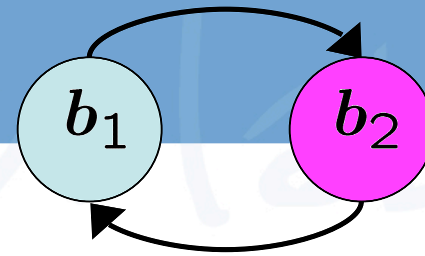
Applications to Hybrid System Identification

René Vidal
Center for Imaging Science
Institute for Computational Medicine
Johns Hopkins University

THE DEPARTMENT OF BIOMEDICAL ENGINEERING
The Whitaker Institute at Johns Hopkins

Center for IMAGING SCIENCE

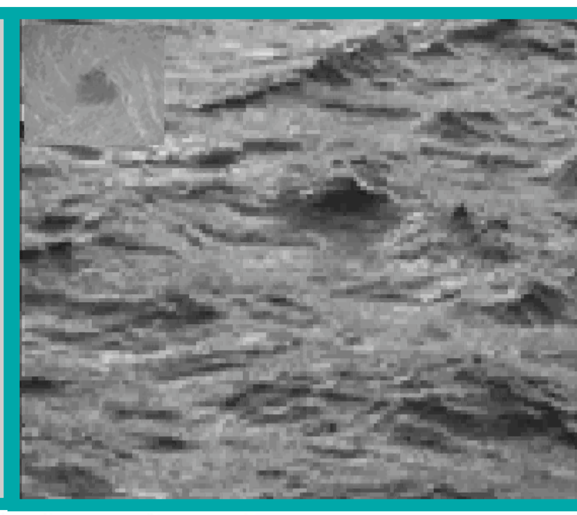
What are hybrid systems?



- Hybrid systems
  - Dynamical models with interacting discrete and continuous behavior
- Previous work
  - Modeling, analysis, stability, observability
  - Verification and control: reachability analysis, safety
- In applications one also needs to worry about identification

Video segmentation

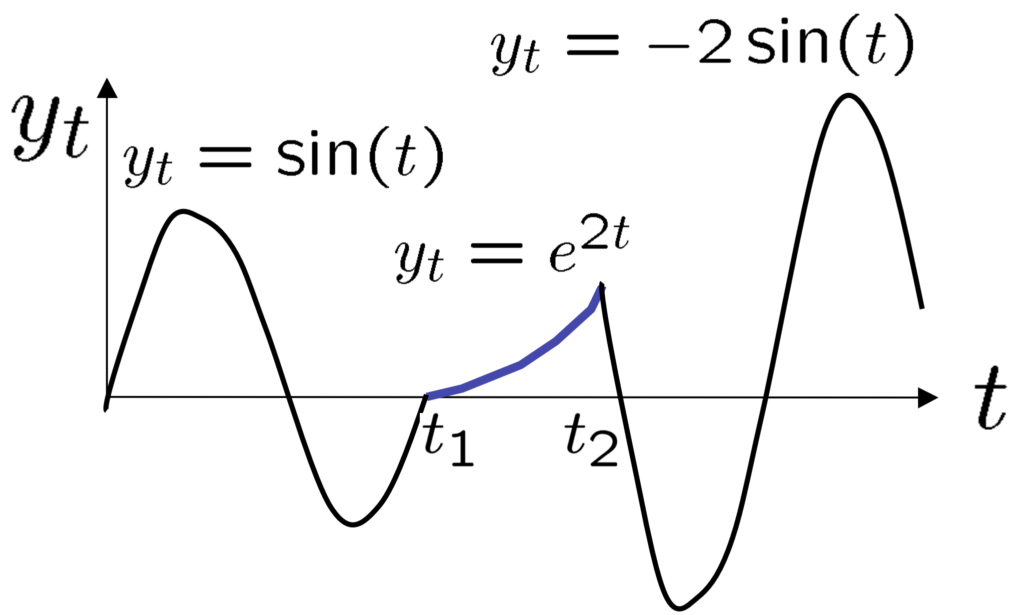
Dynamic textures



Gait recognition

Identification of hybrid systems

- Given input/output data, identify
  - Number of discrete states
  - Model parameters of linear systems
  - Hybrid state (continuous & discrete)
  - Switching parameters (partition of state space)



Main challenges

- Challenging “chicken-and-egg” problem
  - Given switching times, estimate model parameters
  - Given the model parameters, estimate hybrid state
  - Given all above, estimate switching parameters

- Possible solution: iterate
  - Very sensitive to initialization
  - Needs a minimum dwell time
  - Does not use all data

Prior work on hybrid system identification

- Mixed-integer programming: (Bemporad et al. ‘01)
- Clustering approach: k-means clustering + regression + classification + iterative refinement: (Ferrari-Trecate et al. ‘03)
- Greedy/iterative approach: (Bemporad et al. ‘03)
- Bayesian approach: maximum likelihood via expectation maximization algorithm (Juloski et al. ‘05)
- Algebraic approach: (Vidal et al. ‘03 ‘04 ‘05)

Algebraic approach to hybrid system ID

- Key idea
  - Number of models = degree of a polynomial
  - Model parameters = roots (factors) of a polynomial
- Batch methods
  - CDC'03: known #models of equal and known orders
  - HSCC'05: unknown #models of unknown and possibly different orders
- Recursive methods
  - CDC'04: known #models of equal and known orders
  - CDC'05: unknown #models of unknown and possibly different orders

Problem formulation

- Switch ARX system (SARX)

$$y_t = \sum_{j=1}^{n_a(\lambda_t)} a_j(\lambda_t) y_{t-j} + \sum_{j=1}^{n_c(\lambda_t)} c_j(\lambda_t) u_{t-j} \ \ (+ w_t)$$

Given input y_t and output u_t over an interval

$$t = 0, 1, 2, \ldots, T-1,$$

determine:

1. the number n of the systems;
2. the order $\{n_a(i), n_c(i)\}_{i=1}^{n}$ of each system;
3. the system parameters $\{a_j(i), c_j(i)\}_{i=1}^{n}$;
4. the switching λ_t between systems.

A single ARX system: known orders

Knowing Systems Orders $D = n_a + n_c + 1$

Regressors

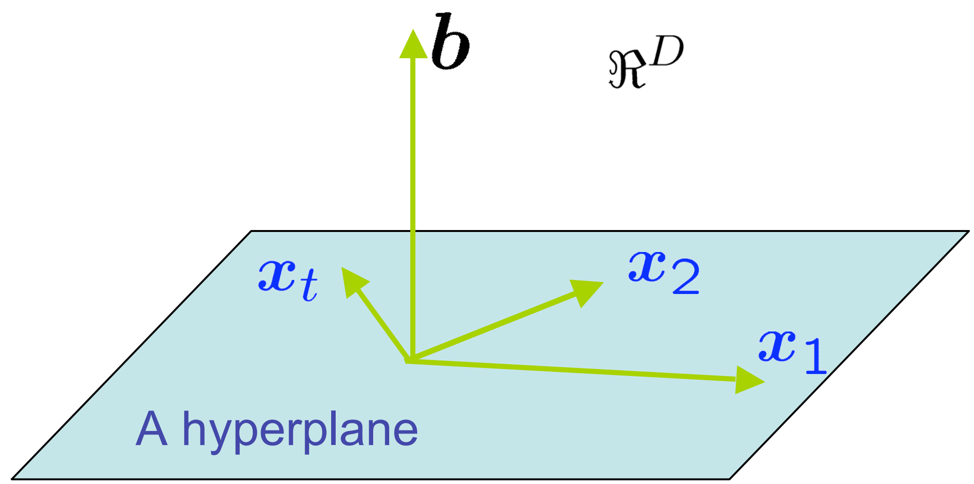
$$\boldsymbol{x}_t \doteq \left[y_t, y_{t-1}, \ldots, y_{t-n_a}, u_{t-1}, u_{t-2}, \ldots, u_{t-n_c}\right]^T \quad \in \Re^D.$$

Parameter vector

$$\boldsymbol{b} \doteq \left[1, -a_1, -a_2, \ldots, -a_{n_a}, -c_1, -c_2, \ldots, -c_{n_c}\right]^T \quad \in \Re^D.$$

Data matrix $L(n_a, n_c)$

$$\overbrace{\begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_t^T \\ \vdots \end{bmatrix}} \boldsymbol{b} = 0$$



A single ARX system: unknown orders

Not Knowing Systems Orders $\bar{n}_a \geq n_a, \bar{n}_c \geq n_c, D = \bar{n}_a + \bar{n}_c + 1$

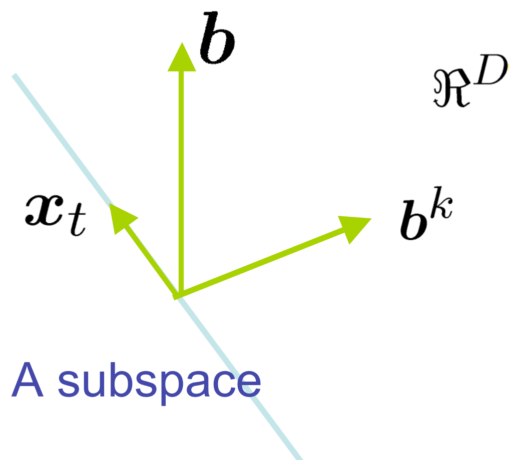
Regressors $\boldsymbol{x}_t \doteq \left[y_t, y_{t-1}, \ldots, y_{t-\bar{n}_a}, u_{t-1}, u_{t-2}, \ldots, u_{t-\bar{n}_c}\right]^T.$

Parameter vectors

$$\begin{aligned}
\boldsymbol{b} &= \left[1, -a_1, \ldots, -a_{n_a}, \mathbf{0}_{1\times(\bar{n}_a-n_a)}, -c_1, \ldots, -c_{n_c}, \mathbf{0}_{1\times(\bar{n}_c-n_c)}\right]^T, \\
\boldsymbol{b}^1 &= \left[\mathbf{0}_1, 1, -a_1, \ldots, -a_{n_a}, \mathbf{0}_{\bar{n}_a-n_a-1}, \mathbf{0}_1, -c_1, \ldots, -c_{n_c}, \mathbf{0}_{\bar{n}_c-n_c-1}\right]^T, \\
\boldsymbol{b}^2 &= \left[\mathbf{0}_2, 1, -a_1, \ldots, -a_{n_a}, \mathbf{0}_{\bar{n}_a-n_a-2}, \mathbf{0}_2, -c_1, \ldots, -c_{n_c}, \mathbf{0}_{\bar{n}_c-n_c-2}\right]^T, \\
&\vdots \qquad\qquad\qquad \vdots
\end{aligned}$$

