# Regularized Least Square Support Vector Machines for Order and Structure Selection of LPV-ARX Models

Manas Mejari, Dario Piga, Alberto Bemporad

IMT School for Advanced Studies, Lucca, Italy.

June 30, 2016

IMT SCHOOL FOR ADVANCED STUDIES LUCCA

# Motivation

- Linear Parameter Varying (LPV) system: an extension of Linear Time Invariant (LTI) framework
- Linear dynamic relation between input and output
- Unlike LTI, the relation changes over time according to a measurable time varying signal, 'scheduling signal' $p$.
- Eg. LPV -ARX models:

$$y(t) = \sum_{i=1}^{n_{\mathrm{a}}} a_i(p(t))y(t-i) + \sum_{i=1}^{n_{\mathrm{b}}} b_i(p(t))u(t-i) + e(t)$$

- LPV model can describe the behaviour of large class of non-linear and time-varying systems.
- Applications: Air-crafts [Marcos et al. '04], Automobiles [Cerone et al '11, Novara et al. '11], Distillation columns [Bachnas et al. '14]

# Problem formulation

## Data-generating system (LPV-ARX)

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}^{\mathrm{o}}} c_i^{\mathrm{o}}(p(t)) x_i(t) + e^{\mathrm{o}}(t)$$

$$x(t) = \begin{bmatrix} y(t-1) & \cdots & y(t-n_{\mathrm{a}}^{\mathrm{o}}) & u(t) & \cdots & u(t-n_{\mathrm{b}}^{\mathrm{o}}) \end{bmatrix}^\top$$

$$c^{\mathrm{o}}(p(t)) = \begin{bmatrix} a_1^{\mathrm{o}}(p(t)) & \cdots & a_{n_{\mathrm{a}}}^{\mathrm{o}}(p(t)) & b_0^{\mathrm{o}}(p(t)) & \cdots & b_{n_{\mathrm{b}}}^{\mathrm{o}}(p(t)) \end{bmatrix}^\top$$

# Problem formulation

## Data-generating system (LPV-ARX)

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}^{\mathrm{o}}} c_i^{\mathrm{o}}(p(t))x_i(t) + e^{\mathrm{o}}(t)$$

## Assumptions

- The dependence of $c_i^{\mathrm{o}}(p(t))$ on the scheduling variable $p(t)$ is not a-priori parametrized.

# Problem formulation

## Data-generating system (LPV-ARX)

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}^{\mathrm{o}}} c_i^{\mathrm{o}}(p(t)) x_i(t) + e^{\mathrm{o}}(t)$$

## Assumptions

- The dependence of $c_i^{\mathrm{o}}(p(t))$ on the scheduling variable $p(t)$ is not a-priori parametrized.
- Sparse Structure, i.e., only few functions $c_i^{\mathrm{o}}(p(t))$ are not zero.

# Problem formulation

## Data-generating system (LPV-ARX)

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}^{\mathrm{o}}} c_i^{\mathrm{o}}(p(t))x_i(t) + e^{\mathrm{o}}(t)$$

## Assumptions

- The dependence of $c_i^{\mathrm{o}}(p(t))$ on the scheduling variable $p(t)$ is not a-priori parametrized.
- Sparse Structure, i.e., only few functions $c_i^{\mathrm{o}}(p(t))$ are not zero.

## Goal

Given the data $\{y(t), u(t), p(t)\}_{t=1}^{N}$ , Estimate :

- the non-zero $p$-dependent functions $c_i^{\mathrm{o}}(p(t))$.
- the correct LPV-model structure i.e. unknowns $n_{\mathrm{a}}^{\mathrm{o}}$ and $n_{\mathrm{b}}^{\mathrm{o}}$

# Problem formulation

LPV-ARX model

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} c_i(p(t)) x_i(t) + e(t)$$

# Problem formulation

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} c_i(p(t)) x_i(t) + e(t)$$

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} \rho_i^\top \phi_i(p(t)) x_i(t) + e(t)$$

# Problem formulation

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} c_i(p(t))x_i(t) + e(t)$$

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} \rho_i^\top \phi_i(p(t))x_i(t) + e(t)$$

- $\rho_i \in \mathbb{R}^{n_{\mathrm{H}}}$ : unknown vector of parameters

# Problem formulation

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_\mathrm{g}} c_i(p(t)) x_i(t) + e(t)$$

$$y(t) = \sum_{i=1}^{n_\mathrm{g}} \rho_i^\top \phi_i(p(t)) x_i(t) + e(t)$$

- $\rho_i \in \mathbb{R}^{n_\mathrm{H}}$ : unknown vector of parameters
- $\phi_i : \mathbb{P} \to \mathbb{R}^{n_\mathrm{H}}$ unknown mapping from the scheduling space to a high-dimensional space (feature space)

