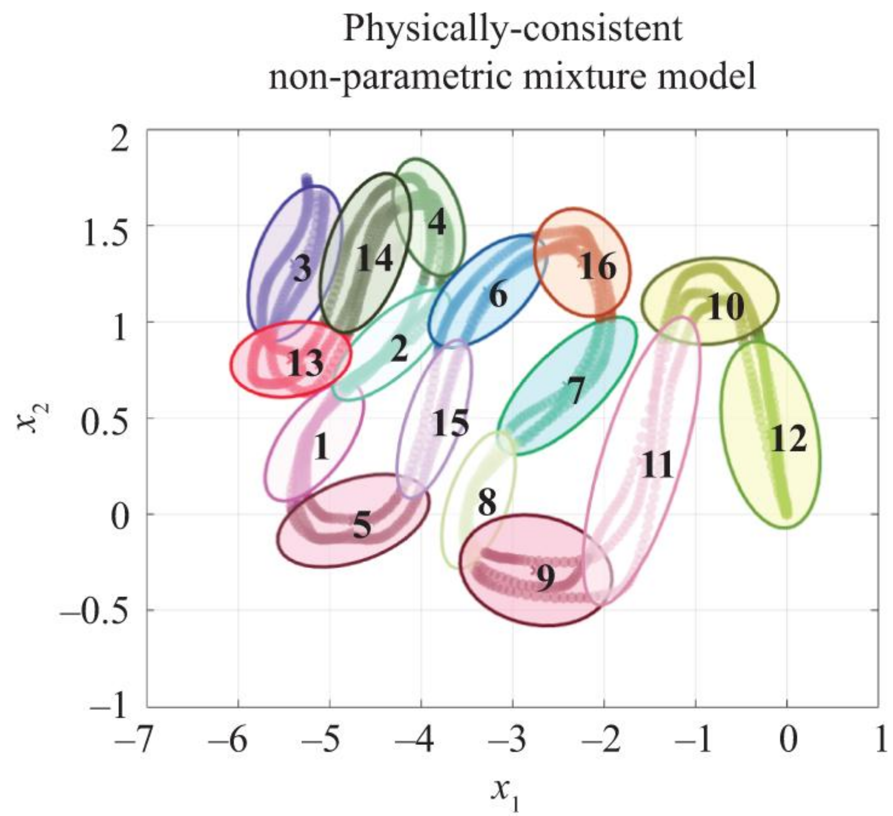
Examples: Physically-Consistent GMM





Examples: Physically-Consistent GMM





LPV-DS final optimization

Once the GMM parameters have been estimated with PC-GMM, we are left with satisfying the set of constraints for stability.

This leads to a non-convex but solvable optimization (see Section 3.4.3 of the book for details).

SEDS like

$$\min_{\Theta_f} J(\Theta_f) \quad \text{subject to}$$

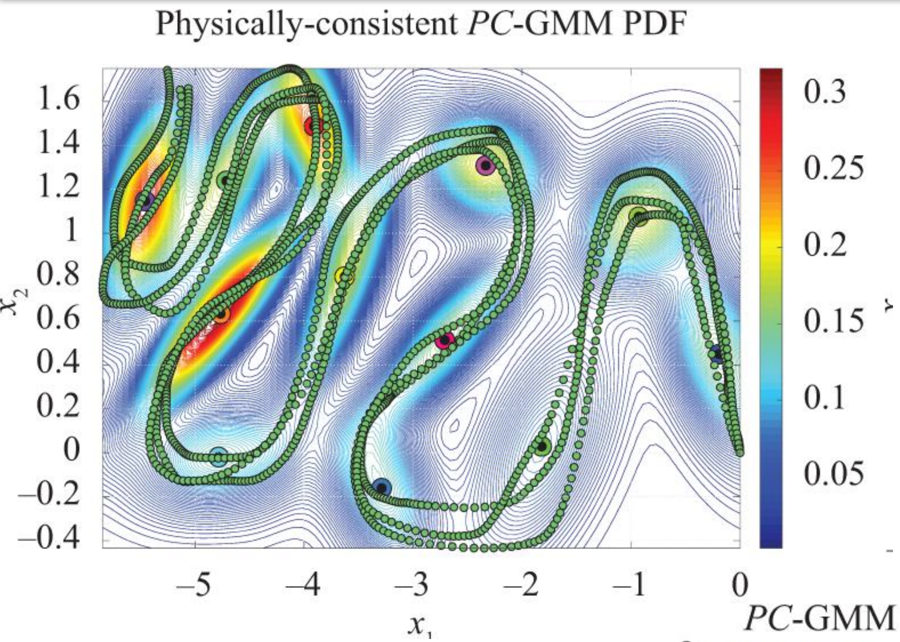
$$(O1)\left\{(A^k)^T + A^k \prec 0, b^k = -A^k x^* \;\; \forall k = 1, \ldots, K\right.$$