Data matrix $L(\bar{n}_a, \bar{n}_c)$

$$\underbrace{\begin{bmatrix} \boldsymbol{x}_1^T \\ \boldsymbol{x}_2^T \\ \vdots \\ \boldsymbol{x}_t^T \\ \vdots \end{bmatrix}}_{L(\bar{n}_a, \bar{n}_c)} [\boldsymbol{b}, \boldsymbol{b}^1, \boldsymbol{b}^2, \ldots] = 0$$

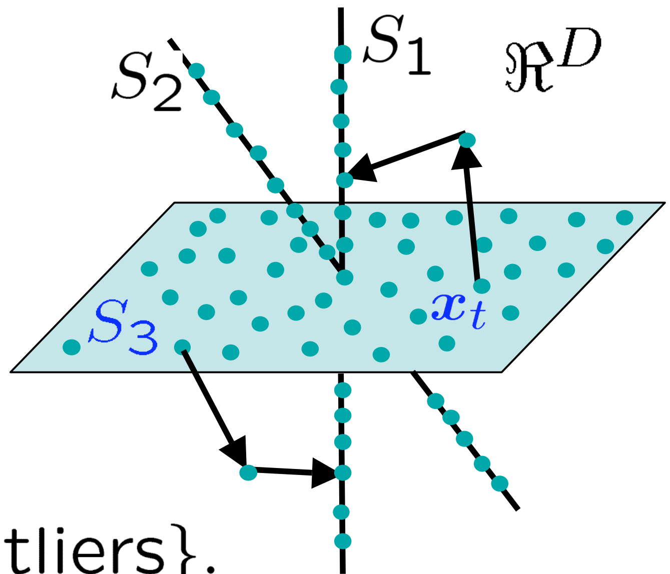


A Switched ARX system

Embedding in $\Re^D$ $\quad \bar{n}_a \geq \max_i n_a(i), \bar{n}_c \geq \max_i n_c(i), D = \bar{n}_a + \bar{n}_c + 1$

Configuration Space of Regressors:

1. Regressors of each system lie on a subspace in $\Re^D$
2. Order of each system is related to the subspace dimension
3. Switching among systems corresponds to switching among the subspaces



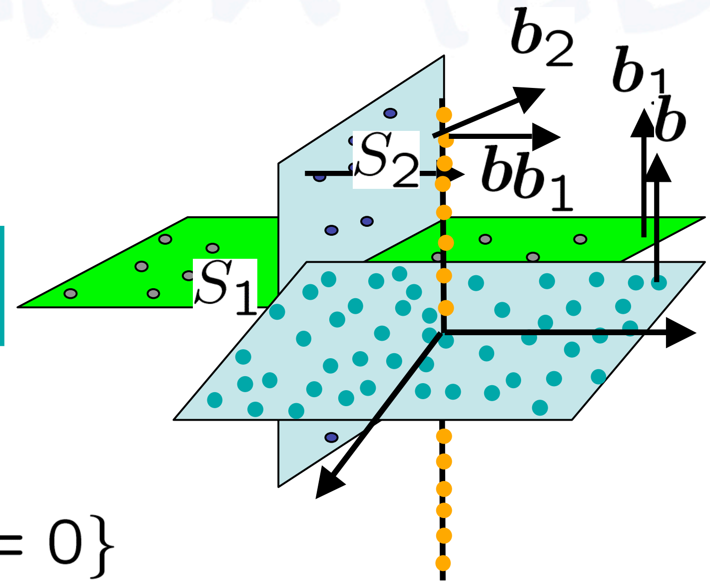
$$Z' \doteq S_1 \cup S_2 \cup \cdots \cup S_n \cup \{\text{outliers}\}.$$
$$Z'' \doteq H_1 \cup H_2 \cup \cdots \cup H_n.$$

Representing n subspaces

- Two planes

$$(b_1^T x = 0) \ \boxed{\text{or}} \ (b_2^T x = 0)$$

$$\boxed{p_2(x) = (b_1^T x)(b_2^T x) = 0}$$



- One plane and one line
  - Plane: $S_1 = \{x : b^T x = 0\}$
  - Line: $S_2 = \{x : b_1^T x = b_2^T x = 0\}$

$$S_1 \cup S_2 = \{x : (b^T x = 0) \ \boxed{\text{or}} \ (b_1^T x = b_2^T x = 0\}$$

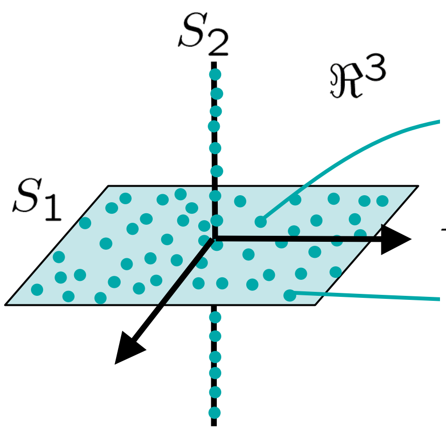
De Morgan's rule

$$S_1 \cup S_2 = \{x : \boxed{(b^T x)(b_1^T x) = 0} \ \boxed{\text{and}} \ \boxed{(b^T x)(b_2^T x) = 0}\}$$

- **A union of n subspaces can be represented with a set of homogeneous polynomials of degree n**

Polynomial fitting

Veronese Map $\nu_2(\boldsymbol{x}) \in \Re^6$



$$L_2 = \begin{bmatrix} (x_1^1)^2 & x_1^1 x_2^1 & x_1^1 x_3^1 & (x_2^1)^2 & x_2^1 x_3^1 & (x_3^1)^2 \\ (x_1^2)^2 & x_1^2 x_2^2 & x_1^2 x_3^2 & (x_2^2)^2 & x_2^2 x_3^2 & (x_3^2)^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ (x_1^N)^2 & x_1^N x_2^N & x_1^N x_3^N & (x_2^N)^2 & x_2^N x_3^N & (x_3^N)^2 \end{bmatrix}$$

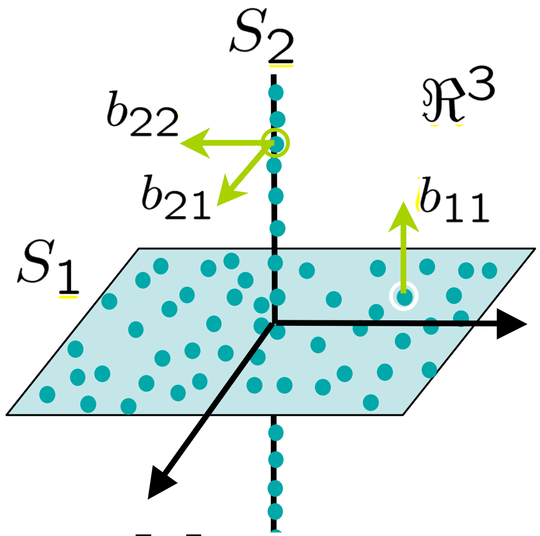
The null space of L_2 is:

$$c_1 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T \quad \rightarrow \quad p_1(\boldsymbol{x}) = (\nu_2(\boldsymbol{x}))^T c_1 = x_1 x_3$$

$$c_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T \quad \rightarrow \quad p_2(\boldsymbol{x}) = (\nu_2(\boldsymbol{x}))^T c_2 = x_2 x_3$$

Null space of L_n contains information about all the polynomials.

Polynomial differentiation



$$P(\boldsymbol{x}) = [p_1(\boldsymbol{x}), p_2(\boldsymbol{x})] = [x_1 x_3, x_2 x_3]$$

$$DP(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial p_1}{\partial x_1} & \frac{\partial p_2}{\partial x_1} \\ \frac{\partial p_1}{\partial x_2} & \frac{\partial p_2}{\partial x_2} \\ \frac{\partial p_1}{\partial x_3} & \frac{\partial p_2}{\partial x_3} \end{bmatrix} = \begin{bmatrix} x_3 & 0 \\ 0 & x_3 \\ x_1 & x_2 \end{bmatrix}$$

$$\boldsymbol{x} = \begin{bmatrix} a \\ b \\ 0 \end{bmatrix} \in S_1 \quad DP(\boldsymbol{x}) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ a & b \end{bmatrix}$$ The columns of $DP(\boldsymbol{x})$ span $S_1^\perp$.

$$\boldsymbol{y} = \begin{bmatrix} 0 \\ 0 \\ c \end{bmatrix} \in S_2 \quad DP(\boldsymbol{y}) = \begin{bmatrix} c & 0 \\ 0 & c \\ 0 & 0 \end{bmatrix}$$ The columns of $DP(\boldsymbol{y})$ span $S_2^\perp$.

The information of the mixture of subspaces can be obtained via polynomial differentiation.

Hybrid decoupling polynomial

$$\boldsymbol{x}_t \doteq \left[y_t, y_{t-1}, \ldots, y_{t-\bar{n}_a}, u_{t-1}, u_{t-2}, \ldots, u_{t-\bar{n}_c}\right]^T.$$

$$\boldsymbol{b}_i = \left[1, -a_1(i), \ldots, -a_{n_a(i)}(i), \mathbf{0}_{1\times(\bar{n}_a - n_a(i))}, -c_1(i), \ldots, -c_{n_c(i)}(i), \mathbf{0}_{1\times(\bar{n}_c - n_c(i))}\right]^T.$$

For all regressors $x \in Z' \subseteq Z''$:

$$p_n(\boldsymbol{x}) \doteq \prod_{i=1}^{n} \left(\boldsymbol{b}_i^T \boldsymbol{x}\right) = \sum c_{n_1,\ldots,n_D} x_1^{n_1} \cdots x_D^{n_D} = \boldsymbol{c}_n^T \nu_n(\boldsymbol{x}) = 0.$$

Lemma 1 (Hybrid Decoupling Polynomial) *The monomial associated with the last non-zero entry of the coefficient vector* $\boldsymbol{c}_n$ *of the hybrid decoupling polynomial* $p_n(\boldsymbol{x}) = \boldsymbol{c}_n^T \nu_n(\boldsymbol{x})$ *has the lowest degree-lexicographic order in all the polynomials of degree* n *in the vanishing ideal* $\mathfrak{a}(Z)$ *or* $\mathfrak{a}(Z')$.

