Trajectory Inverse Kinematics By Conditional Density Models

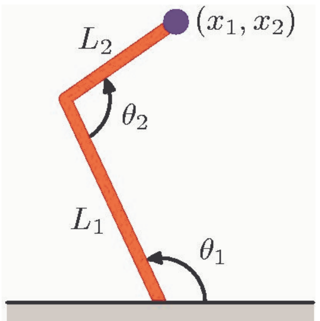
Chao Qin and Miguel Á. Carreira-Perpiñán

EECS, School of Engineering, UC Merced

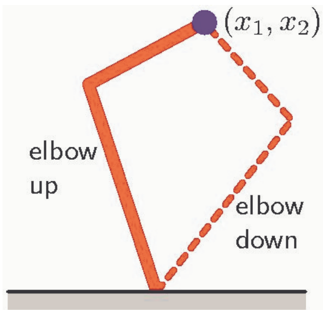
Introduction

- Robot arm inverse kinematics (IK)
  - Infer joint angles $\boldsymbol{\theta}$ from positions of the end-effector $\boldsymbol{x}$
- Pointwise IK: $\boldsymbol{\theta} = \boldsymbol{f}^{-1}(\boldsymbol{x})$
  - Univalued forward mapping: $\boldsymbol{f} : \boldsymbol{\theta} \rightarrow \boldsymbol{x}$
  - Multivalued inverse mapping: $\boldsymbol{f}^{-1} : \boldsymbol{x} \rightarrow \boldsymbol{\theta}$
- Examples

Planar 2-link arm

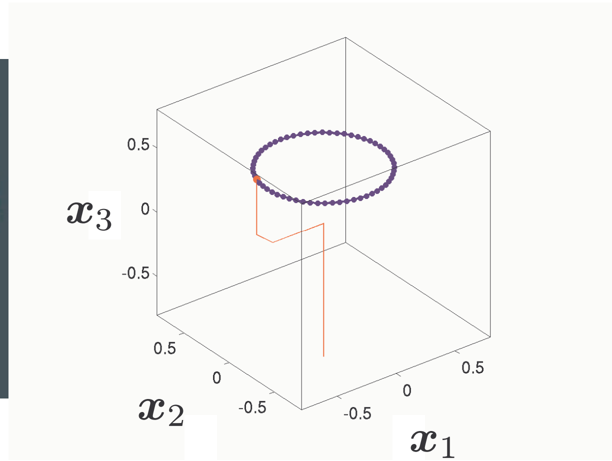


forward kinematics



inverse kinematics

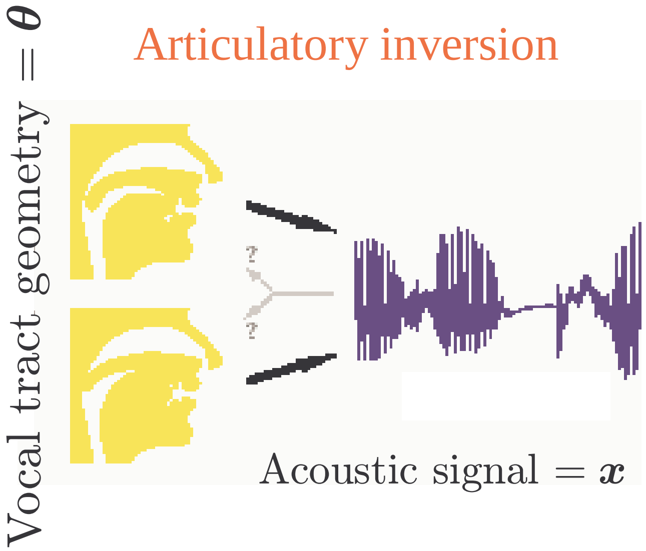
PUMA 560



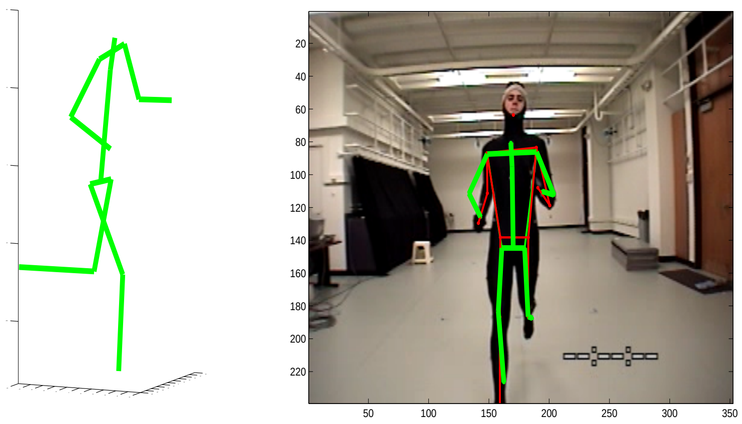
Introduction

- Trajectory IK
  - Given a sequence of positions $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$ in Cartesian workspace of the end-effector, we want to obtain a feasible sequence of joint angles $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_N$ that produce the $\boldsymbol{x}$-trajectory
- Difficulties
  - Multivalued inverse mapping (e.g. elbow up; elbow down)
  - $\boldsymbol{\theta}$-trajectory must be globally feasible, e.g. avoiding discontinuities or forbidden regions
- Trajectory IK in other areas

Articulatory inversion

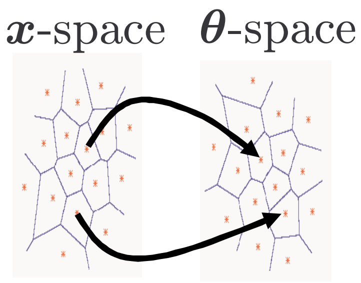


Articulated pose tracking



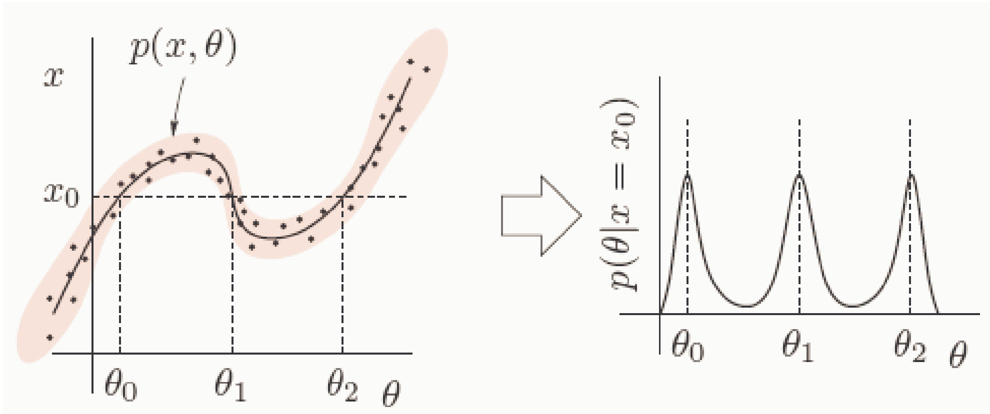
Traditional approaches and their problems