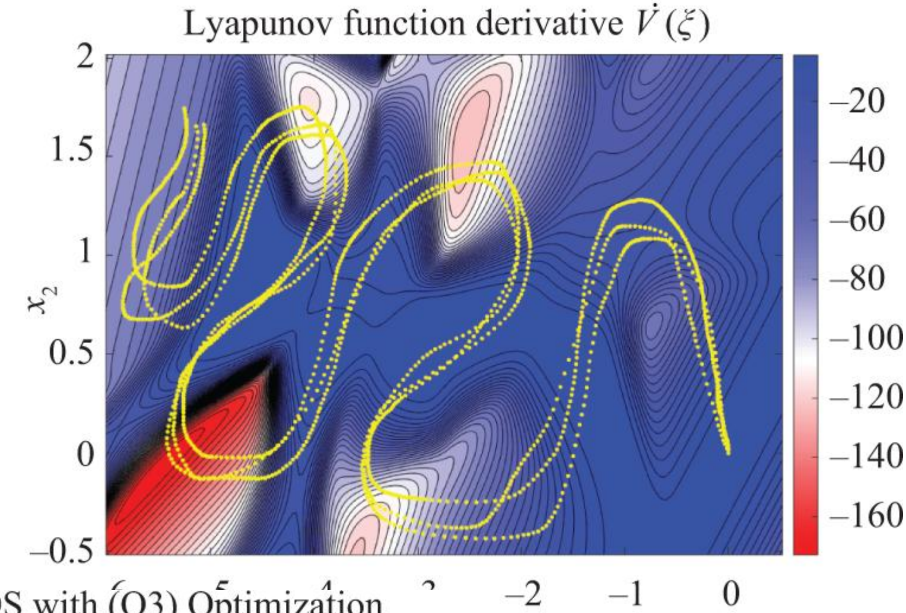
P-QLF

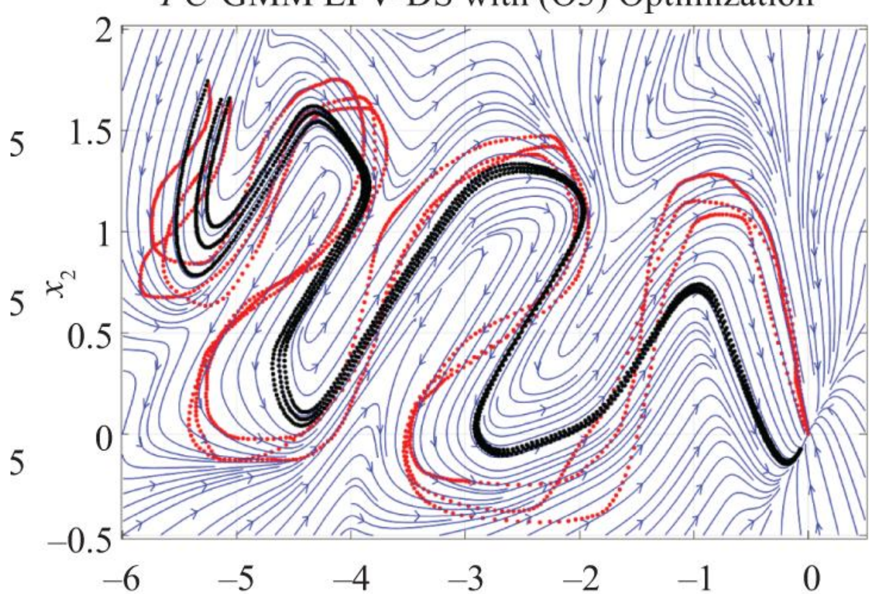
$$(O2)\left\{(A^k)^T P + PA^k \prec 0, \;\; b^k = \mathbf{0} \;\; \forall k = 1, \ldots, K; \;\; P = P^T \succ 0\right.$$