Identifying the hybrid decoupling polynomial

$$p_n(\boldsymbol{x}) \doteq \prod_{i=1}^{n} \left(\boldsymbol{b}_i^T \boldsymbol{x}\right) = \sum c_{n_1,\ldots,n_D} x_1^{n_1} \cdots x_D^{n_D} = \boldsymbol{c}_n^T \nu_n(\boldsymbol{x}) = 0.$$

Theorem 1 (Identifying Hybrid Decoupling Polynomial) *Let $L_n^j \in \Re^{T \times j}$ be the first j columns of $L_n(\bar{n}_a, \bar{n}_c)$, and let*

$$m \doteq \min\left\{j : \operatorname{rank}\left(L_n^j\right) = j - 1\right\}.$$

The coefficient vector $\boldsymbol{c}_n$ of the hybrid decoupling polynomial is

$$\boldsymbol{c}_n = \left[\left(\boldsymbol{c}_n^m\right)^T, \, \boldsymbol{0}_{1 \times (M_n(D) - m)}\right]^T \in \Re^{M_n(D)},$$

where $\boldsymbol{c}_n^m \in \Re^m$ is the unique vector that satisfies

$$L_n^m \boldsymbol{c}_n^m = \boldsymbol{0} \quad \text{and} \quad \boldsymbol{c}_n^m(1) = 1.$$

Lemma 1 (Identifying the Number of ARX Systems) *The polynomial found by Theorem 1 is*

$$p_{\bar{n}}(\boldsymbol{x}) = \boldsymbol{c}_{\bar{n}}^T \nu_{\bar{n}}(\boldsymbol{x}) = \left(\boldsymbol{b}_1^T \boldsymbol{x}\right)\left(\boldsymbol{b}_2^T \boldsymbol{x}\right) \cdots \left(\boldsymbol{b}_n^T \boldsymbol{x}\right) x_1^{\bar{n}-n}.$$

Batch algorithm summary

Algorithm 1 (Identification of an SISO SARX System).

Given the input/output data $\{y_t, u_t\}$ from a sufficiently excited hybrid ARX system, and the upper bound on the number $\bar{n}$ and maximum orders $(\bar{n}_a, \bar{n}_c)$ of its constituent ARX systems:

1. **Veronese Embedding.** Construct the data matrix $L_{\bar{n}}(\bar{n}_a, \bar{n}_c)$ via the Veronese map based on the given number $\bar{n}$ of systems and the maximum orders $(\bar{n}_a, \bar{n}_c)$.

2. **Hybrid Decoupling Polynomial.** Compute the coefficients of the polynomial $p_{\bar{n}}(\boldsymbol{x}) \doteq \boldsymbol{c}_{\bar{n}}^T \nu_{\bar{n}}(\boldsymbol{x}) = \prod_{i=1}^{n} \left(\boldsymbol{b}_i^T \boldsymbol{x}\right) x_1^{\bar{n}-n} = 0$ from the data matrix $L_{\bar{n}}$ according to the previous Theorem and Corollary.

3. **Constituent System Parameters.** Retrieve the parameters $\{\boldsymbol{b}_i\}_{i=1}^{n}$ of each constituent ARX system from $p_{\bar{n}}(\boldsymbol{x})$ according to the GPCA algorithm.

4. **Key System Parameters.** The correct number of system n is the number of $\boldsymbol{b}_i \neq \boldsymbol{e}_1$; The correct orders $n_a(i), n_c(i)$ are determined from such $\boldsymbol{b}_i$ according to their definition; The discrete state is $\lambda_t = \text{argmin}_{i=1,\dots,n} \left(\boldsymbol{b}_i^T \boldsymbol{x}_t\right)^2$.

Stochastic versus deterministic case

$$y_t = \sum_{j=1}^{n_a(\lambda_t)} a_j(\lambda_t) y_{t-j} + \sum_{j=1}^{n_c(\lambda_t)} c_j(\lambda_t) u_{t-j} \ \ (+ w_t)$$

ML-Estimate: minimizing the sum of squares of errors (SSE):

$$\min_{\boldsymbol{b}_i, \lambda} \sum_t w_t^2 = \sum_t (\boldsymbol{b}_{\lambda_t}^T \boldsymbol{x}_t)^2.$$

GPCA: minimizing a weighted SSE:

$$\min_{\boldsymbol{b}_i, \lambda} \sum_t \alpha_t (\boldsymbol{b}_{\lambda_t}^T \boldsymbol{x}_t)^2 \doteq \sum_t \prod_{i \neq \lambda_t} (\boldsymbol{b}_i^T \boldsymbol{x}_t)^2 (\boldsymbol{b}_{\lambda_t}^T \boldsymbol{x}_t)^2.$$

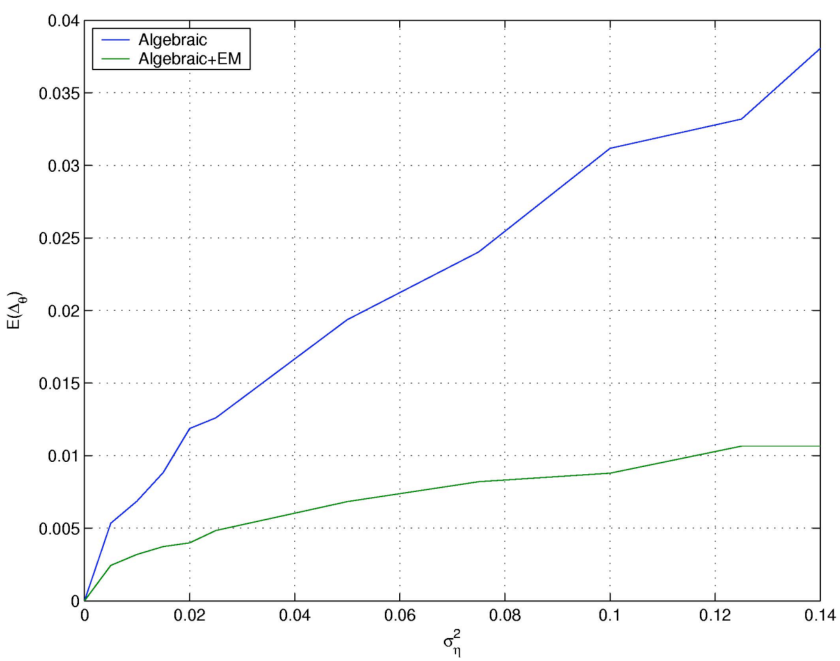
GPCA is a “relaxed” version of expectation maximization (EM) that permits a non-iterative solution.

Simulation results

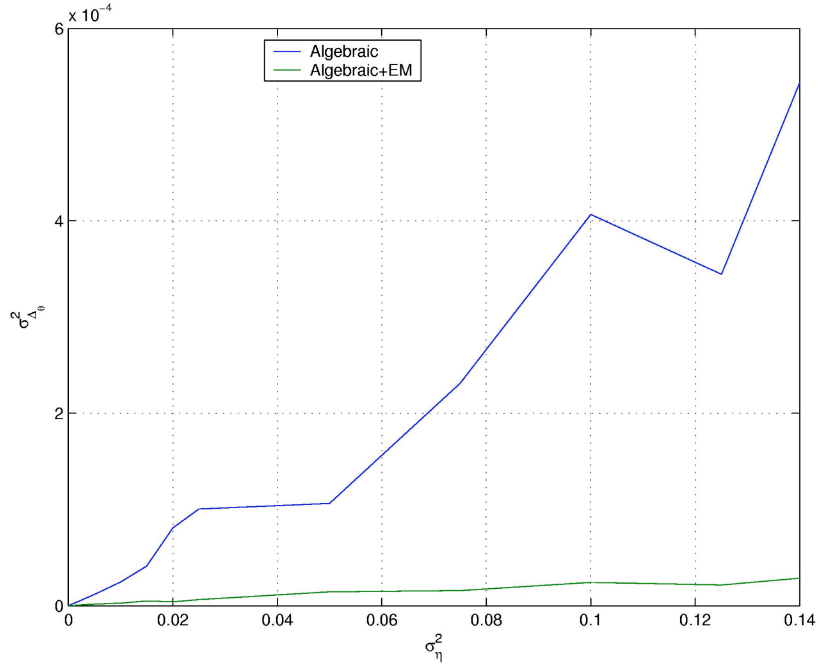
System: $$y_t = \begin{cases} 2u_{t-1} + 10 + w_t & \text{if} \quad u_{t-1} \in [-10, 0], \\ -1.5u_{t-1} + 10 + w_t & \text{if} \quad u_{t-1} \in (0, 10], \end{cases}$$

Error: $$\max_{i=1,\dots,n.} \min_{j=1,\dots,n.} \left\| \hat{\boldsymbol{b}}_i - \boldsymbol{b}_j \right\| / \|\boldsymbol{b}_j\|$$

Mean



Variance



Pick-and-place machine experiment

Four datasets of T = 60,000 measurements from a component placement process in a pick-and-place machine [Juloski:CEP05]

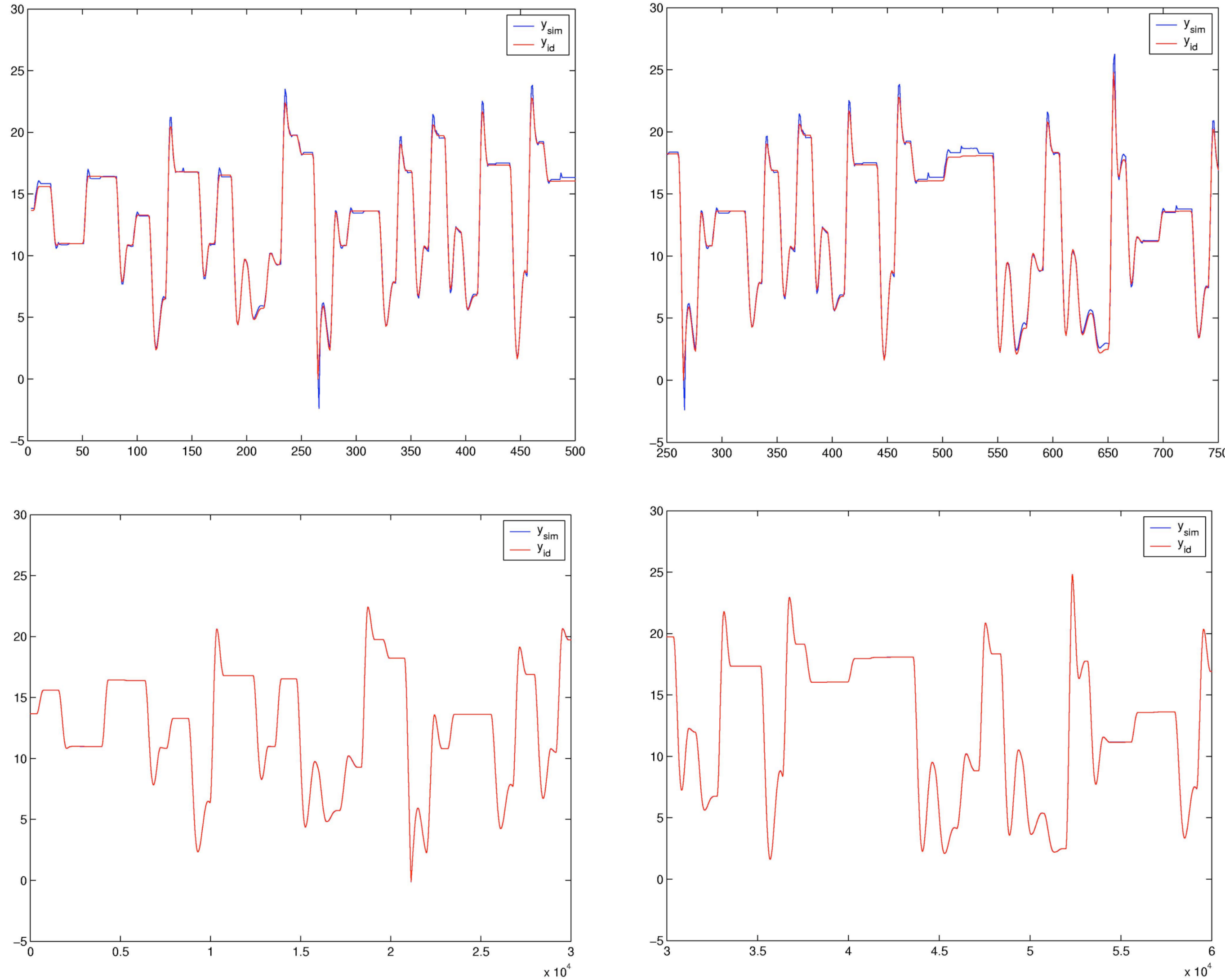
- Training and simulation errors for down-sampled datasets (1/80):

