Parameter Identification in Structured Discrete-Time Uncertainties without Persistency of Excitation

Ouboti Djaneye-Boundjou  
*University of Dayton, stander@udayton.edu*

Follow this and additional works at: [http://ecommons.udayton.edu/stander_posters](http://ecommons.udayton.edu/stander_posters)

**Recommended Citation**  
[http://ecommons.udayton.edu/stander_posters/800](http://ecommons.udayton.edu/stander_posters/800)
Parameter Identification in Structured Discrete-Time Uncertainties without Persistency of Excitation

Oubot Djaneye-Boundjou, Advisor: Raúl Ordóñez
Department of Electrical and Computer Engineering, University of Dayton

**Background**
- System Identification:
  - Function approximation
  - Sys-ID usage: machine learning, adaptive control, ...
  - Present study:
    - Discrete-time (DT) structured uncertainties
    - \( f(x(k)) = \theta^T \phi(x(k)) \) (1)
  - Example:
    - \( f_1(x(k)) = \frac{1}{4} + 10 \exp \left( \frac{-(x(k) - 2)^2}{4} \right) - \frac{1}{b(x(k))} \theta^T \phi(x(k)) \) (2)
  - Approximator:
    - \( \hat{f}(\phi(x(k)), \hat{\theta}(k)) = \hat{\theta}(k)^T \hat{\phi}(x(k)) \) (3)
- Parameter error:
  - \( \hat{\theta}(k) = \theta(k) \) is not computable
- Compute estimation error:
  - \( q(k) = \hat{f}(\phi(x(k)), \hat{\theta}(k)) - f(x(k)) \)
- Note:
  - 1 equation with \( r_0 \) unknowns

**Parameter Identification (PI) Problem**

Drive \( \hat{\theta}(k) \rightarrow 0^+ \) or \( \hat{\theta}(k) \rightarrow \theta \), causing \( q(k) \rightarrow 0 \), as \( k \rightarrow \infty \)

**Motivation**
- PI, i.e., \( \hat{\theta}(k) \rightarrow 0 \), leads to improved estimation performance
- Literature: traditional approximation methods guarantee PI provided persistency of excitation (very restrictive)
- Present study:
  - Develop an adaptive estimator with PI guarantees
  - Relax persistency of excitation requirement

**Normalized Gradient (NG) Descent**
- NG: traditional approach to approximation
- NG adaptation law:
  - \( \hat{\theta}(k+1) = \hat{\theta}(k) - \eta \frac{\phi(x(k)) q(k)}{m^2(k)} \) instantaneous update
  - \( \eta > 0 \): step size or learning rate or gain
  - \( m(k) \): normalization signal ensuring \( \psi(x(k)) = \phi(x(k)) \)
- Lyapunov stability analysis: we can show that \( \hat{\theta}(k) \) remains bounded for all \( k \) if \( 0 < \eta < \pi_{NG} \)
- PI, i.e., \( \hat{\theta}(k) \rightarrow 0 \), only if \( \phi(x(k)) \) is persistently exciting

**Concurrent Learning (CL) Preliminaries**
- CL: first introduced in continuous-time framework
- Use of memory:
  - Record past data: for \( k_0 < \tau_j < k \), with \( j = 1, 2, \ldots, c_z \)
  - History stack of \( \psi(x(\tau_j)) \) vectors
- CL condition: \( \Omega \) contains \( r_0 \) linearly independent \( \psi(x(\tau_j)) \)
  - Less restrictive than persistency of excitation

**Gradient-Based CL in DT**
- Gradient-Based CL adaptation law:
  - Given an initial \( \hat{\theta}(k_0) \),
  - \( \hat{\theta}(k+1) = \hat{\theta}(k) - \eta \frac{\phi(x(k)) q(k)}{m^2(k)} \) (4)
- Estimation error based on recorded data:
  - \( q(k) = \hat{f}(\phi(x(\tau_j)), \hat{\theta}(k)) - f(x(\tau_j)) - \hat{\theta}^T(k) \phi(x(\tau_j)) \)
- Lyapunov stability analysis: granted CL condition is met, \( \Omega = ZZ^T \)
  - Computable in simulation
  - Positive definite and we prove that \( \hat{\theta}(k) \rightarrow 0 \) exponentially (PI) if \( 0 < \eta < \pi_{CL} \)

**Numerical Simulations**
- Here, \( f = f_1 \) is approximated
- \( x \) is varied from \( x_L = -2 \pi \) to \( x_H = 3 \pi \) uniformly
- How good is \( \hat{f} \) if \( \theta(k) \) is frozen at each \( i \) to reconstruct \( f \)? Consider metric
  - \( e(k) = \int_{\tau_i}^{\tau_{i+1}} \| \hat{f}(\phi(x), \hat{\theta}(k)) - f(x) \| dx \)

**Future Work**
- How will CL fare with unstructured uncertainties?
- Apply CL adaptation law within a control loop

**Use of memory**
- History stack of \( \psi(x(\tau_j)) \) vectors
- \( Z \subseteq \mathbb{R}^{r_0} \): vector of \( r_0 \) values
- \( \psi(x(\tau_j)) \) values

**Uncertainty**
- Figure 1: On-line approximation
- Figure 2: Metric \( e(k) \)

**Department of Electrical and Computer Engineering, University of Dayton**