$$(O3)\left\{(A^k)^T P + PA^k \prec Q^k, \;\; Q^k = (Q^k)^T \prec 0, \;\; b^k = -A^k x^* \;\; \forall k = 1, \ldots, K.\right.$$

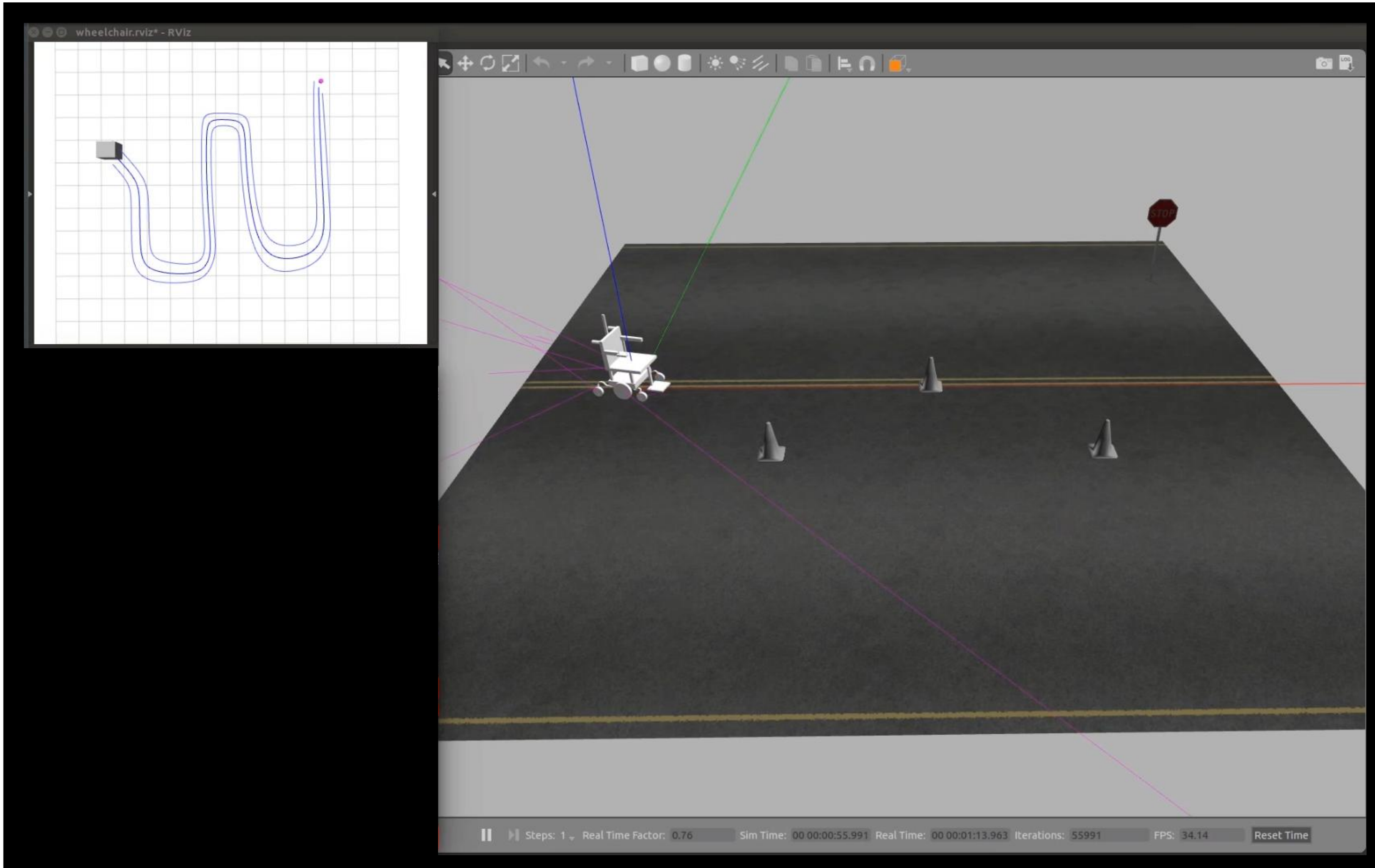
LPV-DS final optimization





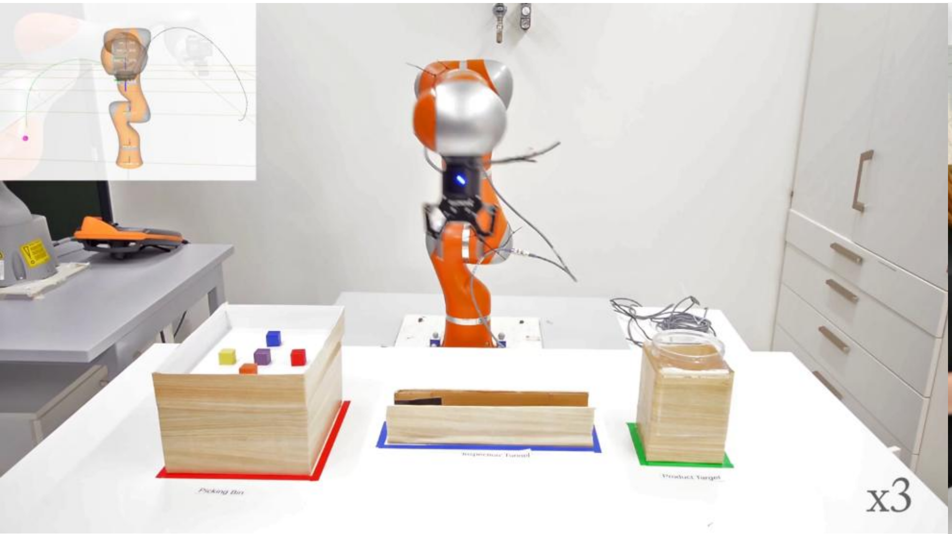


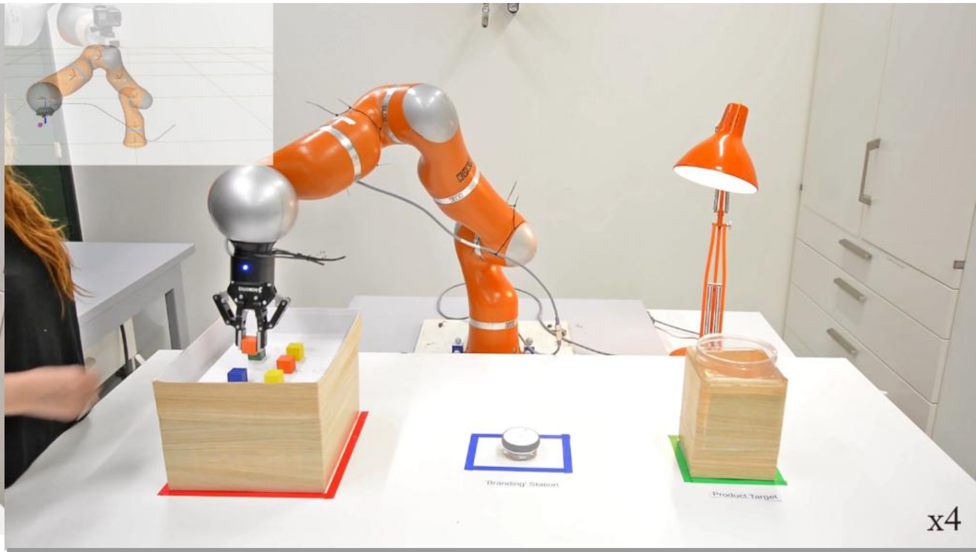
Result on previous example

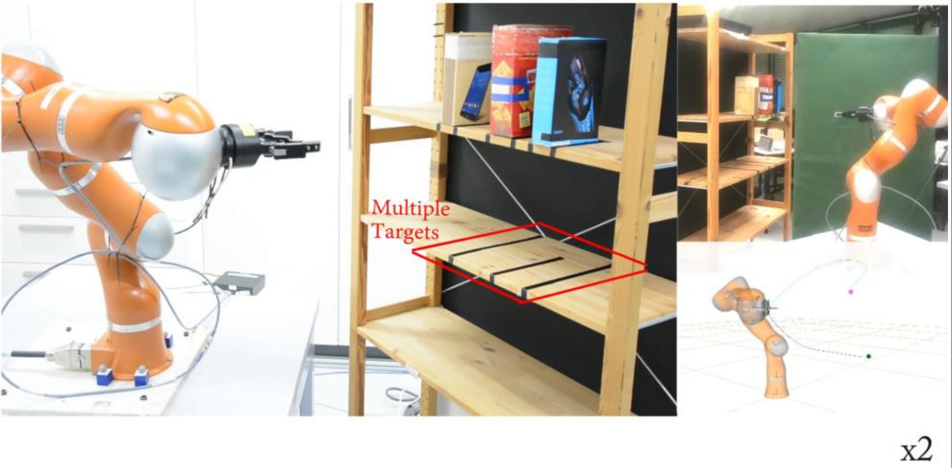