- Analytical methods (PaulZhang'86): only possible for simple arms
- Local methods
  - Jacobian pseudoinverse (Whitney'69, Liegeois'77)
    - Linearizes the forward mapping: $\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{\theta}) \to \dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} \to \dot{\boldsymbol{\theta}} = J^{+}(\boldsymbol{\theta})\dot{\boldsymbol{x}}$
    - Breaks down at singularity: $\boldsymbol{J}(\boldsymbol{\theta})$ becomes singular
    - High cost and numerical error accumulates
  - Analysis-by-synthesis: $\boldsymbol{\theta}^{*} = \mathrm{argmin}_{\boldsymbol{\theta}} \|\boldsymbol{x} - \boldsymbol{f}(\boldsymbol{\theta})\|^2$
- Global methods (Nakamura&Hanafusa'87, Martin et al'89)
  - Use variational approaches: $\min \int_{t_0}^{t_1} G(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}, t)\, dt$
  - Need boundary conditions
  - Still have problems with singularities



- Machine learning methods
  - Neural network
  - Distal learning (Jordan&Rumelhart'92)
  - Ensemble neural network (DeMers&Kreutz-Delgado'96, DeMers&Kreutz-Delgado'98)
  - Locally weighted linear regression (D'Souza et al'01)

Trajectory IK by conditional density modes



- Derive the multivalued functional relationship $\boldsymbol{f}^{-1}: \boldsymbol{x} \to \boldsymbol{\theta}$ from the conditional dist $p(\boldsymbol{\theta}|\boldsymbol{x})$

– Estimate (offline) $p(\boldsymbol{\theta}|\boldsymbol{x})$ from a training set $\{(\boldsymbol{\theta}_i, \boldsymbol{x}_i)\}$

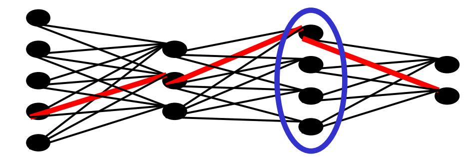
– Online, given $\boldsymbol{x}$-trajectory, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$

1. for $n = 1, \ldots, N$

   find all modes from $p(\boldsymbol{\theta}|\boldsymbol{x} = \boldsymbol{x}_n)$

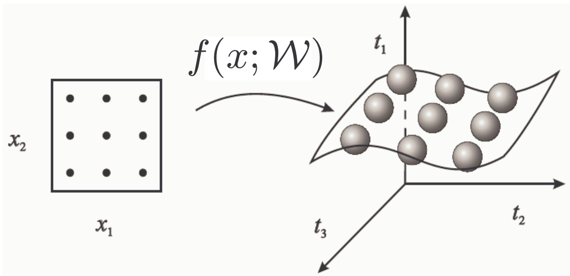
2. Search in the graph over all modes to minimize

$$\underbrace{\sum_{n=1}^{N-1} \|\boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n\|}_{\text{continuity constraint}} + \lambda \underbrace{\sum_{n=1}^{N} \|\boldsymbol{x}_n - \boldsymbol{f}(\boldsymbol{\theta}_n)\|}_{\text{forward constraint}}$$



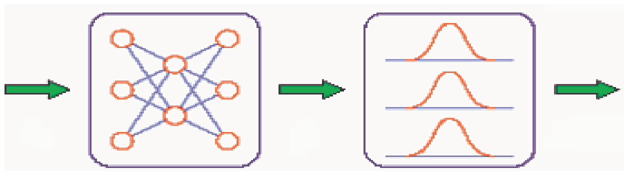
Offline step: learning conditional density $p(\boldsymbol{\theta}|\boldsymbol{x})$

- Given a training set $\{(\boldsymbol{\theta}_i, \boldsymbol{x}_i)\}$, estimate $p(\boldsymbol{\theta}|\boldsymbol{x})$ by:
  - Learning the full density $p(\boldsymbol{\theta}, \boldsymbol{x})$. We use Generative Topographic Mapping (GTM)
    - A constrained Gaussian mixture in space $(\boldsymbol{\theta}, \boldsymbol{x})$



$t = (\boldsymbol{\theta}, \boldsymbol{x})$

  - Learning directly $p(\boldsymbol{\theta}|\boldsymbol{x})$. We use Mixture Density Network (MDN)
    - A combination of neural network and Gaussian mixture

$\boldsymbol{x}$  $p(\boldsymbol{\theta}|\boldsymbol{x}) = GM : \{\pi_m(\boldsymbol{x}), \mu_m(\boldsymbol{x}), \sigma_m(\boldsymbol{x})\}$

– Advantages
  - Represent inverses by modes from the conditional density
  - Deal with topological changes naturally (modes split/merge)

Online steps

1. Finding modes of $p(\boldsymbol{\theta}|\boldsymbol{x})$ by *Gaussian mean-shift (GMS)* (Carreira-Perpinan'00)
   - Start from every centroid of the GM and iterate

$$\boldsymbol{\theta}^{(\tau+1)} = \sum_{m=1}^{M} p(m|\boldsymbol{\theta}^{(\tau)}; \boldsymbol{x}) \boldsymbol{\mu}_m(\boldsymbol{x})$$

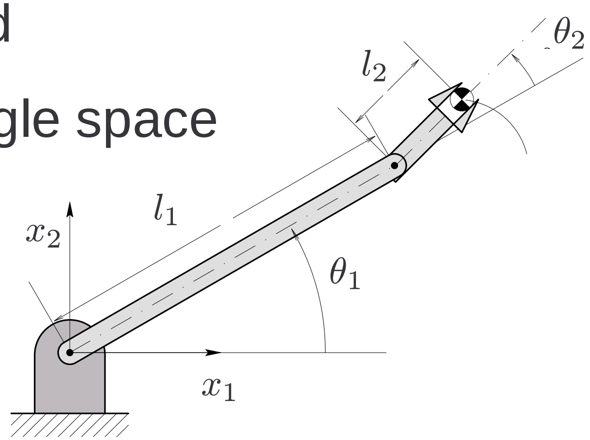
$$p(m|\boldsymbol{\theta}^{(\tau)}; \boldsymbol{x}) \propto \pi_m(\boldsymbol{x}) \exp\left(-\frac{\|\boldsymbol{\theta}^{(\tau)} - \boldsymbol{\mu}_m(\boldsymbol{x})\|^2}{2 \cdot \sigma_m(\boldsymbol{x})^2}\right)$$