# Problem formulation

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} c_i(p(t)) x_i(t) + e(t)$$

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} \rho_i^\top \phi_i(p(t)) x_i(t) + e(t)$$

- $\rho_i \in \mathbb{R}^{n_{\mathrm{H}}}$ : unknown vector of parameters
- $\phi_i : \mathbb{P} \to \mathbb{R}^{n_{\mathrm{H}}}$ unknown mapping from the scheduling space to a high-dimensional space (feature space)
- Neither the maps $\phi_i$ nor the dimension $n_{\mathrm{H}}$ is a-priori specified (potentially, the feature space can be an infinite-dimensional space)

# Problem formulation

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} c_i(p(t))x_i(t) + e(t)$$

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} \rho_i^\top \phi_i(p(t))x_i(t) + e(t)$$

- $\rho_i \in \mathbb{R}^{n_{\mathrm{H}}}$ : unknown vector of parameters
- $\phi_i : \mathbb{P} \to \mathbb{R}^{n_{\mathrm{H}}}$ unknown mapping from the scheduling space to a high-dimensional space (feature space)
- Neither the maps $\phi_i$ nor the dimension $n_{\mathrm{H}}$ is a-priori specified (potentially, the feature space can be an infinite-dimensional space)
- $n_g = n_a + n_b + 1$ such that $n_a > n_a^{\mathrm{o}}$ and $n_b > n_b^{\mathrm{o}}$

# LS-SVM Estimate

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} \underbrace{\rho_i^\top \phi_i(p(t))}_{c_i(p(t))} x_i(t) + e(t)$$

# LS-SVM Estimate

LPV-ARX model

$$y(t) = \sum_{i=1}^{n_\mathrm{g}} \underbrace{\rho_i^\top \phi_i(p(t))}_{c_i(p(t))} x_i(t) + e(t)$$

LS-SVM: primal problem

$$\min_{\rho_i, e} \quad \sum_{t=1}^{N} e^2(t)$$

$$\text{s.t.} \quad e(t) = y(t) - \sum_{i=1}^{n_\mathrm{g}} \rho_i^\top \phi_i(p(t)) x_i(t)$$

# LS-SVM Estimate

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_g} \underbrace{\rho_i^\top \phi_i(p(t))}_{c_i(p(t))} x_i(t) + e(t)$$

## LS-SVM: primal problem

$$\min_{\rho_i, e} \sum_{i=1}^{n_g} \rho_i^\top \rho_i + \frac{\lambda}{2} \sum_{t=1}^{N} e^2(t)$$

$$\text{s.t.} \quad e(t) = y(t) - \sum_{i=1}^{n_g} \rho_i^\top \phi_i(p(t)) x_i(t)$$

# LS-SVM Estimate

## LPV-ARX model

$$y(t) = \sum_{i=1}^{n_g} \underbrace{\rho_i^\top \phi_i(p(t))}_{c_i(p(t))} x_i(t) + e(t)$$

## LS-SVM: primal problem

$$\min_{\rho_i, e} \sum_{i=1}^{n_g} \rho_i^\top \rho_i + \frac{\lambda}{2} \sum_{t=1}^{N} e^2(t)$$

$$\text{s.t.} \quad e(t) = y(t) - \sum_{i=1}^{n_g} \rho_i^\top \phi_i(p(t)) x_i(t)$$

- $\rho_i$ can not be computed as explicit representation of $\phi_i$ is unknown.

# LS-SVM Estimate

## LS-SVM: dual problem

$$\max_{\alpha} \inf_{\rho, e} \mathcal{L}(\rho, e, \alpha)$$

### Lagrangian

$$\mathcal{L}(\rho,e,\alpha) = \frac{1}{2}\sum_{i=1}^{n_g} \rho_i^T \rho_i + \frac{\lambda}{2}\sum_{t=1}^{N} e^2(t) - \sum_{t=1}^{N} \alpha_t \left( e(t) - y(t) + \sum_{i=1}^{n_f} \rho_i^\top \phi_i(p(t)) x_i(t) \right)$$

# LS-SVM Estimate

## KKT Optimality Conditions

$$\frac{\partial \mathcal{L}}{\partial \rho_i} = 0 \rightarrow \rho_i = \sum_{t=1}^{N} \alpha_t \phi_i(p(t)) x_i(t)$$

$$\frac{\partial \mathcal{L}}{\partial e(t)} = 0 \rightarrow e(t) = \frac{1}{\lambda} \alpha_t$$

$$\frac{\partial \mathcal{L}}{\partial \alpha_t} = 0 \rightarrow e(t) = y(t) - \sum_{i=1}^{n_{\mathrm{g}}} \rho_i^\top \phi_i(p(t)) x_i(t)$$