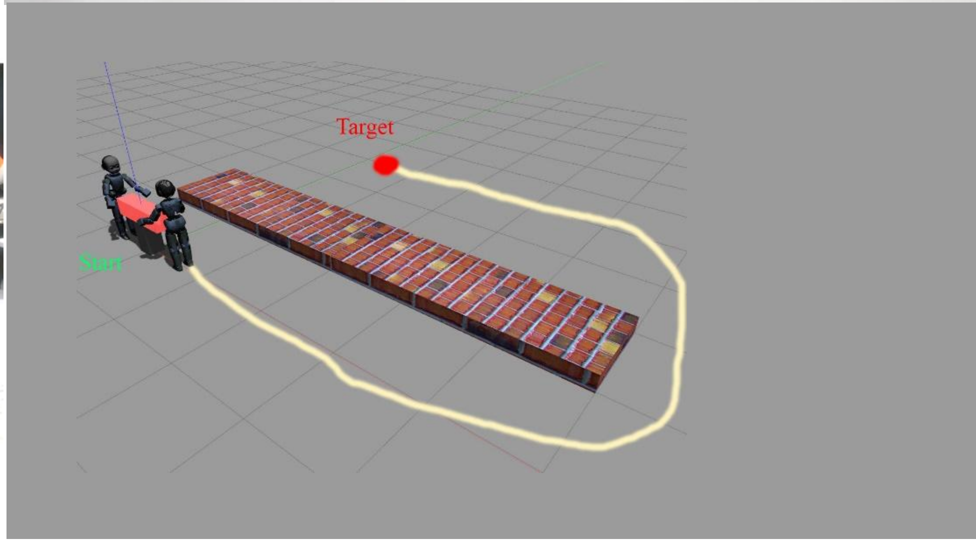
N. Figueroa and A. Billard. . A physically-consistent Bayesian non-parametric Mixture Model for dynamical system learning. CoRL. Oct 2018.

Learning Physically-Consistent Gaussian Mixture Model









N. Figueroa and A. Billard. . *A physically-consistent Bayesian non-parametric Mixture Model for dynamical system learning. CoRL. Oct 2018.*

Summary LPV-DS

LPV-DS was offered as an alternative to SEDS to enable learning of more complex, and nonlinear DS from demonstrations.

SEDS	LPV - DS
Fix by hand number of Gaussians	**Learns automatically number of Gaussians**
Conservative stability constraints → Cannot learn highly non-linear trajectories	**Less conservative stability constraints → Can embed large non-linearities**