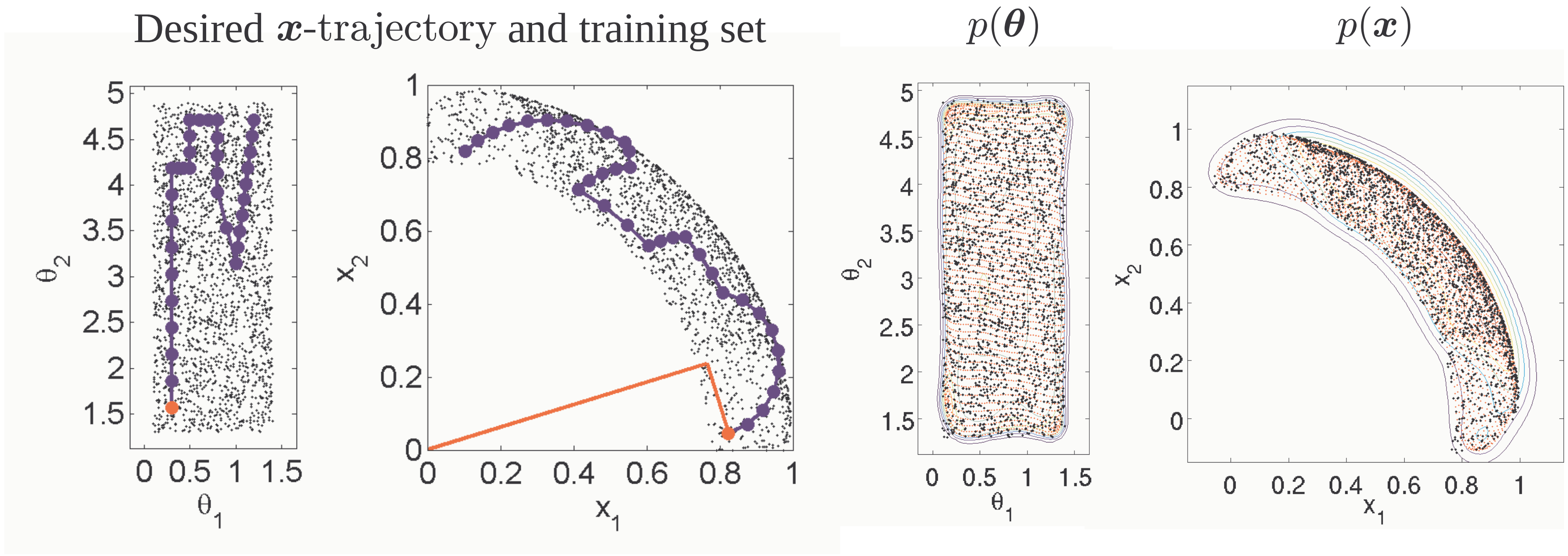
   - Complexity: $\mathcal{O}(kNM^2)$

2. Obtaining a unique $\boldsymbol{\theta}$-trajectory by global optimization
   - Minimize $\mathcal{C} + \lambda \mathcal{F}$ $(\lambda \geq 0)$ over the set of modes with dynamic programming
     - $\mathcal{C} = \sum_{n=1}^{N-1} \|\boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n\|$: *continuity constraint* (integrated 1st derivative) penalizes sudden angle changes
     - $\mathcal{F} = \sum_{n=1}^{N} \|\boldsymbol{x}_n - \boldsymbol{f}(\boldsymbol{\theta}_n)\|$: *forward constraint* (integrated workspace error) penalizes spurious inverses
   - Complexity: $\mathcal{O}(N\nu^2)$

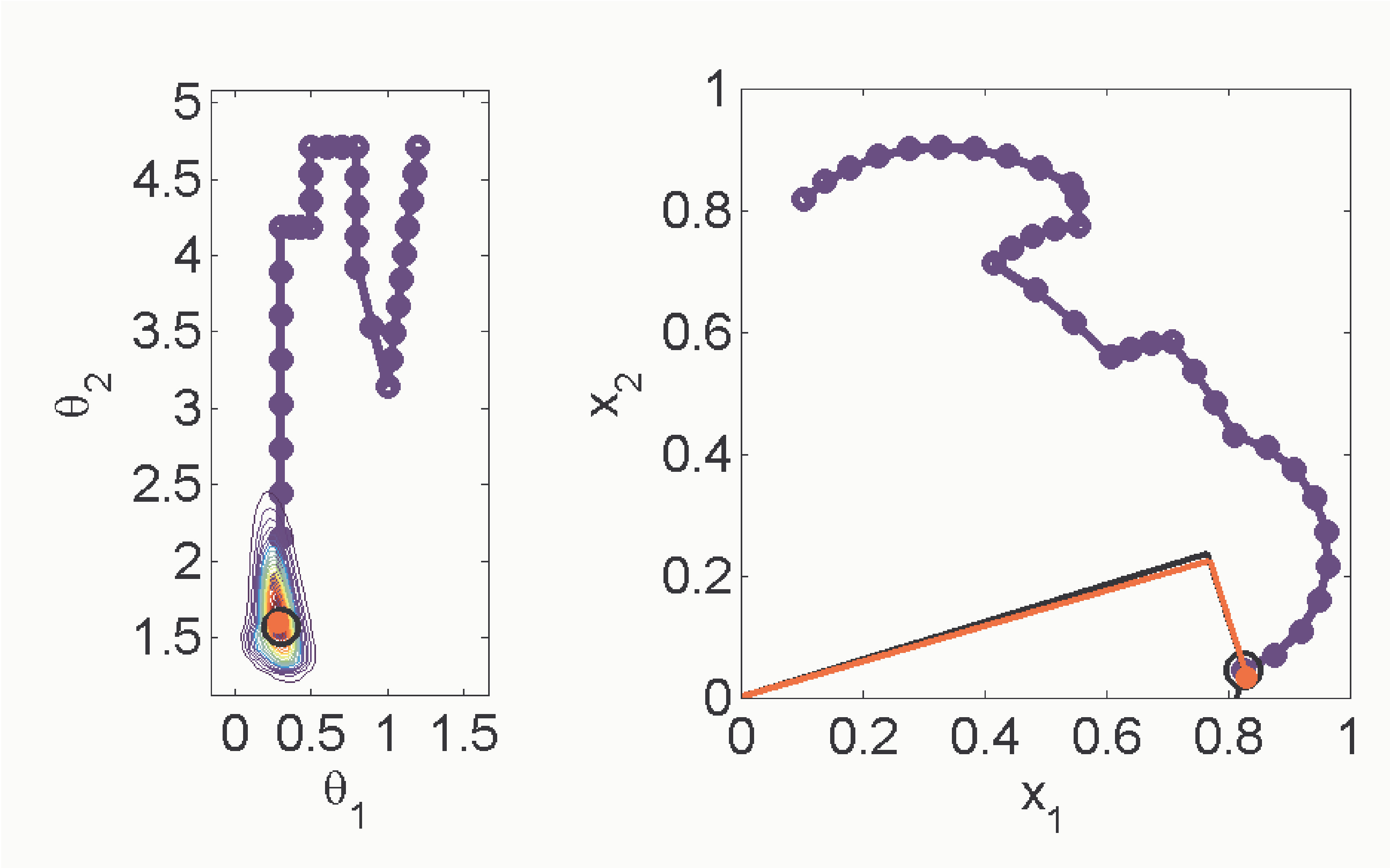
Experiments: planar 2-link robot arm

- Limit the angle domain to [0.3,1.2]x[1.5,4.7] rad
- Generate 2000 pairs by uniformly sampling angle space
- Train density models:
  - GTM: M=225 and 2500 components
  - MDN: M=2 components and 10 hidden units