Dataset	n	n_a	n_c	SSR	SSE
1	2	2	2	0.0803	0.1195
2	2	2	2	0.4765	0.4678
3	2	2	2	0.6692	0.7368
4	2	2	2	3.1004	3.8430

- Training and simulation errors for complete datasets:

Dataset	n	n_a	n_c	SSR	SSE
1 with all points	2	2	2	$4.9696 \cdot 10^{-6}$	$5.3426 \cdot 10^{-6}$
2 with all points	2	2	2	$9.2464 \cdot 10^{-6}$	$7.9081 \cdot 10^{-6}$
3 with all points	2	2	2	$2.3010 \cdot 10^{-5}$	$2.5290 \cdot 10^{-5}$
4 with all points	2	2	2	$7.5906 \cdot 10^{-6}$	$9.6362 \cdot 10^{-6}$

Pick-and-place machine experiment



Sub-sampled
(1 every 80)

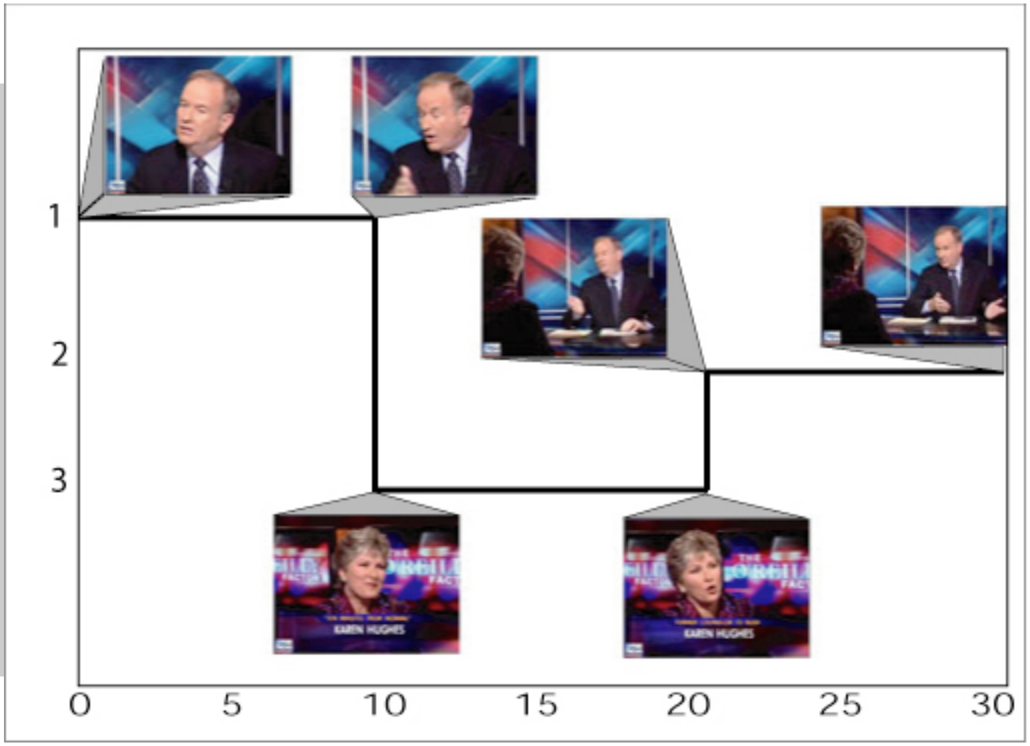
All data
(60,000)

Training

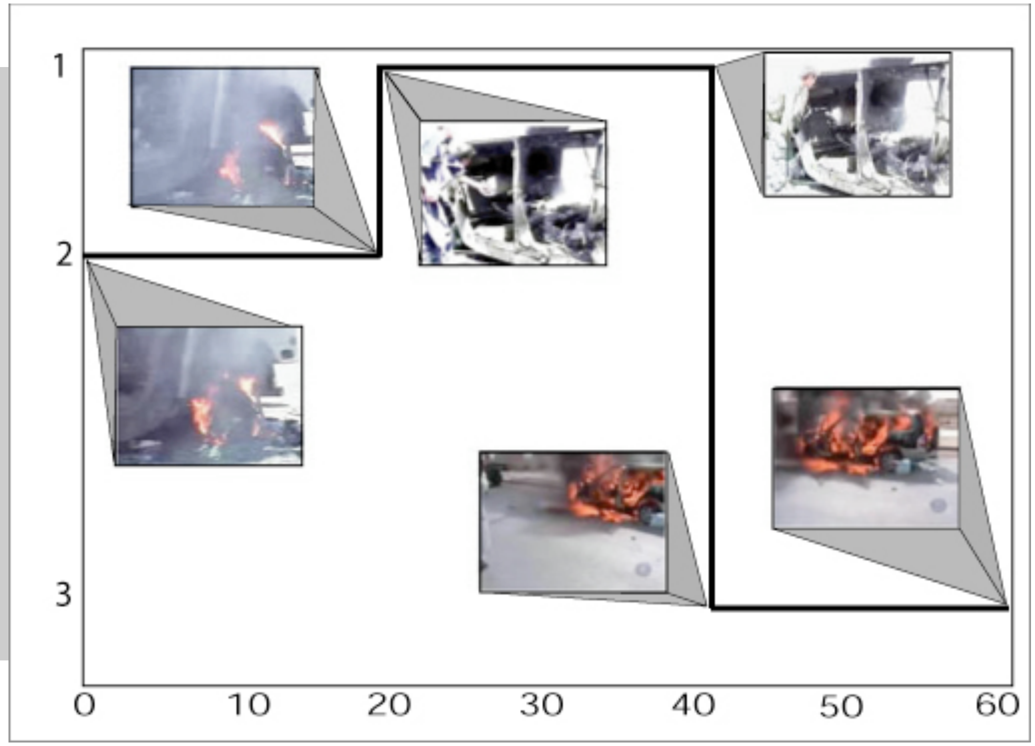
Simulation

Center for
IMAGING
SCIENCE

Application in video segmentation



Application in video segmentation



Conclusions for batch method

- Identification of SARX of unknown and possibly different dimensions
  - Decouple identification and filtering
  - Algebraic solution that can be used for initialization
    - Polynomial fitting + rank constraint
    - Polynomial differentiation
  - Does not need minimum dwell time

- Ongoing work
  - MIMO ARX models: multiple polynomials (HSCC'08)

Recursive identification algorithms

- Most existing methods are batch
  - Collect all input/output data
  - Identify model parameters using all data
- Not suitable for online/real time operation

- Contributions
  - Recursive identification algorithm for Switched ARX
    - No restriction on switching mechanism
    - Does not depend on value of the discrete state
    - Based on algebraic geometry and linear system ID
    - Key idea: identification of multiple ARX models is equivalent to identification of a single ARX model in a lifted space
  - Persistence of excitation conditions that guarantee exponential convergence of the identified parameters

Recall the notation

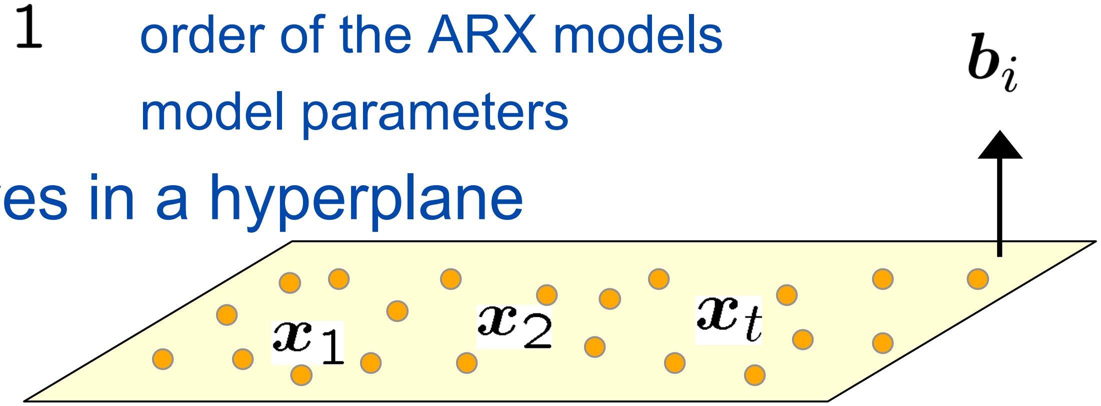
- The dynamics of each mode are in ARX form

$$y_t = \sum_{j=1}^{n_a} a_j(\lambda_t) y_{t-j} + \sum_{j=1}^{n_c} c_j(\lambda_t) u_{t-j}$$

  - u_t, y_t input/output
  - $\lambda_t \in \{1, 2, \ldots, n\}$ discrete state
  - $K = n_a + n_c + 1$ order of the ARX models
  - $a_j(\lambda_t), c_j(\lambda_t)$ model parameters

- Input/output data lives in a hyperplane

$$\boldsymbol{b}_i^T \boldsymbol{x}_t = 0$$



  - I/O data
  - Model parameters

$$\boldsymbol{x}_t = [u_{t-n_c}, \ldots, u_{t-1}, y_{t-n_a}, \ldots, y_{t-1}, -y_t]^T$$

$$\boldsymbol{b}_i = [c_{n_c}(i), \ldots, c_1(i), a_{n_a}(i), \ldots a_1(i), 1]^T$$