## Output

$$y(t) = \sum_{i=1}^{n_{\mathrm{g}}} \left( \sum_{t=1}^{N} \alpha_t x_i(t) \phi_i^T(p(t)) \right) \phi_i(p(t)) x_i(t) + \frac{1}{\lambda} \alpha_t$$

## Output

$$Y = \left( \sum_{i=1}^{n_g} \Omega_i + \frac{1}{\lambda} I_N \right) \alpha$$

- Kernel Matrix $\Omega_i = X_i \left(\Phi_i\right)^T \Phi_i X_i$

$$\begin{aligned} Y &= [y(1) \quad y(2) \quad \cdots \quad y(N)]^T \\ \Phi_i &= [\phi_i(p(1)) \quad \phi_i(p(2)) \quad \cdots \quad \phi_i(p(N))] \\ X_i &= \mathrm{diag}(x_i(1) \quad x_i(2) \quad \cdots \quad x_i(N)) \end{aligned}$$

# Kernel Trick

## Kernel Trick

$$
\begin{aligned}
\Omega_i &= X_i (\Phi_i)^T \Phi_i X_i \\
[\Omega_i]_{j,k} &= x_i(j) \phi_i^T (p(j)) \phi_i(p(k)) x_i(k) \\
&= x_i(j) K_i(p(j), p(k)) x_i(k)
\end{aligned}
$$

- $K_i(p(j), p(k)) : \mathbb{R}^{n_p} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}$ : A positive definite Kernel function
- Eg. Radial Basis Function (RBF)

$$
K_i(p(j), p(k)) = exp\left( -\frac{\|p(j) - p(k)\|_2^2}{\sigma_i^2} \right)
$$

# LS-SVM Estimate

LS-SVM Coefficients

$$\hat{c}_i(.) = \rho_i^T \phi_i(.) = \sum_{t=1}^{N} \alpha_t K_i(p(t), \cdot) x_i(t)$$

- Coefficients are determined without specifying the underlying dependency on the scheduling parameter.
- However, accurate sparsity structure of LPV model is not selected.
- **Regularized LS-SVM (R-LS-SVM):**
  1. Estimate coefficients $\hat{c}_i(.)$ using LS-SVM
  2. Scale the estimated coefficients using $p-$ dependent polynomial weights
  3. Penalize the weights to shrink the estimated coefficients
  4. Re-estimate the non-null model coefficients.

# Regularized LS-SVM

## Proposed method

- Scale the coefficients with polynomial weights :

$$\tilde{c}_i(.) = w_i(p(t))\hat{c}_i(.)$$
$$\tilde{c}_i(.) = \mathbf{w}_i^T\varphi_i(p(t))\hat{c}_i(.)$$

- Penalize $\|\mathbf{w}_i\|_\infty$ in the cost

- $\mathbf{w}_i \in \mathbb{R}^{n_w}$ : unknown weight vector
- $\varphi_i(p(t)) : \mathbb{R}^{n_p} \to \mathbb{R}^{n_w}$ : a-priory selected basis function.

# Model Order Selection with R-LS-SVM

Scaled Output of LPV-ARX model

$$y(t) = \sum_{i=1}^{n_g} \mathbf{w}_i^T \varphi(p(t))\hat{c}_i(p(t))x_i(t) + e(t)$$

Solve Primal problem

$$\min_{\{\mathbf{w}_i\}_{i=1}^{n_g}} \sum_{t=1}^{N} \left( y(t) - \sum_{i=1}^{n_g} \mathbf{w}_i^T \varphi_i(p(t))\hat{c}_i(p(t))x_i(t) \right)^2 + \mu \left( \sum_{i=1}^{n_g} \|\mathbf{w}_i\|_\infty \right)$$

- The group Lasso penalty $\sum_{i=1}^{n_g} \|\mathbf{w}_i\|_\infty$ attempts at zeroing as many coefficients as possible
- vector $\mathbf{w}_i$ is enforced to be either identically zero or full.