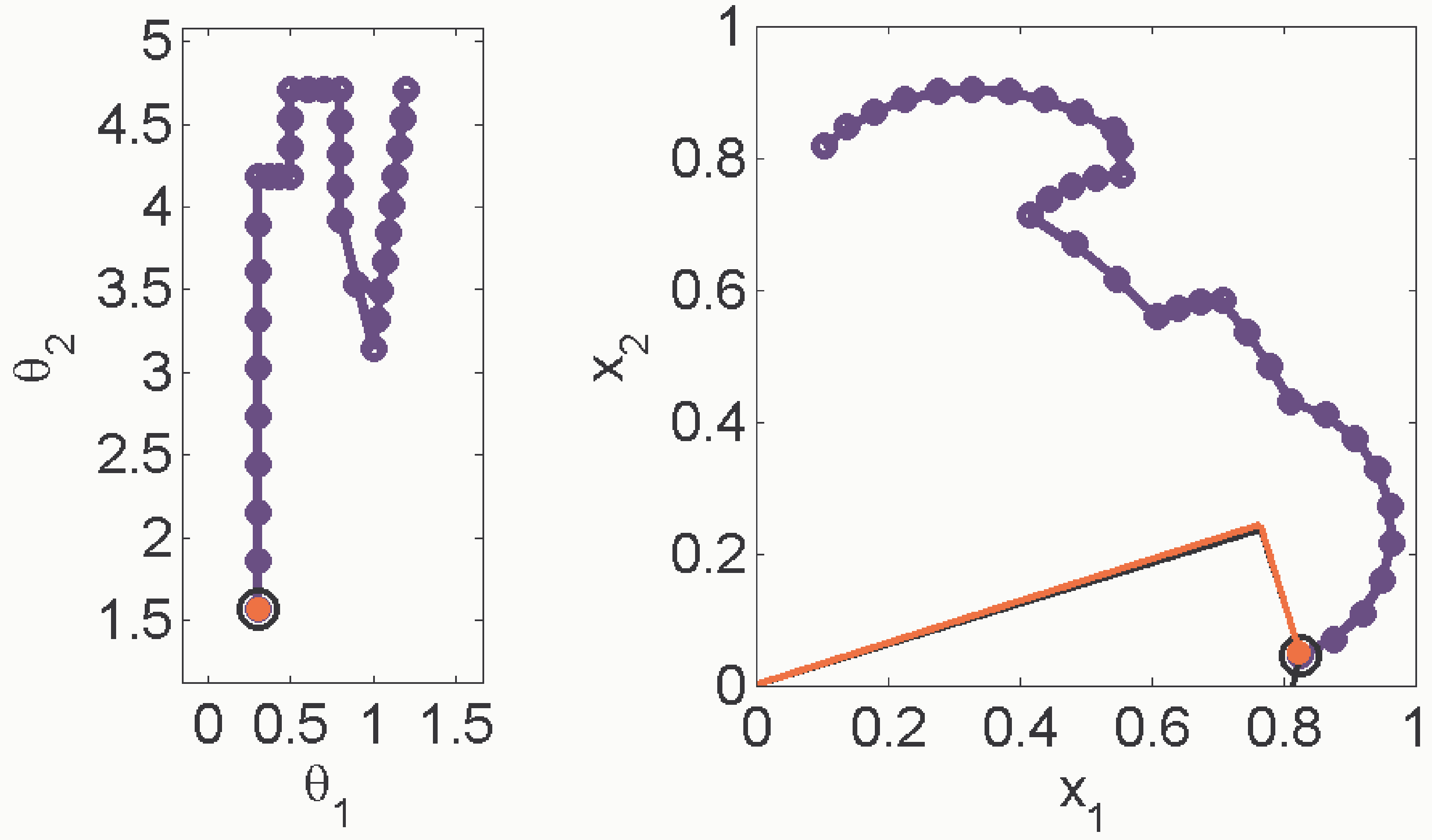




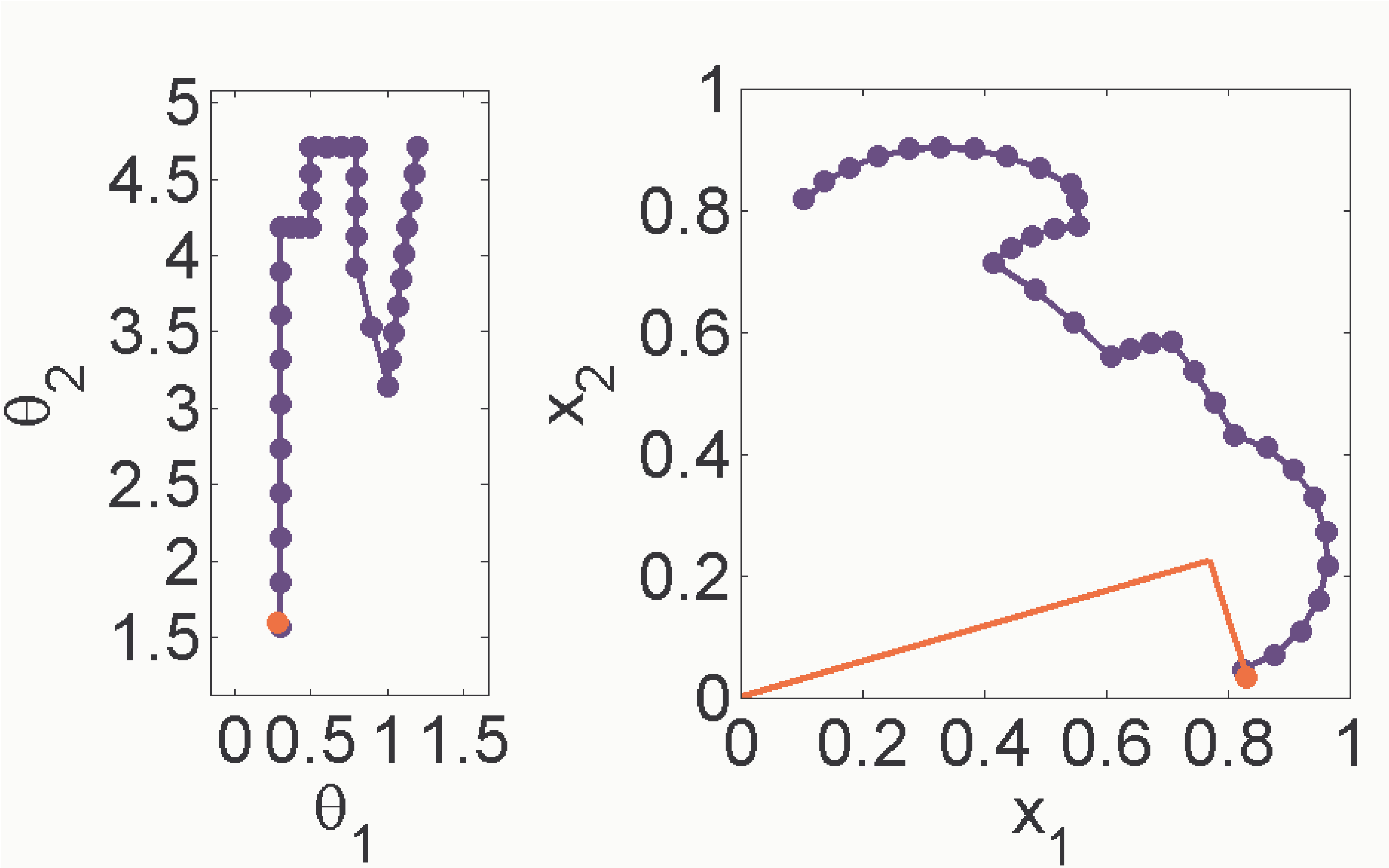
Conditional density $p(\boldsymbol{\theta}|\boldsymbol{x})$ by GTM



