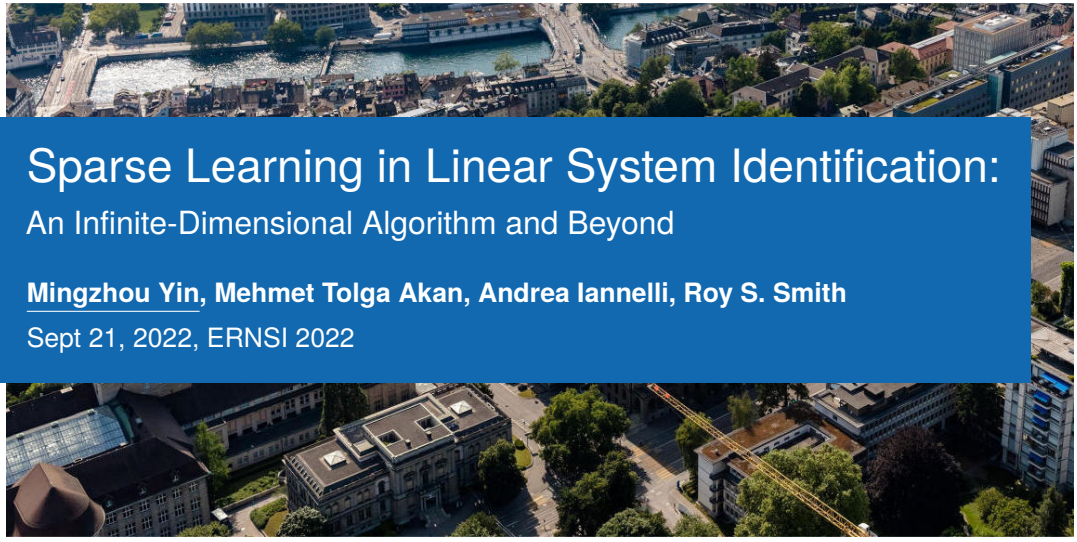
ETH zürich

Automatic Control Laboratory



Sparse Learning in Linear System Identification:

An Infinite-Dimensional Algorithm and Beyond

Mingzhou Yin, Mehmet Tolga Akan, Andrea Iannelli, Roy S. Smith

Sept 21, 2022, ERNSI 2022

From parameter estimation to function learning

Parameter estimation (classical statistics, $n \ll N$)

- Prediction error method (maximum likelihood estimation)
- Main issue: model structure / order selection

Smoothness-promoting learning (non-parametric statistics, $n \approx N$)

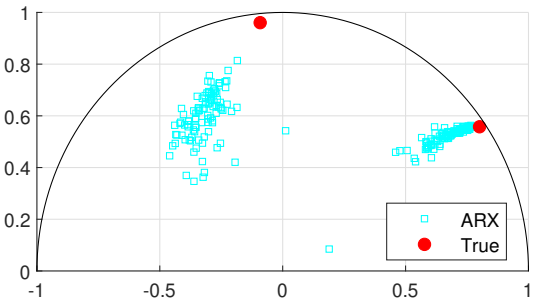
- Kernel-based identification (RKHS, Gaussian process)
- Main issue: interpretability of complexity measure

Sparsity-promoting learning (high-dimensional statistics, $n \gg N$)

- Variable selection, lasso, compressive sensing

Motivation: pole location estimation

- Key in system threoretic analysis & classical control design
- ... yet often neglected in linear system identification



4th-order discrete-time system
20 dB SNR, data length $N = 100$
ARX model with known order
100 Monte Carlo simulations

- Harder for kernel-based id: complexity controlled by induced norm of RKHS

Atomic norm regularization

- Sparse model decomposition: $G_0(q) = \sum_{k \in K} c_k A_k(q)$
  $A_k(q)$: set of model features / *'atoms'*
  $c_k \in \mathbb{C}$: *sparse* coefficients to be identified

- Assuming low-order stable systems, select first-order stable 'atoms'

$$A_k(q) = \frac{1 - |k|^2}{q - k}, \quad K = \left\{ k = \alpha \cdot \mathrm{e}^{j\beta} \,|\, \alpha \in [0, 1), \beta \in [0, 2\pi) \right\}$$