Recursive identification of ARX models

- True model parameters $\boldsymbol{b} = [c_{n_c}, \ldots, c_1, a_{n_a}, \ldots a_1, 1]^T$
- Equation error identifier

$$\hat{\boldsymbol{b}}_{t+1} = \hat{\boldsymbol{b}}_t - \mu \begin{bmatrix} \frac{\Pi_1 \boldsymbol{x}_t (\hat{y}_t - y_t)}{1 + \mu \left(\sum_{j=1}^{na} y_{t-j}^2 + \sum_{j=1}^{nc} u_{t-j}^2 \right)} \\ 0 \end{bmatrix}$$

- Persistence of excitation: $\hat{\boldsymbol{b}}_t \rightarrow \boldsymbol{b}$ exponentially if

$$\rho_1 I_{K-1} \prec \sum_{t=j}^{j+S} \Pi_1 \boldsymbol{x}_t \boldsymbol{x}_t^T \Pi_1^T \prec \rho_2 I_{K-1}$$

$$\rho_3 I_{K-1} \prec \sum_{t=j}^{j+S-n_a+1} \boldsymbol{u}_t \boldsymbol{u}_t^T \prec \rho_4 I_{K-1}$$

Center for IMAGING SCIENCE

Overestimating the system order: single mode

$\bar{n}_c, \bar{n}_a$ order upper bounds, $\bar{n}_c > n_c, \bar{n}_a > n_a$

$$y_t = a_1 y_{t-1} + 0 y_{t-2} + c_1 u_{t-1} + 0 c_2 u_{t-2}$$

$$\boldsymbol{b} = \begin{bmatrix} 0 & c_1 & 0 & a_1 & 1 \end{bmatrix}$$

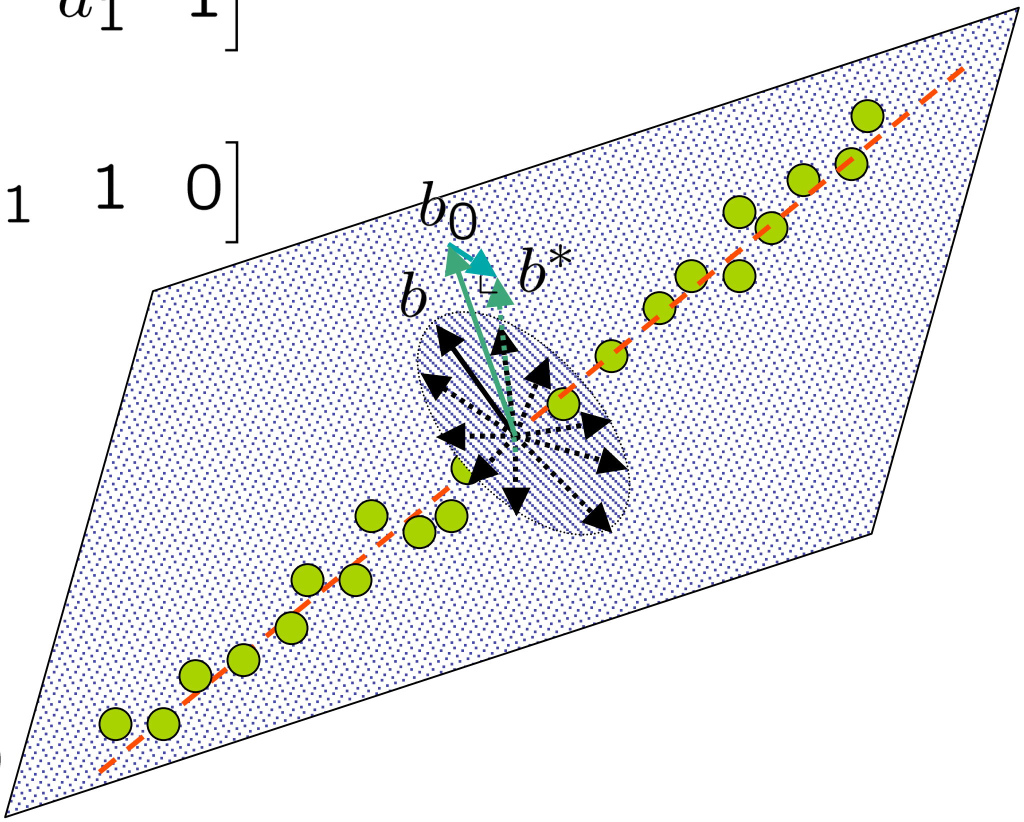
$$y_{t-1} = a_1 y_{t-2} + c_1 u_{t-2}$$

$$\boldsymbol{b}^1 = \begin{bmatrix} c_1 & 0 & a_1 & 1 & 0 \end{bmatrix}$$

There is no unique solution!

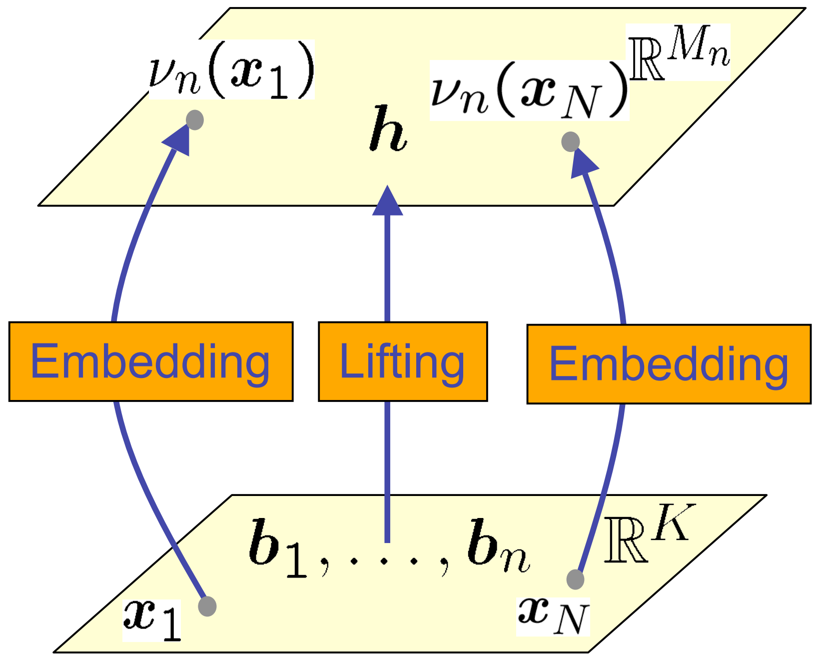
$\hat{b} \to b^*$

$$\boldsymbol{b}^* = \boldsymbol{b} + P^{\perp}(\boldsymbol{b}_0 - \boldsymbol{b})$$



Recursive identification of SARX models

- Identification of a SARX model is equivalent to identification of a single lifted ARX model



- Can apply equation error identifier and derive persistence of excitation condition in lifted space

Recursive identification of hybrid model

- Recall equation error identifier for ARX models

$$\hat{\boldsymbol{b}}_{t+1} = \hat{\boldsymbol{b}}_t - \mu \begin{bmatrix} \frac{\Pi_1 \boldsymbol{x}_t(\hat{y}_t - y_t)}{1+\mu\left(\sum_{j=1}^{na} y_{t-j}^2 + \sum_{j=1}^{nc} u_{t-j}^2\right)} \\ 0 \end{bmatrix}$$

- Equation error identifier for SARX models

$$\hat{\boldsymbol{h}}_{t+1} = \hat{\boldsymbol{h}}_t - \mu \begin{bmatrix} \frac{\Pi_n \nu_n(\boldsymbol{x}_t)(\hat{\boldsymbol{h}}_t^T \nu_n(\boldsymbol{x}_t))}{1+\mu\|\Pi_n \nu_n(\boldsymbol{x}_t)\|^2} \\ 0 \end{bmatrix}$$

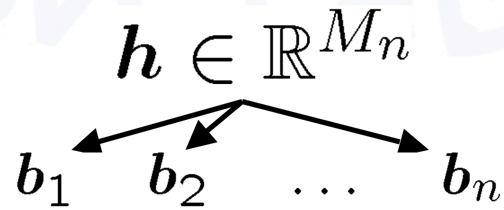
$$\rho_1 I_{M_n(K)-1} \prec \sum_{t=j}^{j+S} \Pi_n \nu_n(\boldsymbol{x}_t) \nu_n^T(\boldsymbol{x}_t) \Pi_n^T \prec \rho_2 I_{M_n(K)-1}$$

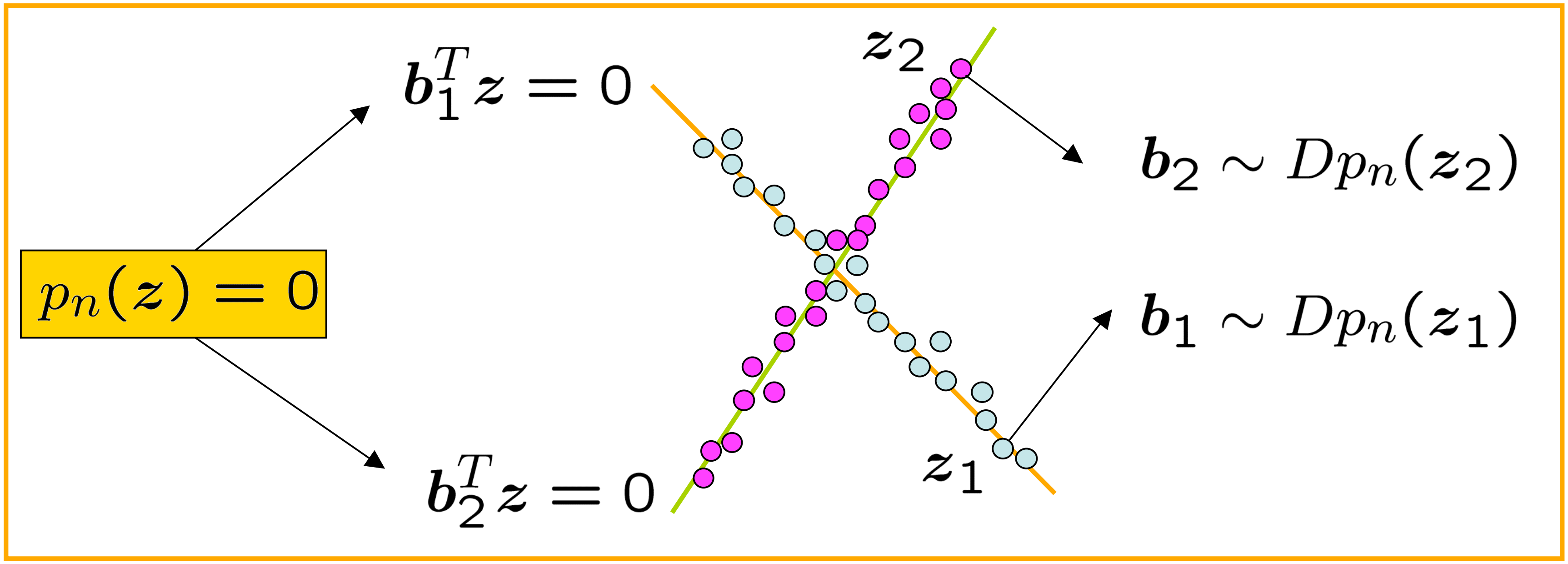
implies that $\boldsymbol{h} - \hat{\boldsymbol{h}}_t \to 0$ exponentially