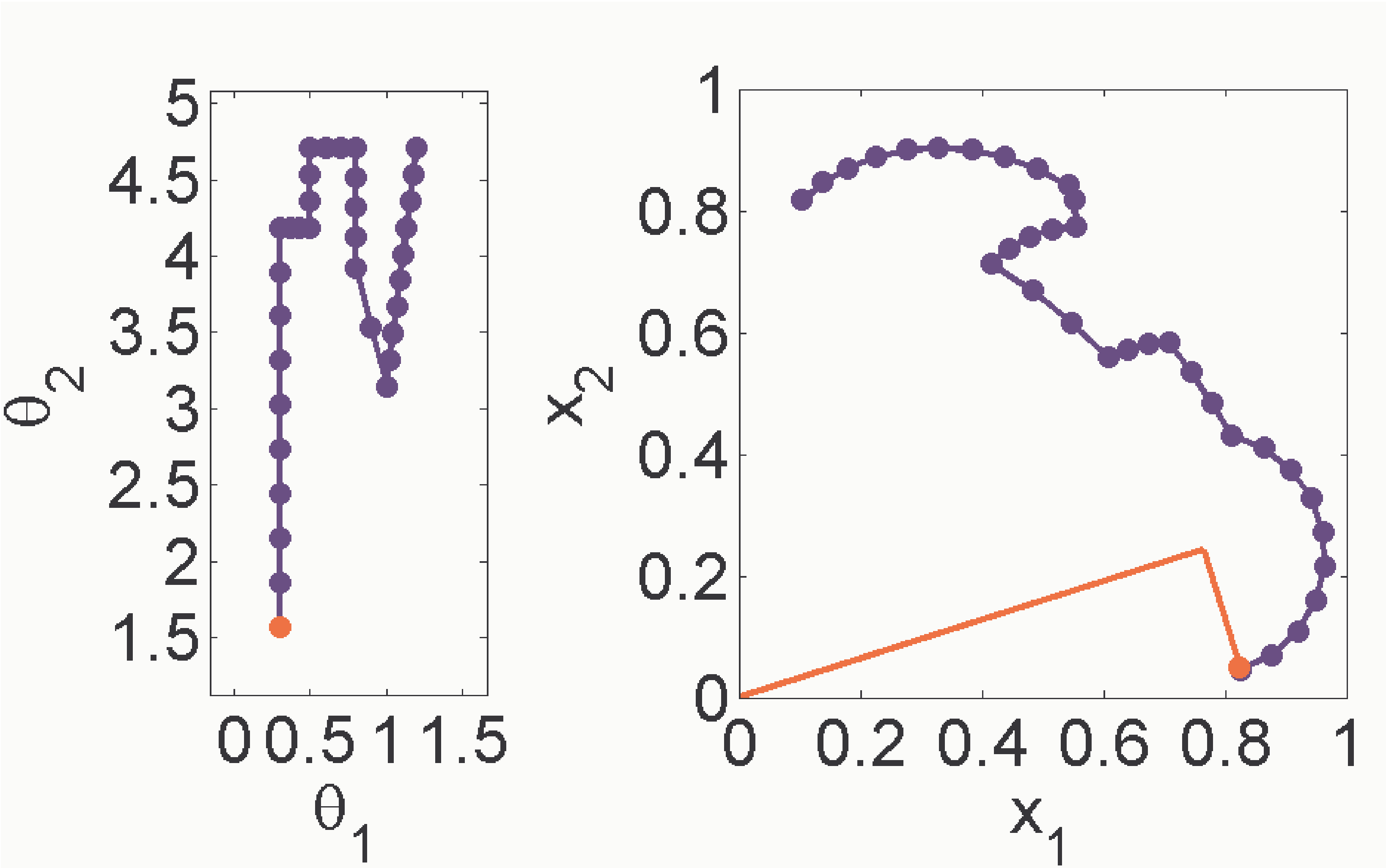
Conditional density $p(\boldsymbol{\theta}|\boldsymbol{x})$ by MDN



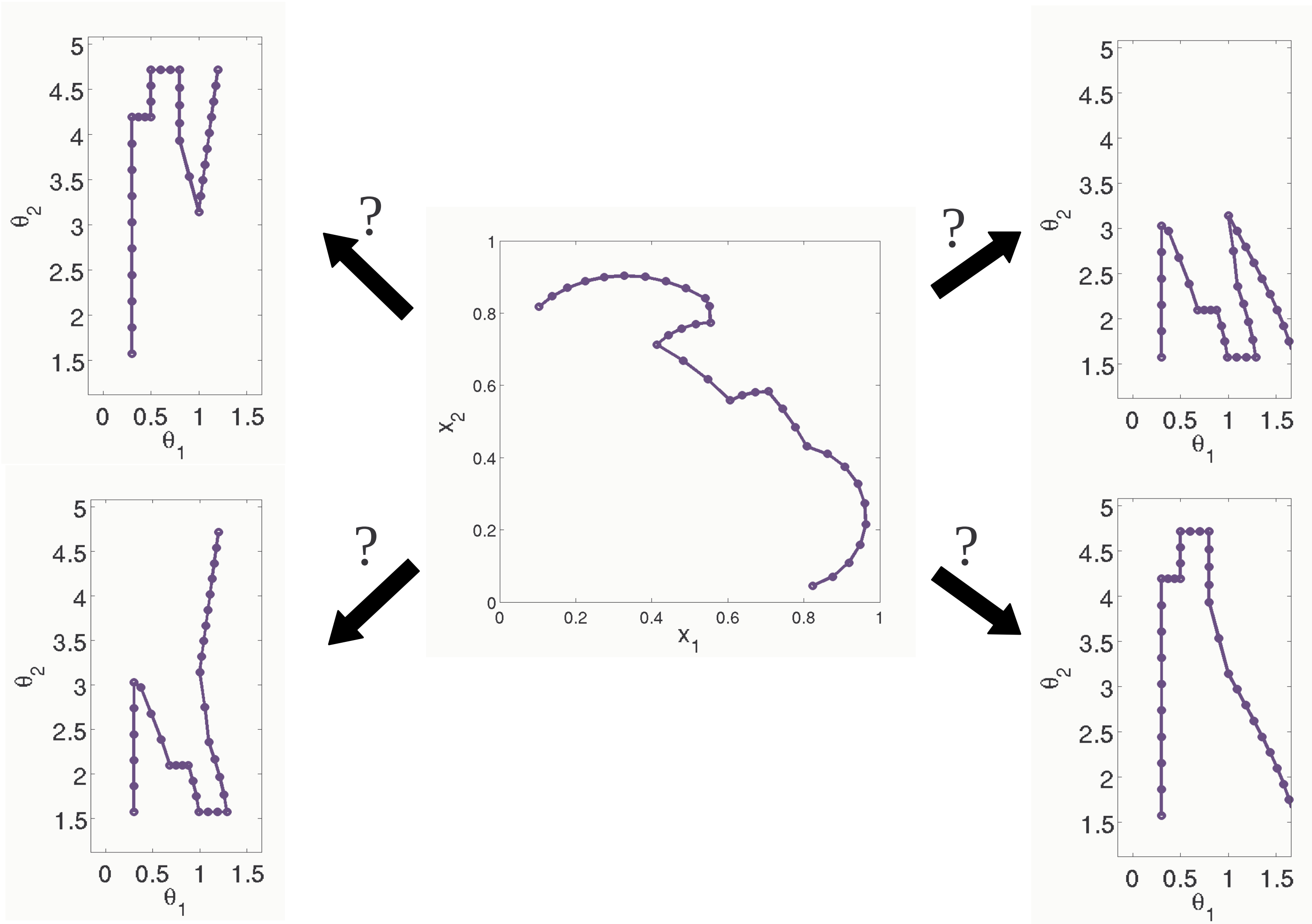
Trajectory reconstruction by GTM (modes)



Trajectory reconstruction by MDN (modes)