- ... pole location estimated *simultaneously*: $S = \{k \,|\, |c_k| > 0\}$

- **Approach:** l_1-norm regularization

Current gaps

- Infinitely many pole locations (K is an infinite set)
  $\rightarrow$ discretization leads to error
- l_1-norm regularization is prone to large bias
  $\rightarrow$ hard to obtain good bias-variance trade-off
- Variable 'screening' rather than variable selection
  $\rightarrow$ lots of false positives in pole location estimation

Current gaps

- Infinitely many pole locations (K is an infinite set)
  $\rightarrow$ discretization leads to error
- l_1-norm regularization is prone to large bias
  $\rightarrow$ hard to obtain good bias-variance trade-off
- Variable 'screening' rather than variable selection
  $\rightarrow$ lots of false positives in pole location estimation

This work[1]

- **Infinite-dimensional algorithm**
- **Adaptive reweighting**
- **Stability selection**

[1] Yin M., Akan M.T., Iannelli A., Smith R.S. (2022). Infinite-Dimensional Sparse Learning in Linear System Identification. Accepted for presentation at IEEE Conference on Decision and Control. arXiv:2203.14731.

Atomic norm regularization in linear SysID

Problem: Identify discrete-time linear system $y(t) = G_0(q)u(t) + v(t)$ & its pole locations from i/o data sequence

$$\mathbf{u} = [u(1)\ u(2)\ \dots\ u(N)]^\top, \quad \mathbf{y} = [y(1)\ y(2)\ \dots\ y(N)]^\top$$

Approach: Consider the first-order stable atomic decomposition, coe's c_k is identified by solving complex-valued lasso problem

$$\underset{\{c_k\}_{k\in K}}{\text{minimize}} \quad \left\| \mathbf{y} - \sum_{k\in K} c_k\, \phi_k \right\|_2^2 + \lambda \sum_{k\in K} |c_k|$$

ϕ_k: response of $A_k(q)$ under input $\mathbf{u}$

$\sum_{k\in K} |c_k|$: *atomic norm* of identified model w.r.t. atoms $A_k(q)$

Real-valued formulation

- For real-valued systems, poles are in conjugate pairs
- ... only need to consider the upper half of the unit disk

$$\hat{K} = \left\{ k = \alpha \cdot \mathrm{e}^{j\beta} \,|\, \alpha \in [0, 1), \beta \in [0, \pi] \right\}$$

- **Equivalent real-valued problem:**

$$\underset{\{\gamma_k\}_{k \in \hat{K}}}{\text{minimize}} \left\| \mathbf{y} - \sum_{k \in \hat{K}} \zeta_k \gamma_k \right\|_2^2 + 2\lambda \sum_{k \in \hat{K}} \|\gamma_k\|_2 \qquad (\star)$$

$$\gamma_k = \begin{bmatrix} \Re(c_k) & \Im(c_k) \end{bmatrix}^\top, \quad \zeta_k = \begin{bmatrix} 2\Re(\phi_k) & -2\Im(\phi_k) \end{bmatrix}$$

$\sim$ a standard group lasso problem

Solution: identified TF:

$$\hat{G}(q) = \sum_{k \in \hat{K}} [\,1 \;\; j\,]\, \gamma_k^\star A_k(q) + [\,1 \;\; -j\,]\, \gamma_k^\star A_{\bar{k}}(q)$$

estimated pole locations

$$\hat{S} = \{k \mid \|\gamma_k^\star\|_2 > 0\} \cup \left\{\bar{k} \mid \|\gamma_k^\star\|_2 > 0\right\}$$

But how to solve this infinite-dimensional problem?

- Finite-dimensional approximation (error $\propto 1/\sqrt{n(\hat{K}_d)}$)
- Feature generation algorithm (*this work*)

Observation from the optimality conditions

- The optimality conditions of $(\star)$ are

$$\begin{cases} \left\| \zeta_k^\top R \right\|_2 \leq \lambda, & \text{if } \|\gamma_k^\star\|_2 = 0 \\ \zeta_k^\top R + \lambda \gamma_k^\star / \|\gamma_k^\star\|_2 = 0, & \text{if } \|\gamma_k^\star\|_2 > 0 \end{cases}, \quad \overbrace{R = \mathbf{y} - \sum_{k \in \hat{K}} \zeta_k \gamma_k^\star}^{\text{output residuals}}$$

- For a finite-dimensional solution w.r.t. $\hat{K}_d = \{k_1, k_2, \ldots, k_p\}$, if a new atom is added $\hat{K}_d^+ := \hat{K}_d \cup \{k_{p+1}\}$, the trivial solution

$$\gamma_i^\star(\hat{K}_d^+) = \begin{cases} \gamma_i^\star(\hat{K}_d), & i = 1, \ldots, p, \\ \mathbf{0}, & i = p+1, \end{cases}$$

holds iff $\left\| \zeta_{k_{p+1}}^\top R(\hat{K}_d) \right\|_2 \leq \lambda$.