Center for IMAGING SCIENCE

Recursive identification of ARX models

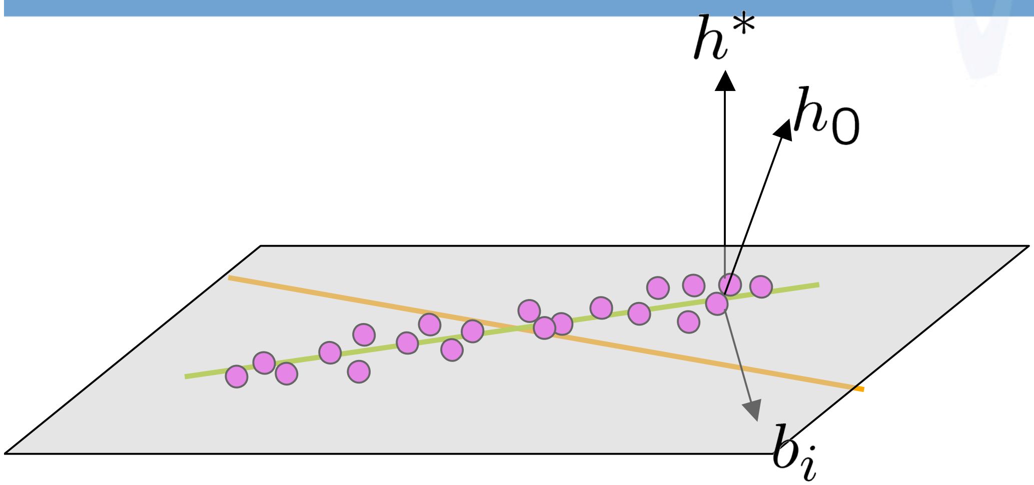
$$p_n(\boldsymbol{z}) = \nu_n(\boldsymbol{z})^T \boldsymbol{h} = (\boldsymbol{b}_1^T \boldsymbol{z}) \cdots (\boldsymbol{b}_n^T \boldsymbol{z})$$





$$\boldsymbol{b}_i = \left. \frac{Dp_n(\boldsymbol{z})}{e_K^T Dp_n(\boldsymbol{z})} \right|_{\boldsymbol{z}=\boldsymbol{z}_i} \Longrightarrow \hat{\boldsymbol{b}}_{\lambda_t} = \frac{D\nu_n^T(\boldsymbol{x}_t)\hat{\boldsymbol{h}}_t}{e_K^T D\nu_n^T(\boldsymbol{x}_t)\hat{\boldsymbol{h}}_t} \qquad \hat{\boldsymbol{b}}_{\lambda_t} \longrightarrow \boldsymbol{b}_i$$

Overestimating the number of modes



$$\hat{h} \longrightarrow h^*$$

$$\boldsymbol{h}^{*T}\nu_n(\boldsymbol{x}) = \boldsymbol{q}_{\bar{n}-n}(\boldsymbol{x})\underbrace{(\boldsymbol{b}_1^T\boldsymbol{x}_t)(\boldsymbol{b}_2^T\boldsymbol{x}_t)\cdots(\boldsymbol{b}_n^T\boldsymbol{x}_t)}_{\boldsymbol{h}^T\nu_n(\boldsymbol{x})} = 0$$

$\hat{\boldsymbol{h}}$ converges exponentially to a multiple of $\boldsymbol{h}$

$$\boldsymbol{b}_{\lambda_t} = \left.\frac{D\nu_n(\boldsymbol{x})^T\boldsymbol{h}}{e_K^T D\nu_n(\boldsymbol{x})^T\boldsymbol{h}}\right|_{\boldsymbol{x}=\boldsymbol{x}_i} \qquad \hat{\boldsymbol{b}}_{\lambda_t} = \left.\frac{D\nu_n(\boldsymbol{x})^T\boldsymbol{h}^*}{e_K^T D\nu_n(\boldsymbol{x})^T\boldsymbol{h}^*}\right|_{\boldsymbol{x}=\boldsymbol{x}_i}$$

$$\hat{\boldsymbol{b}}_{\lambda_t} \longrightarrow \boldsymbol{b}_i$$

Overestimating system order: multiple modes

Given: 2 models estimated to be of the following form:

$$z_i = a_{1,i}y + a_{2,i}x$$

Hybrid decoupling polynomial: $\begin{bmatrix} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 \end{bmatrix} \begin{bmatrix} x^2 \\ xy \\ xz \\ y^2 \\ yz \\ z^2 \end{bmatrix}$

If models are actually of the form

2 $$z_1 = a_{1,1}y + a_{2,1}x, z_2 = a_{1,2}y$$

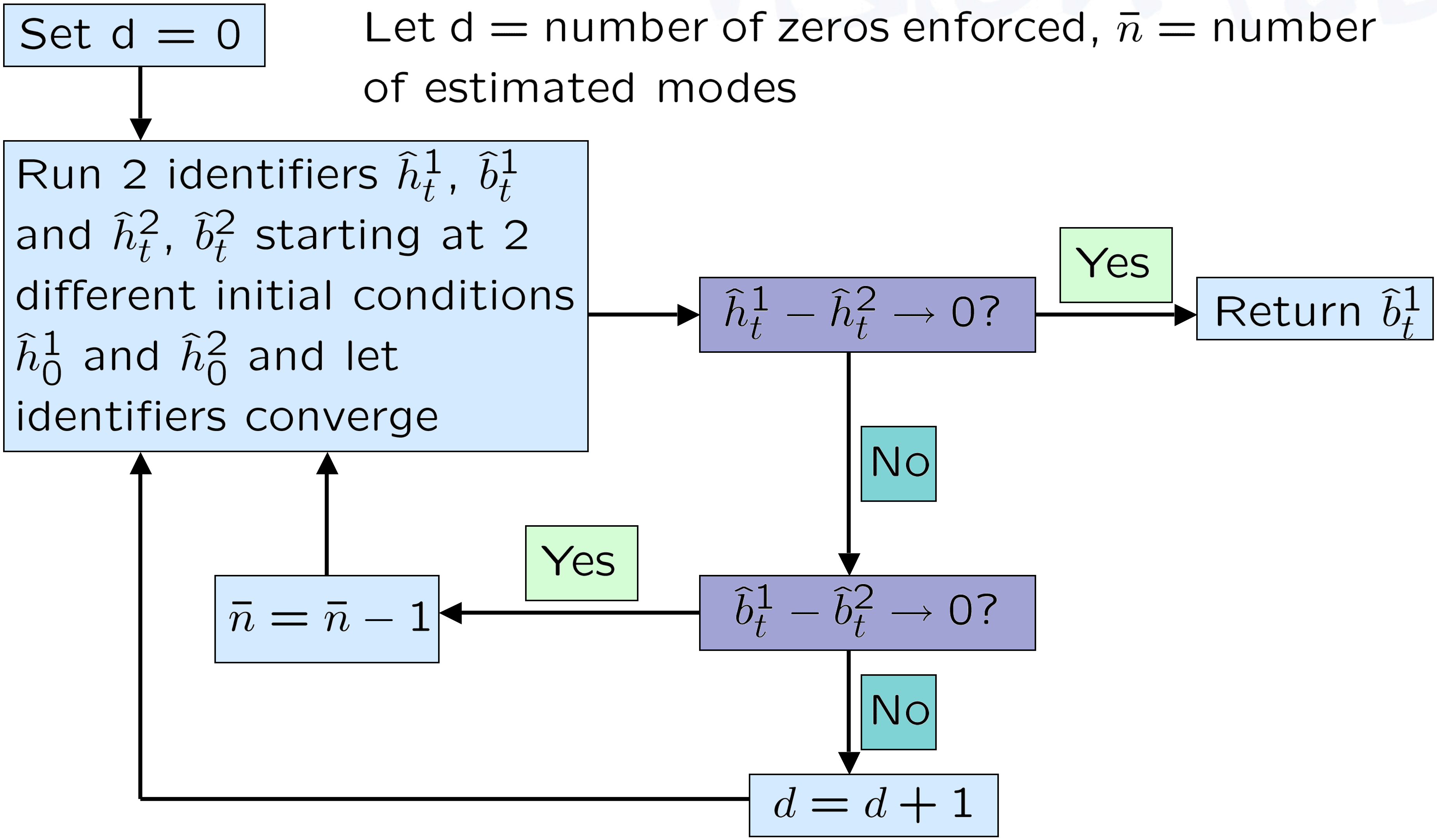
then $h_1 = h_2 = h_3 = 0$

$\widehat{h} \rightarrow h^*, h^*$ is a function of the true parameter vector h and h_0

Using the above idea, enforce zeros in the estimated hybrid parameter vector $\widehat{h}$ to obtain h^*, whose derivatives give the desired parameter vectors b_i

Final recursive identification algorithm

Let d = number of zeros enforced, $\bar{n}$ = number of estimated modes



Experimental results

$$y_t = a(\lambda_t) y_{t-1} + c(\lambda_t) u_{t-1} + w_{t-1}$$

mode $\lambda_t \in \{1, 2\}$ with period 20 sec.
input $u_t \sim \mathcal{N}(0, 1)$,
noise $w_t \sim \mathcal{N}(0, \sigma^2)$

Parameters:
$a(1) = -0.9, a(2) = 0.7$
$c(1) = 0.8, c(2) = -1$
$\boldsymbol{h} = [-0.8, 1.46, -0.2, -0.63, -0.2, 1]^T \in \mathbb{R}^6$