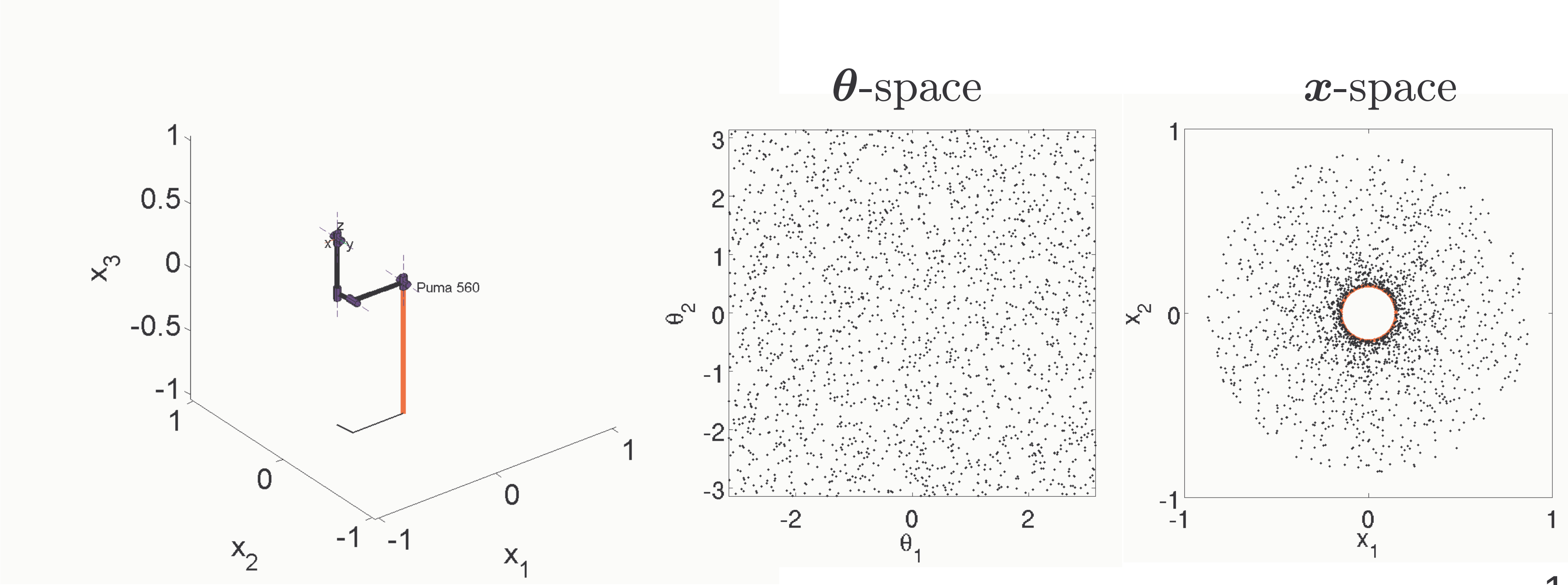
Global ambiguity



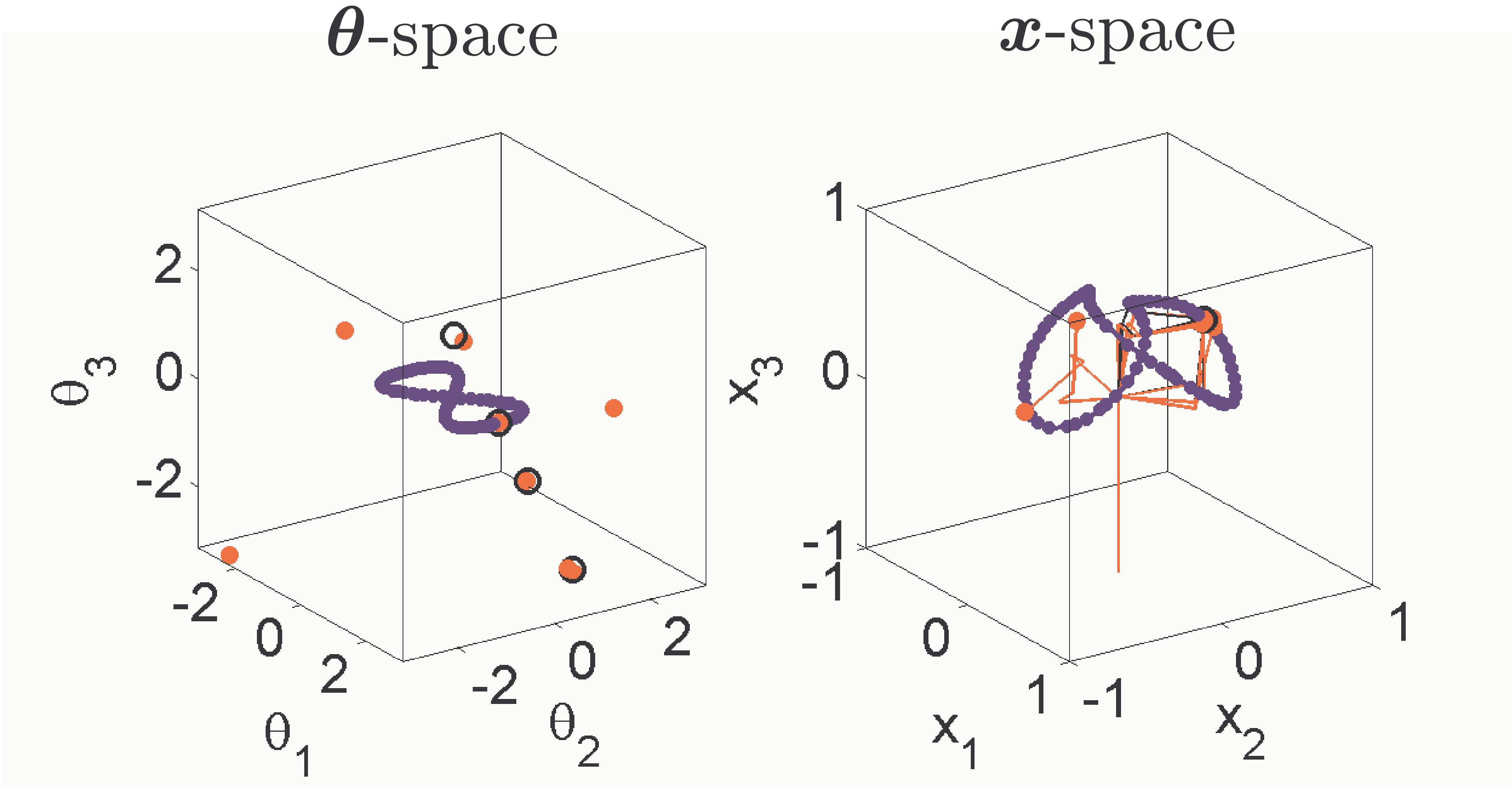
- At singular configurations, pseudoinverse doesn't know how many branches exist and local methods get stuck here
- Forbidden regions: can rule out some trajectories