# Simulation Examples

## LPV data-generating system

$$y(t) = a_1^{\mathrm{o}}(p(t))y(t-1) + a_2^{\mathrm{o}}(p(t))y(t-2) + b_5^{\mathrm{o}}(p(t))u(t-5) + e_{\mathrm{o}}(t)$$

$$a_1^{\mathrm{o}}(p(t)) = \begin{cases} -0.5, & \text{if } p(t) > 0.5 \\ -p(t), & \text{if } -0.5 \leq p(t) \leq 0.5 \\ 0.5, & \text{if } p(t) < -0.5 \end{cases}$$

$$a_2^{\mathrm{o}}(p(t)) = \sin(2\pi p(t)); \quad b_5^{\mathrm{o}}(p(t)) = p^3(t)$$

$$N = 500, \qquad \mathrm{SNR} = 7 \mathrm{~dB}$$

# Numerical Example 1

LPV data-generating system

$$y(t) = a_1^{\mathrm{o}}(p(t))y(t-1) + a_2^{\mathrm{o}}(p(t))y(t-2) + b_5^{\mathrm{o}}(p(t))u(t-5) + e_{\mathrm{o}}(t)$$

Over-parameterized LPV model

$$y(t) = \sum_{i=1}^{10} a_i(p(t))y(t-i) + \sum_{j=0}^{10} b_j(p(t))u(t-j) + e(t)$$

- Step 1 : Radial Basis Function kernels ($\sigma = 0.4$) used for LS-SVM , with $\lambda = 600$
- Step 2 : $2^{\mathrm{nd}}$ order polynomials used, penalty on group Lasso $\mu = 5$

# Numerical Example 1

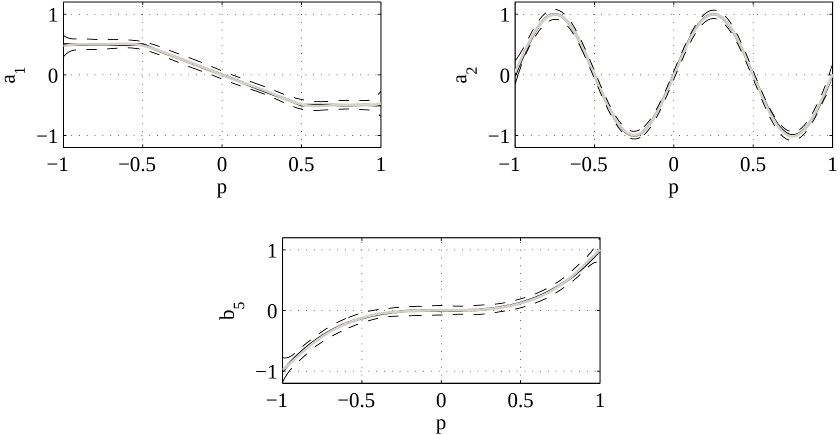
- Mean and Standard deviations of Maximum absolute value of coefficients $a_i(.)$ and $b_i(.)$ over 100 Monte-Carlo Simulations:

| Coef | True Val. | Mean (R-LS-SVM) | Mean (LS-SVM) |
|---|---|---|---|
| $a_1$ | 0.5 | 0.4766 | 0.6245 |
| $a_2$ | 1 | 0.9667 | 1.0498 |
| $a_3$ | 0 | 2.61e-11 | 0.1853 |
| $a_4$ | 0 | 2.25e-11 | 0.1788 |
| $a_5$ | 0 | 2.63e-11 | 0.1858 |
| $a_6$ | 0 | 3.29e-11 | 0.1939 |
| $a_7$ | 0 | 2.47e-11 | 0.1987 |
| $a_8$ | 0 | 2.24e-11 | 0.1891 |
| $a_9$ | 0 | 2.31e-11 | 0.1935 |
| $a_{10}$ | 0 | 3.02e-11 | 0.1904 |

| Coef | True Val. | Mean (R-LS-SVM) | Mean (LS-SVM) |
|---|---|---|---|
| $b_0$ | 0 | 1.79e-11 | 0.1236 |
| $b_1$ | 0 | 1.68e-11 | 0.1372 |
| $b_2$ | 0 | 1.78e-11 | 0.1292 |
| $b_3$ | 0 | 1.73e-11 | 0.1204 |
| $b_4$ | 0 | 1.56e-11 | 0.1213 |
| $b_5$ | 1 | 0.8748 | 1.0133 |
| $b_6$ | 0 | 1.85e-11 | 0.1270 |
| $b_7$ | 0 | 1.99e-11 | 0.1350 |
| $b_8$ | 0 | 1.96e-11 | 0.1333 |
| $b_9$ | 0 | 1.74e-11 | 0.1201 |
| $b_{10}$ | 0 | 1.71e-11 | 0.1329 |

- System Structure Detected!
- BFR (best fit rate)$= \max\left(1 - \frac{\|y - \hat{y}\|_2}{\|y - E[y]\|_2}, 0\right) = 0.92$