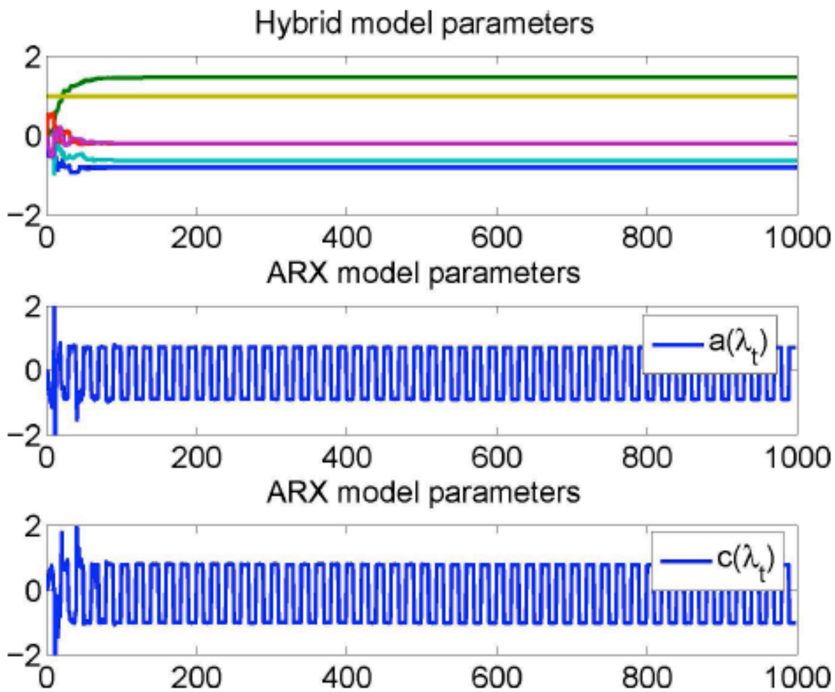
Experiment	$\bar{n}$	$\bar{n}_a$	$\bar{n}_c$	n	n_a	n_c
1	2	1	1	2	1	1
2	2	1	1	4	1	1
3	2	1	1	2	2	2
4	2	1	1	3	2	2

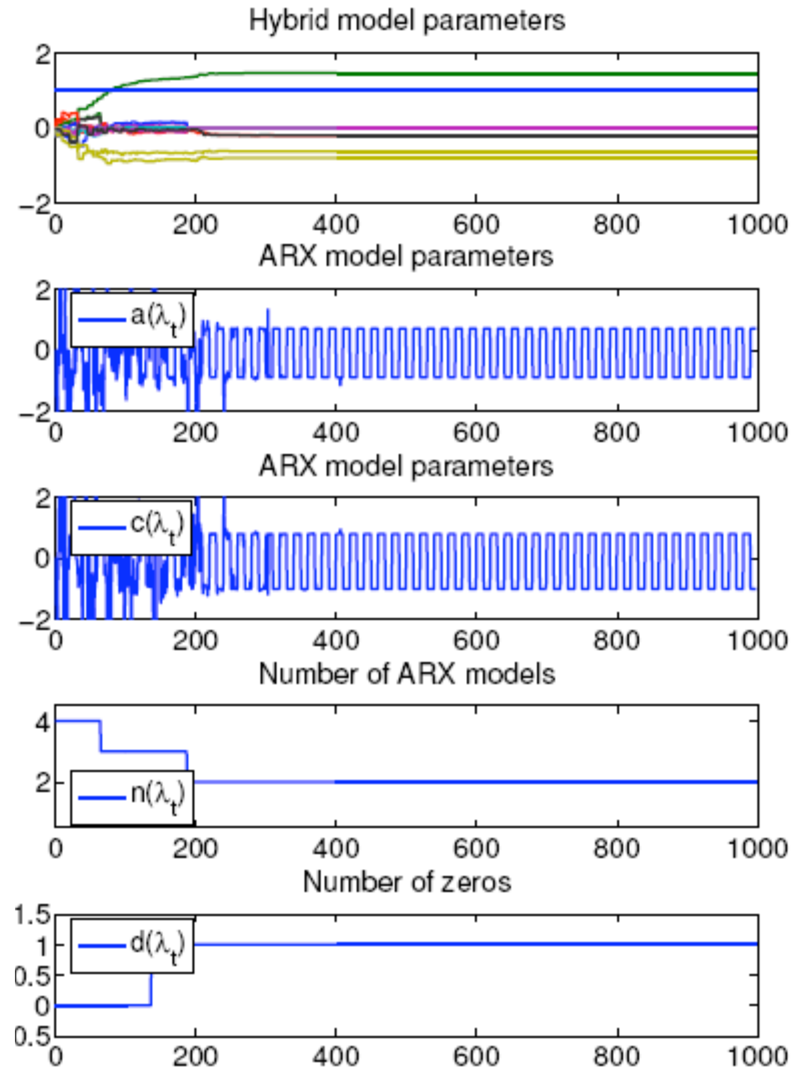
1. correct number of modes and orders
2. correct number of orders, overestimated number of modes
3. correct number of modes, overestimated number of orders
4. overestimated number of modes and orders

Center for IMAGING SCIENCE

Cases 1 & 2 (noiseless)