Experiments: PUMA 560 robot arm

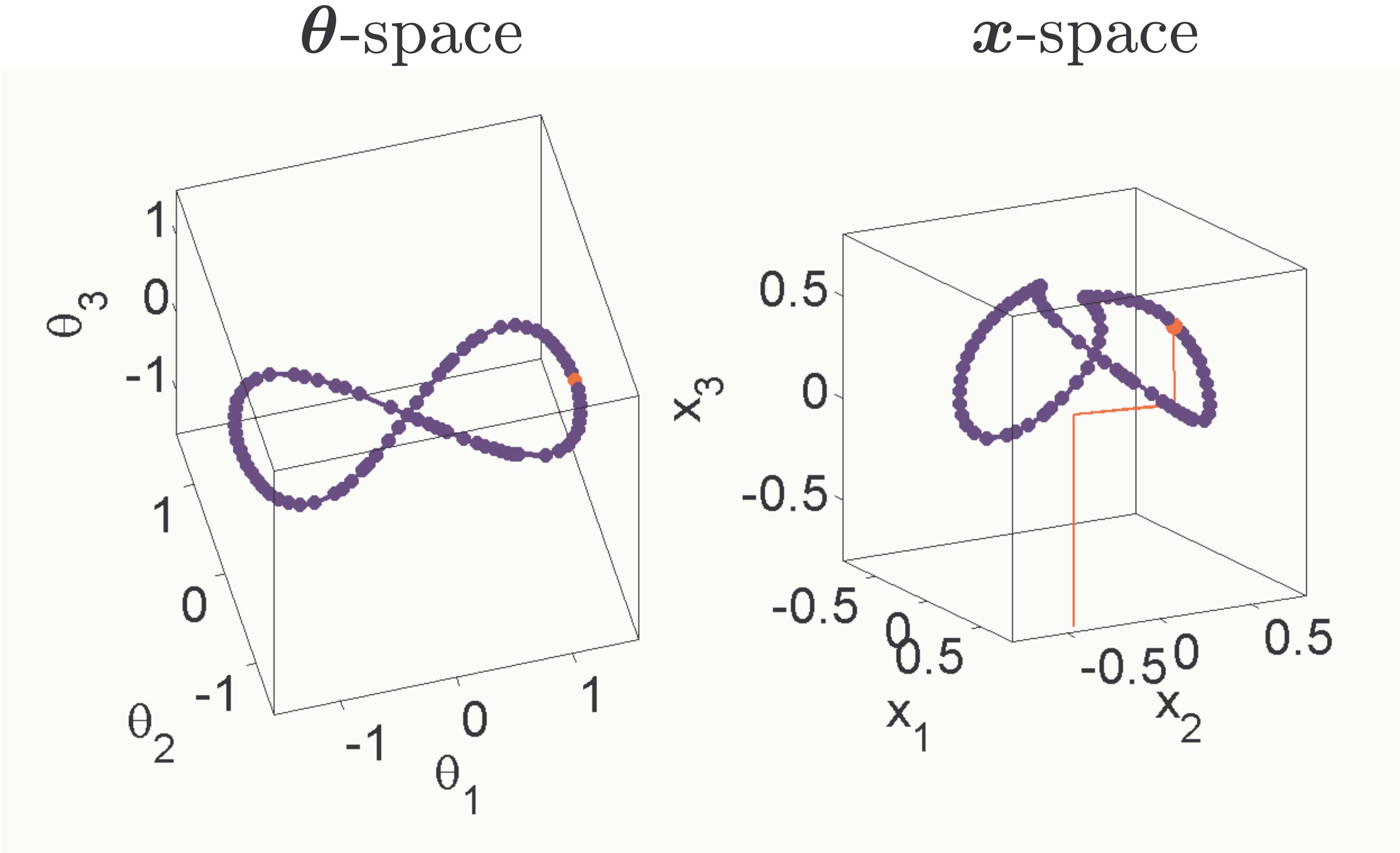
- 3D angle space (ignore orientation) and 3D workspace
- Generate a training set of 5000 pairs
- Train conditional density models
  - MDN: M=12 components, 300 hidden units
- 4 inverses for a workspace point (combinations of elbow up/down)



Conditional density $p(\boldsymbol{\theta}|\boldsymbol{x})$ by MDN

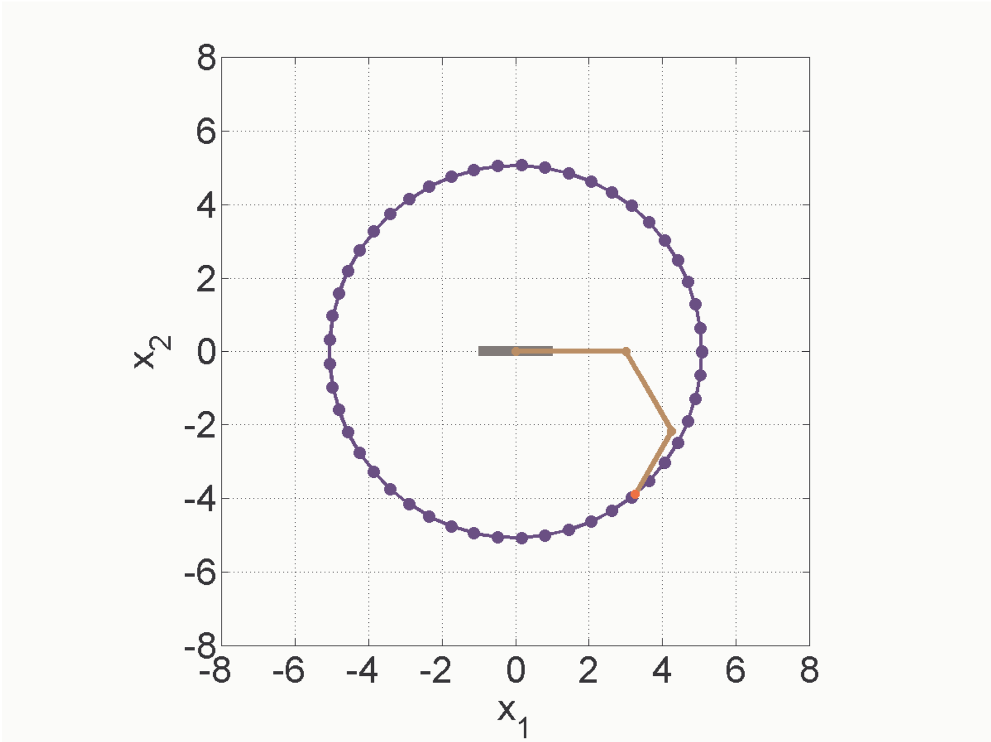


Reconstruction of figure-8 loop by MDN (modes)