The infinite-dimensional algorithm

- k_{p+1} is only a meaningful atom when $\left\|\zeta_{k_{p+1}}^{\top} R(\hat{K}_d)\right\|_2 > \lambda$
- **Greedy algorithm:** Add new atom k_{p+1} that maximizes $\left\|\zeta_{k_{p+1}}^{\top} R(\hat{K}_d)\right\|_2$

The infinite-dimensional algorithm

- k_{p+1} is only a meaningful atom when $\left\|\zeta_{k_{p+1}}^\top R(\hat{K}_d)\right\|_2 > \lambda$
- **Greedy algorithm:** Add new atom k_{p+1} that maximizes $\left\|\zeta_{k_{p+1}}^\top R(\hat{K}_d)\right\|_2$

Input: $(\mathbf{u}, \mathbf{y})$, $\epsilon > 0$, $l_{\max}$
Initialize $\hat{K}_d^0$ and solve $(\star)$ for $\gamma^\star(\hat{K}_d^0)$
for $l = 1, \ldots, l_{\max}$ **do**
  $k^+ \leftarrow \underset{k \in \hat{K}}{\text{argmax}} \left\|\zeta_k^\top R(\hat{K}_d^{l-1})\right\|_2$ $(\triangle)$
  if $\left\|\zeta_{k^+}^\top R(\hat{K}_d^{l-1})\right\|_2 \geq \lambda + \epsilon$ **then**
    $\hat{K}_d^l \leftarrow \hat{K}_d^{l-1} \cup \{k^+\}$
    Solve $(\star)$ for $\gamma^\star(\hat{K}_d^l)$
  else
    Break
  end if
end for
Output: $\hat{K}_d^l$, $\gamma^\star(\hat{K}_d^l)$

The infinite-dimensional algorithm

- k_{p+1} is only a meaningful atom when $\left\|\zeta_{k_{p+1}}^\top R(\hat{K}_d)\right\|_2 > \lambda$
- **Greedy algorithm:** Add new atom k_{p+1} that maximizes $\left\|\zeta_{k_{p+1}}^\top R(\hat{K}_d)\right\|_2$

Proposition:

- Optimality conditions satisfied with ϵ-tolerance
- Objective decreases every iteration even if ($\triangle$) not solved exactly

Input: $(\mathbf{u}, \mathbf{y})$, $\epsilon > 0$, $l_{\max}$
Initialize $\hat{K}_d^0$ and solve ($\star$) for $\gamma^\star(\hat{K}_d^0)$
for $l = 1, \ldots, l_{\max}$ **do**
  $k^+ \leftarrow \underset{k \in \hat{K}}{\text{argmax}} \left\|\zeta_k^\top R(\hat{K}_d^{l-1})\right\|_2$ ($\triangle$)
  if $\left\|\zeta_{k^+}^\top R(\hat{K}_d^{l-1})\right\|_2 \geq \lambda + \epsilon$ **then**
    $\hat{K}_d^l \leftarrow \hat{K}_d^{l-1} \cup \{k^+\}$
    Solve ($\star$) for $\gamma^\star(\hat{K}_d^l)$
  else
    Break
  end if
end for
Output: $\hat{K}_d^l$, $\gamma^\star(\hat{K}_d^l)$

Numerical example

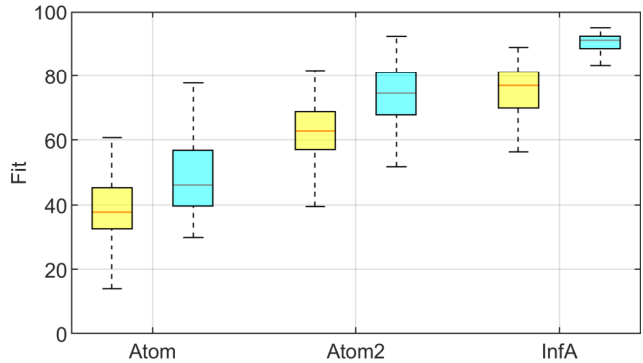
Atom: discretized solution with 50 poles