	$\bar{n}$	$\bar{n}_a$	$\bar{n}_e$	n	n_a	n_c
1	2	1	1	2	1	1
2	2	1	1	4	1	1



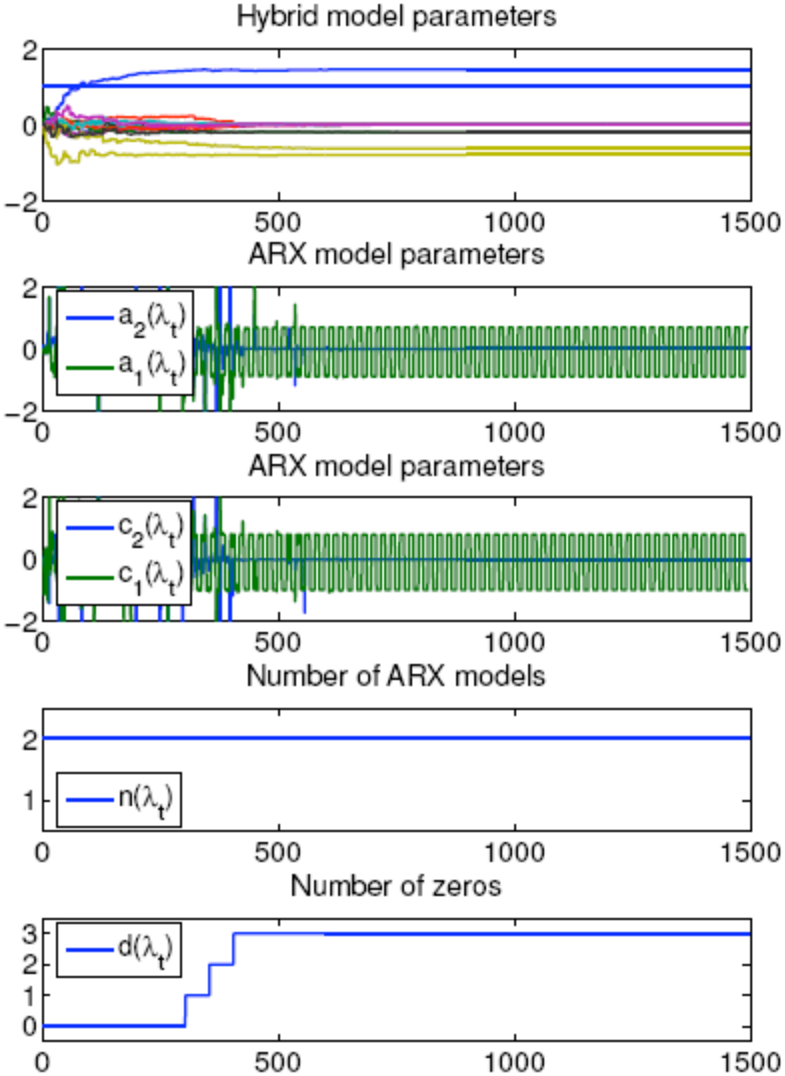


Case 1

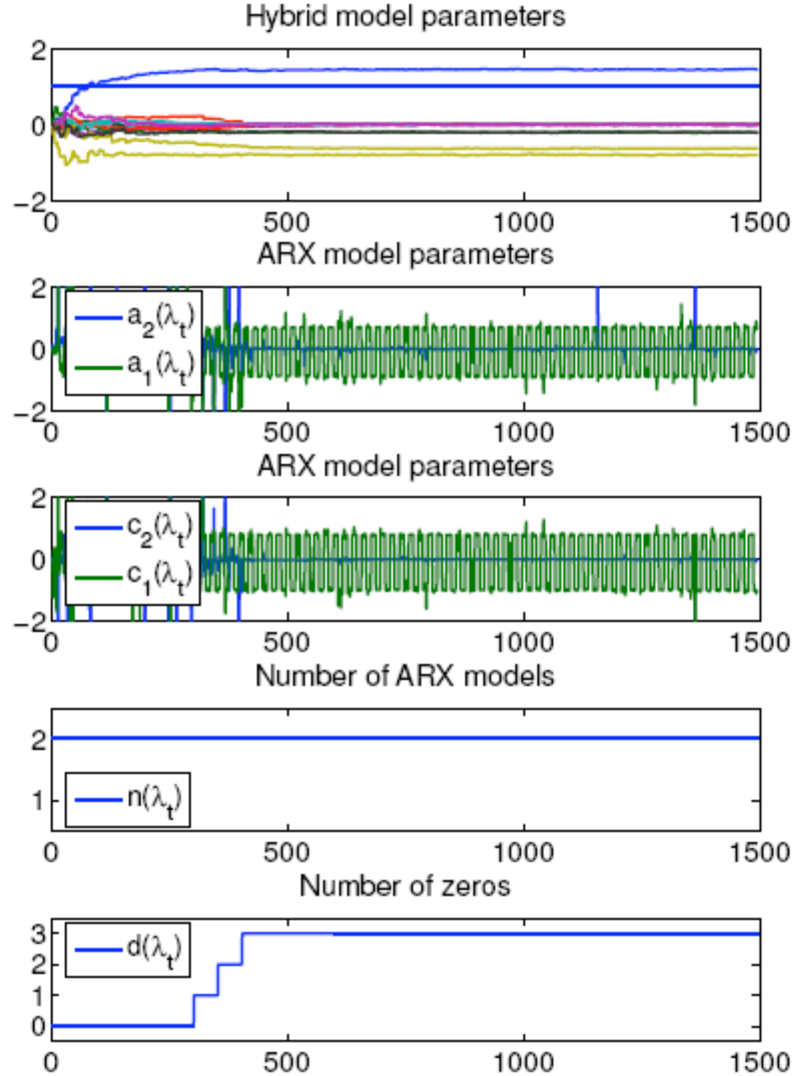
Case 2

Case 3

	$\bar{n}$	$\bar{n}_a$	$\bar{n}_c$	n	n_a	n_c
3	2	1	1	2	2	2



Noiseless case

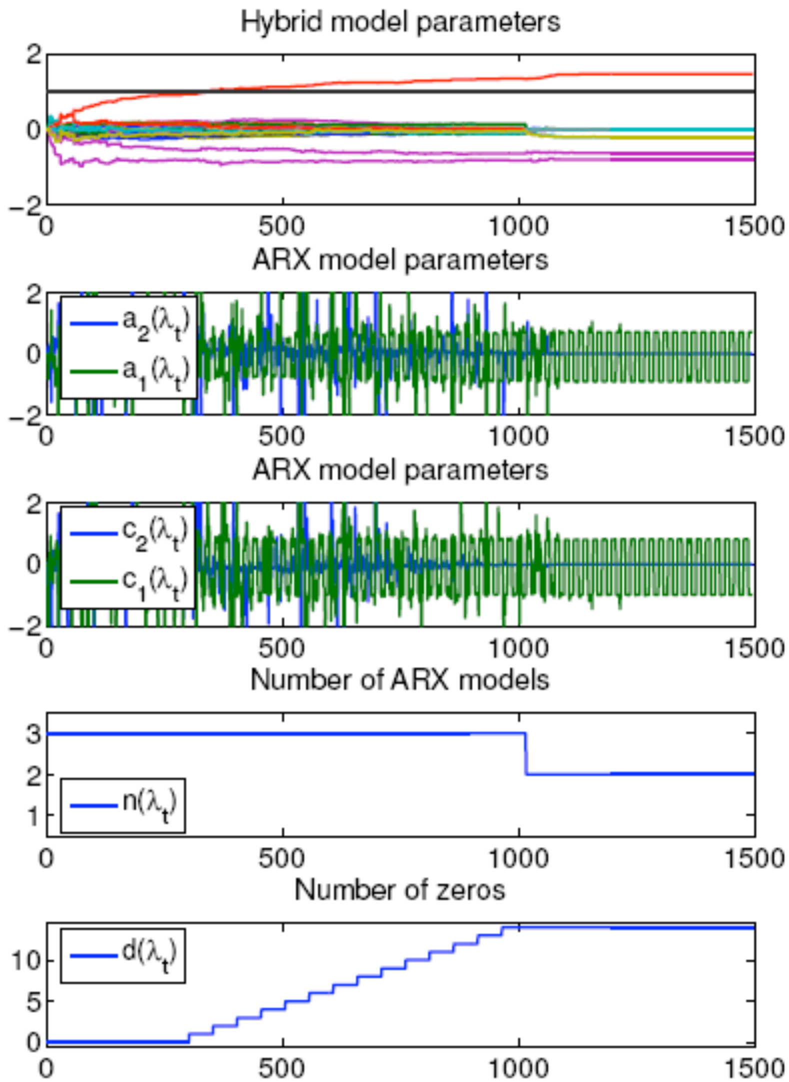


Noisy case, $\sigma = 0.01$

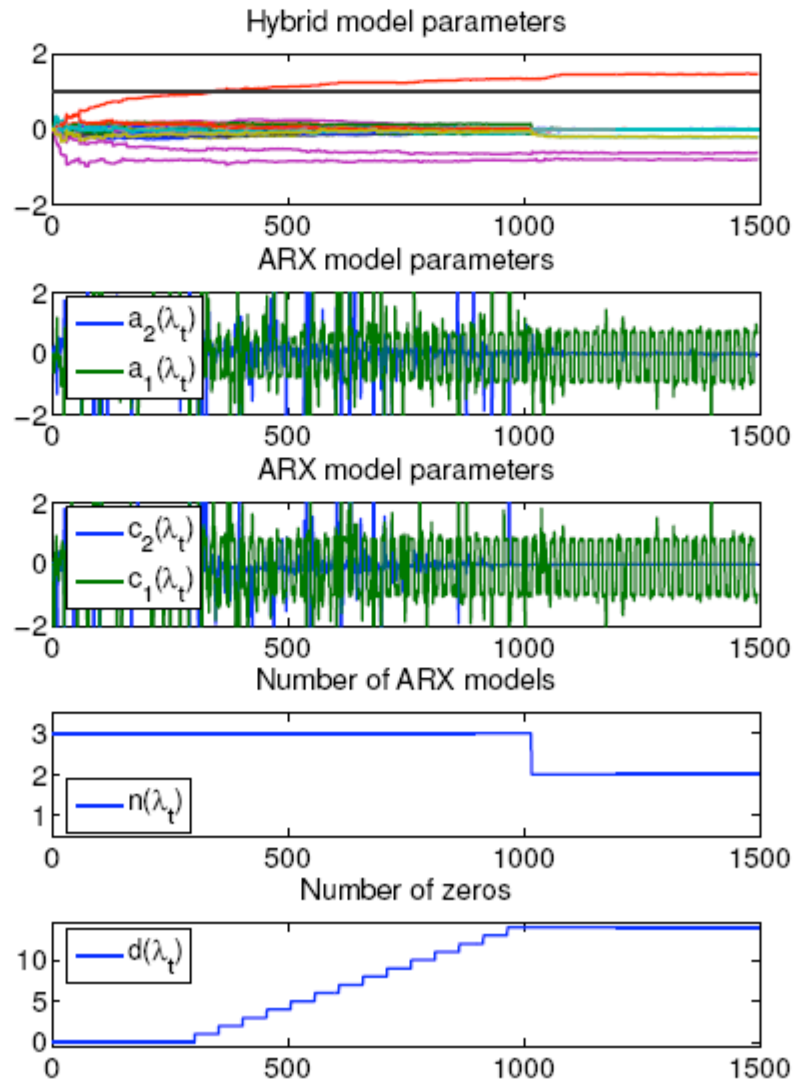
Center for IMAGING SCIENCE

Case 4

	$\bar{n}$	$\bar{n}_a$	$\bar{n}_c$	n	n_a	n_c
4	2	1	1	3	2	2



Noiseless case

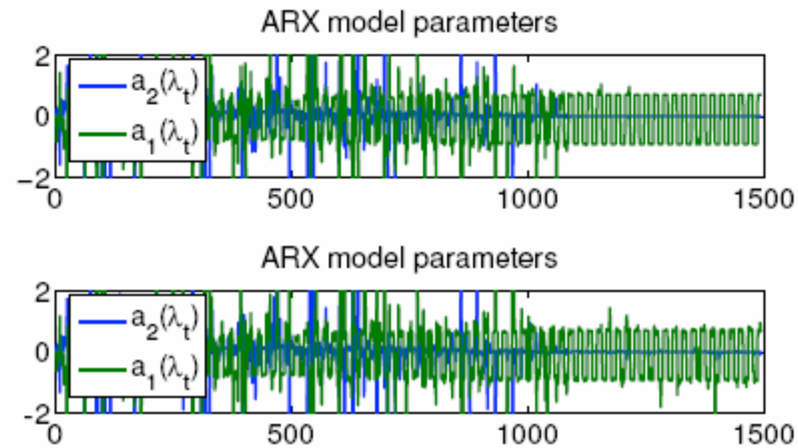


Noisy case, σ = 0.01

Experimental results - summary

Experiment	h and b convergence	Spiking
1	40 ms	None
2	200 ms	Minimal
3 (noiseless)	400 ms	Minimal
3 (noisy)	400 ms	More than noiseless
4 (noiseless)	1100 ms	Minimal
4 (noisy)	1100 ms	Significant

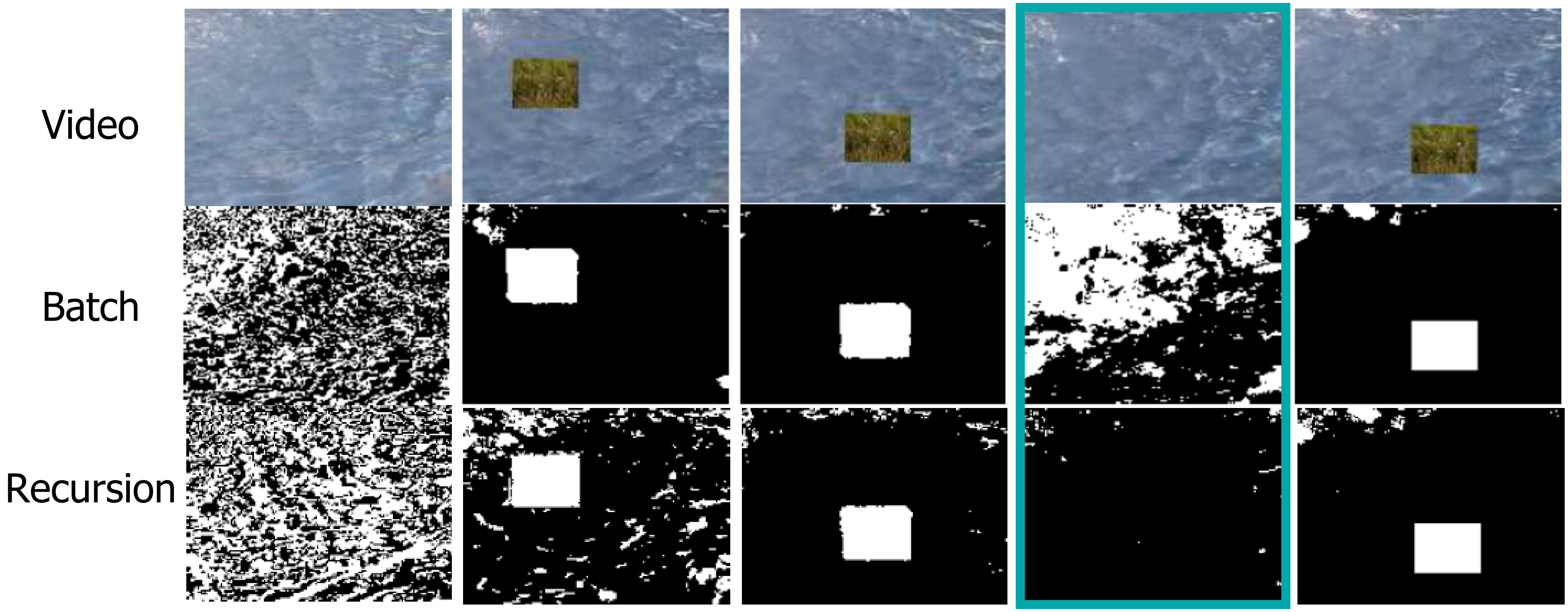
Case 4 Noiseless vs Noisy



Experiment	$\bar{n}$	$\bar{n}_a$	$\bar{n}_c$	n	n_a	n_c
1	2	1	1	2	1	1
2	2	1	1	4	1	1
3	2	1	1	2	2	2
4	2	1	1	3	2	2

Center for IMAGING SCIENCE

Temporal video segmentation



Conclusions and open issues

- Contributions
  - A recursive identification algorithm for hybrid ARX models of unknown number of modes and order
  - A persistence of excitation condition on the input/output data that guarantees exponential convergence

- Open issues
  - Persistence of excitation condition on the mode and input sequences only
  - Extend the model to multivariate SARX models
  - Extend the model to more general, possibly non-linear hybrid systems