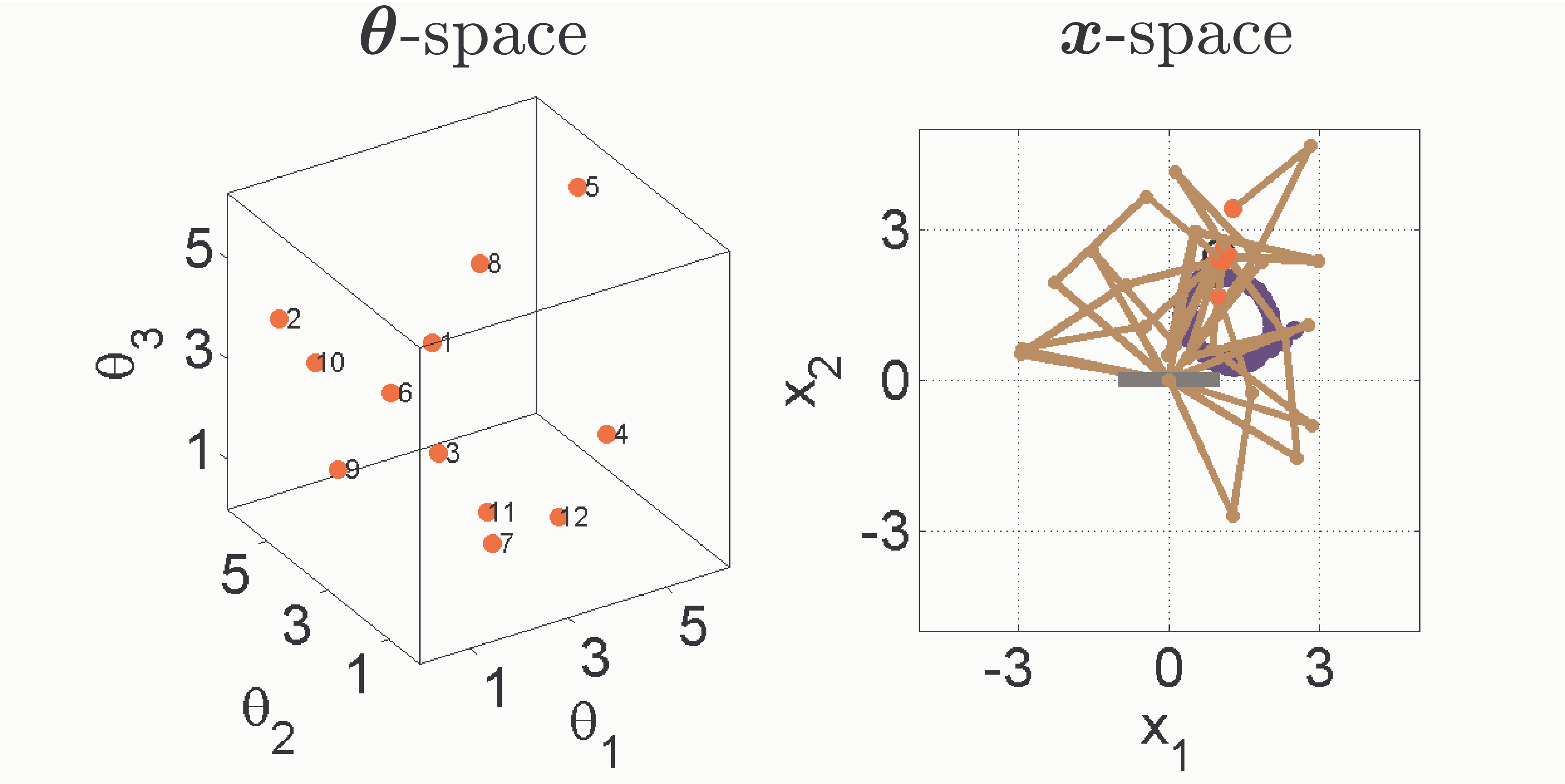


Experiments: redundant planar 3-link arm

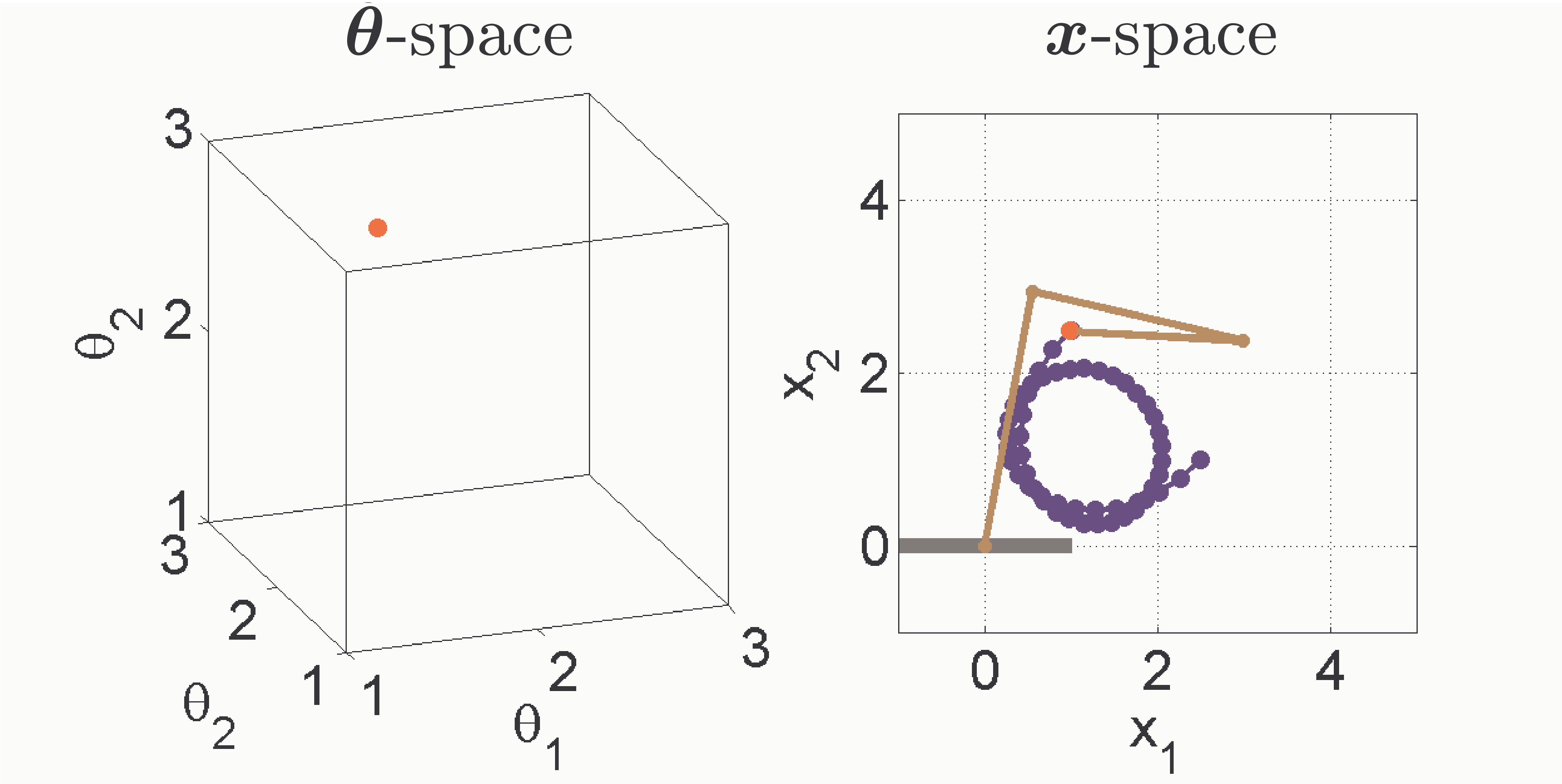
- Consider a redundant manipulator with 3D angle space and 2D workspace
- Generate a training set of 5000 pairs
- Train conditional density models
  - MDN: M=36 components, 300 hidden units



Conditional density $p(\boldsymbol{\theta}|\boldsymbol{x})$ by MDN



Reconstruction of loopy trajectory by MDN (modes)



Discussion

- Data collection: need a training set $\{(\boldsymbol{\theta}_i, \boldsymbol{x}_i)\}$
- Run time
  - Bottleneck: mode-finding (may be greatly accelerated)
  - Run time per point (Matlab implementation)

	Worst (ms)	Average (ms)	Best (ms)
Our method	50	10	4
Pseudoinverse	200	30	10

Conclusions

- Propose a machine learning method for trajectory IK that:
  - Models all the branches of the inverse mapping
  - Can deal with trajectories containing singularities, where the inverse mapping changes topology (mode split/merge); and with complicated angle domains caused by mechanical constraints (no modes)
  - Obtain accurate solutions if the density model is accurate
- The method
  - Learns a conditional density that implicitly represents all branches of the inverse mapping given a training set
  - Obtains the inverse mappings by finding the modes of the conditional density using a Gaussian mean-shift algorithm
  - Finds the angle trajectory by minimising a global, trajectory-wide constraint over the entire set of modes
- Future work will apply it to other trajectory IK problems
  - Articulatory inversion in speech, articulated pose tracking in vision, animation in graphics

Work funded by NSF CAREER award IIS-0754089