Atom2: discretized solution with 500 poles

InfA: inf-dim solution starting from 50 poles

Yellow: 20 dB SNR

Cyan: 40 dB SNR



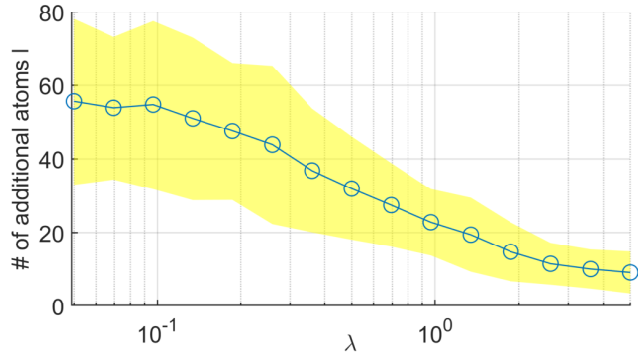
4th-order system, data length $N = 100$, 100 simulations, λ selected by cross-validation

Numerical example

Atom: discretized solution with 50 poles

Atom2: discretized solution with 500 poles

InfA: inf-dim solution starting from 50 poles



4th-order system, data length $N = 100$, 100 simulations, λ selected by cross-validation

Numerical example

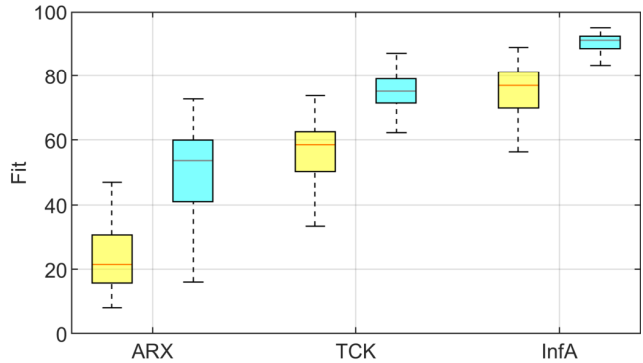
ARX: ARX model with known order

TCK: Kernel-based id with TC kernel

InfA: inf-dim solution starting from 50 poles

Yellow: 20 dB SNR

Cyan: 40 dB SNR



4th-order system, data length $N = 100$, 100 simulations, λ selected by cross-validation

After solving the inf-dim group lasso...

- Ideally we want to regularize the number of poles
- Convex relaxation: $\sum_{k\in K} |c_k| \;\rightarrow$ more penalty for large coefficients $\rightarrow$ large bias

- Iterative reweighting to regularize less for large coe's: **adaptive group lasso**

$$\underset{\gamma}{\text{minimize}} \left\| \mathbf{y} - \sum_{k\in \hat{K}_d^l} \zeta_k \gamma_k \right\|_2^2 + 2\lambda \sum_{k\in \hat{K}_d^l} \frac{\|\gamma_k\|_2}{\left\|\gamma_k^{\star,-}\right\|_2 + \epsilon'} \tag{1}$$

$\gamma_k^{\star,-}$: optimal solution from previous iteration

Numerical example

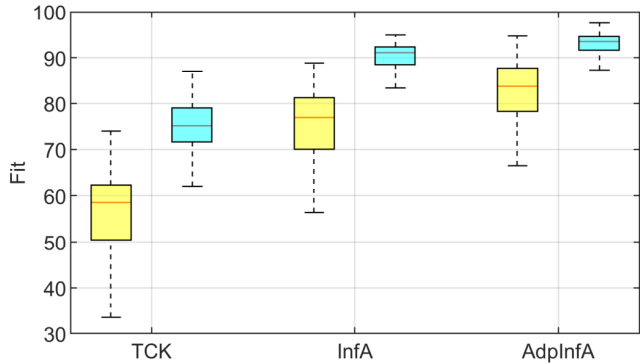
TCK: Kernel-based id with TC kernel

InfA: inf-dim solution starting from 50 poles

AdpInfA: adaptive reweighting with 2 iterations

Yellow: 20 dB SNR

Cyan: 40 dB SNR



4th-order system, data length $N = 100$, 100 simulations, λ selected by cross-validation