# Estimated Coefficients



# Numerical Example 2

## Multidimensional Scheduling variable

- LASSO-like method proposed in [Piga et al. CDC '13] requires to grid the scheduling space
- this limits it's applicability to LPV system with one or two dimensional scheduling vector
- R-LS-SVM proposed here is computationally more efficient and applicable to high dimensions of scheduling space

$$\begin{aligned} y(t) = {} & a_1^{\mathrm{o}}(p(t))y(t-1) + a_2^{\mathrm{o}}(p(t))y(t-2) \\ & + b_4^{\mathrm{o}}(p(t))u(t-4) + b_5^{\mathrm{o}}(p(t))u(t-5) + e_{\mathrm{o}}(t) \end{aligned}$$

$$p(t) = [p_1(t) \quad p_2(t) \quad p_3(t)]^T$$

# Numerical Example 2

LPV data-generating system

$$
\begin{aligned}
y(t) = {} & a_1^{\text{o}}(p(t))y(t-1) + a_2^{\text{o}}(p(t))y(t-2) \\
& + b_4^{\text{o}}(p(t))u(t-4) + b_5^{\text{o}}(p(t))u(t-5) + e_{\text{o}}(t)
\end{aligned}
$$

$$
\begin{aligned}
a_1^{\text{o}}(p(t)) &= 0.3p_1^2(t) + 0.2p_2^2(t) - 0.1p_3^2(t) \\
a_2^{\text{o}}(p(t)) &= 0.2p_1(t) - 0.3p_2(t) + 0.1p_3(t) \\
b_4^{\text{o}}(p(t)) &= 0.2\sin(2\pi p_1(t)) + \sin(2\pi p_2(t)) \\
b_5^{\text{o}}(p(t)) &= 0.4\cos(2\pi p_2(t)) + 0.3\sin(2\pi p_3(t))
\end{aligned}
$$

$$
N = 3000, \qquad \mathrm{SNR} = 15 \text{ dB}
$$

# Numerical Example 2

LPV data-generating system

$$y(t) = a_1^{\mathrm{o}}(p(t))y(t-1) + a_2^{\mathrm{o}}(p(t))y(t-2) \\ + b_4^{\mathrm{o}}(p(t))u(t-4) + b_5^{\mathrm{o}}(p(t))u(t-5) + e_{\mathrm{o}}(t)$$

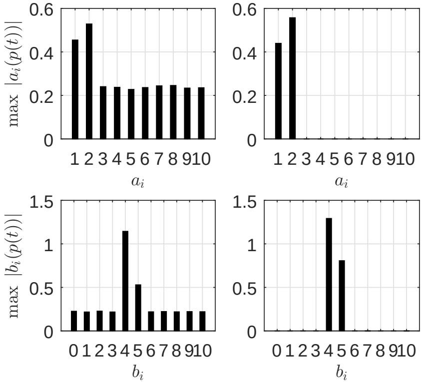
Overparameterized LPV model

$$y(t) = \sum_{i=1}^{10} a_i(p(t))y(t-i) + \sum_{j=0}^{10} b_j(p(t))u(t-j) + e(t)$$

- Step 1 : Radial Basis Function kernels ($\sigma = 0.5$) used for LS-SVM , with $\lambda = 900$
- Step 2 : $2^{\mathrm{nd}}$ order polynomials used, penalty on group Lasso $\mu = 25$

# Estimation Results

- Mean of Maximum absolute value of coefficients $a_i(.)$ and $b_i(.)$ over 100 Monte-Carlo Simulations:
  Left panel (LS-SVM) , right panel (R-LS-SVM)




- Best Fit Rate: 0.88 with R-LS SVM and 0.66 using LS-SVM

# Conclusions

- Data-driven identification of LPV-ARX systems

# Conclusions

- Data-driven identification of LPV-ARX systems
- The model coefficients are estimated without specifying the underlying dependency on the scheduling variables

# Conclusions

- Data-driven identification of LPV-ARX systems
- The model coefficients are estimated without specifying the underlying dependency on the scheduling variables
- Accurate structure of underlying sparse LPV-data generating system is obtained in terms of model order with Regularized LS-SVM

# Conclusions

- Data-driven identification of LPV-ARX systems
- The model coefficients are estimated without specifying the underlying dependency on the scheduling variables
- Accurate structure of underlying sparse LPV-data generating system is obtained in terms of model order with Regularized LS-SVM
- R-LS-SVM is computationally efficient, applicable to systems with high-dimensions of scheduling vector.

Thank You