Numerical example

InfA: inf-dim solution starting from 50 poles

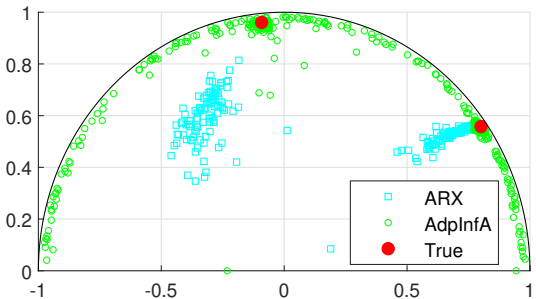
AdpInfA: adaptive reweighting with 2 iterations

	InfA	AdpInfA
20 dB SNR		
Bias^2 $[\times 10^{-2}]$	2.63	0.91
Var $[\times 10^{-2}]$	3.80	2.70
MSE $[\times 10^{-2}]$	6.44	3.60
40 dB SNR		
Bias^2 $[\times 10^{-2}]$	0.43	0.07
Var $[\times 10^{-2}]$	0.76	0.52
MSE $[\times 10^{-2}]$	1.18	0.59

4th-order system, data length $N = 100$, 100 simulations, λ selected by cross-validation

Back to pole location estimation

- *AdpInfA* looking good for model fitting, however...



- Lots of false positives!
- Lasso only guarantees non-active being ‘screened out’ with high probability
- **variable screening** instead of **variable selection** (‘p-value lottery’)

Stability selection

- Subsampling to increase 'stability' of solution
- **Complementary pairs stability selection** (CPSS)

  Input: $(\mathbf{u}, \mathbf{y})$, $\tau \in (0.5, 1]$, n_s
  for $i = 1, \ldots, n_s$ **do**
  Generate complementary pairs of random subsamples

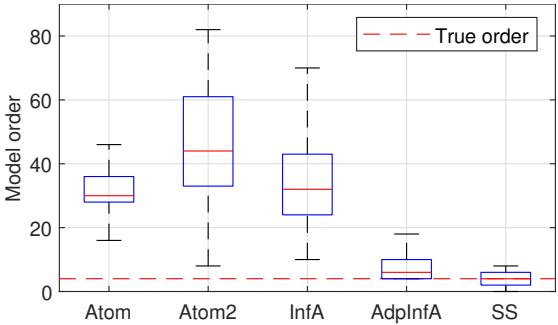
  $$B_i \subset \{1, 2, \ldots, N\}, \ \bar{B}_i \leftarrow \{1, 2, \ldots, N\} \setminus B_i$$

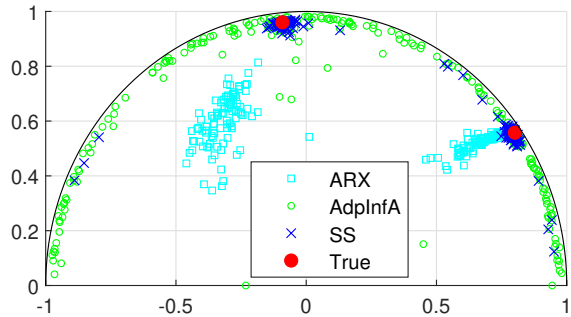
  Find active set of poles $\hat{S}_{B_i}$, $\hat{S}_{\bar{B}_i}$ by applying *AdpInfA* on subsamples
  end for
  $\hat{S} \leftarrow \left\{ k \,\middle|\, \frac{1}{2n_s} \sum_{i=1}^{n_s} \left(\mathbb{1}_{\hat{S}_{B_i}}(k) + \mathbb{1}_{\hat{S}_{\bar{B}_i}}(k) \right) \geq \tau \right\}$, $\mathbb{1}$: indicator function.
  Output: $\hat{S}$

- Control false positives when $\tau > 0.5$

Numerical example

SS: CPSS with $n_s = 50$ subsamples, $\tau = 0.9$





4th-order system, data length $N = 100$, 20 dB SNR, 100 simulations, fixed λ choice

ETH zürich

An infinite-dimensional atomic norm regularization algorithm

- Avoid discretization error by using a greedy algorithm to generate new candidate poles
- Better model fit by debiasing estimates with adaptive reweighting
- Accurate pole location estimation with stability selection

- To improve: computation complexity

AUTOMATIC CONTROL LABORATORY

Mingzhou Yin
myin@ethz.ch