Adaptive Control and Recursive System Identification: Lecture 8

Martin Enqvist

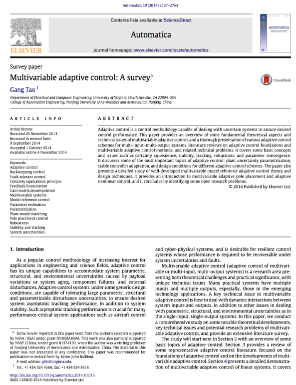
Linköping University

March 24, 2015

Linköpings universitet

Recent Advances

- Renewed interest for aerospace applications
- L_1 adaptive control (?)
- Machine learning approaches
- Multivariable adaptive control
- Adaptive nonlinear control
- New nonlinear filtering methods

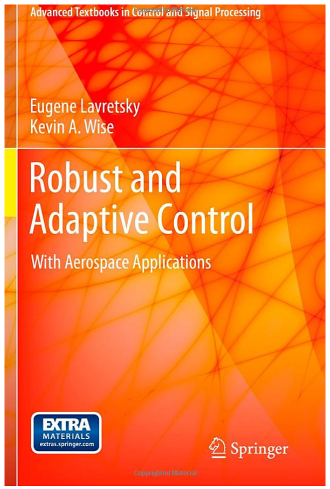


Survey paper (Automatica 2014)

1: Adaptive Control for Aerospace Applications

Some key topics:

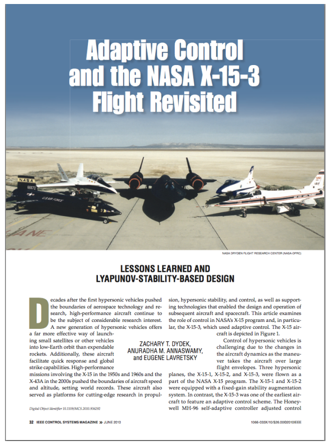
- Robust baseline controllers
- Augmentation structures with adaptive controllers
- Model reference adaptive control
- Reconfigurable flight control
- Several adaptive controllers used in real flight tests



(Springer 2012)

1: Adaptive Control for Aerospace Applications

- On Nov 15, 1967, there was a fatal crash with the NASA X-15-3 test vehicle, partly due to problems with the adaptive flight control system.
- This crash has been discussed and analyzed recently, and most of the anomalous behavior leading up to the crash has been reconstructed in simulations.
- Furthermore, it has been shown in simulations that the accident might have been avoided if a modern adaptive controller would have been used instead of the original one.



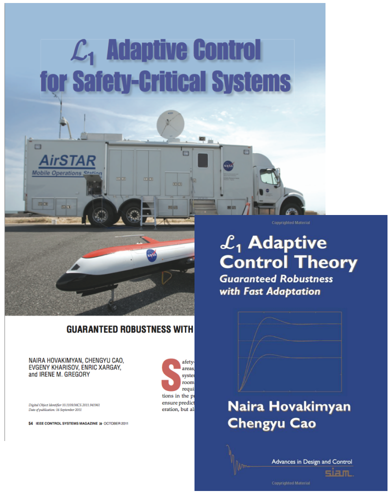
(IEEE CSM 2010)

2: L_1 Adaptive Control

MRAC + stable strictly proper filter at the input

Reported key properties:

- Fast adaptation
- Robust adaptation
- Very high adaptive gains
- L_1 bounds on various signals



(IEEE CSM 2011, SIAM 2010)

2: L_1 Adaptive Control…

System:

$$\dot{x} = A_m x + b\theta^{*T} x + bu,$$
$$y = c^T x,$$

where A_m has all eigenvalues in the LHP. Let

$$\dot{\hat{x}} = A_m \hat{x} + b\theta^T x + bu, \quad \hat{x}(0) = x_0,$$
$$\dot{\theta} = \Gamma x (x - \hat{x})^T P b, \quad \theta(0) = \theta_0,$$

where $A_m^T P + P A_m = -Q$ and $\Gamma > 0$ is a scalar. Lyapunov analysis $\Rightarrow$ $\tilde{\theta} = \theta - \theta^*$ and $\tilde{x} = x - \hat{x}$ bounded. Sufficiently rich and bounded $u \Rightarrow$ $\tilde{\theta}$ and $\tilde{x}$ converge to zero.

2: L_1 Adaptive Control…

MRAC control law:

$$u = -\theta^T x + k_0 r$$

This control law will make x follow the reference model

$$\dot{\hat{x}} = A_m \hat{x} + b_m r, \quad b_m = k_0 b$$

L_1 adaptive control law:

$$u = C(p)(-\theta^T x + k_0 r)$$

where $C(p)$ is a stable strictly proper transfer function with $C(0) = 1$ and $k_0 = -1/(c^T A_m^{-1} b)$

2: L_1 Adaptive Control: Recent Criticism

The additional filter gives

- worse tracking performance compared to standard MRAC
- smaller stability margins compared to standard MRAC

The high adaptive gain may give

- a stiff adaptive law
- worse robustness concerning unmodeled dynamics


L_1-Adaptive Control: Stability, Robustness, and Interpretations

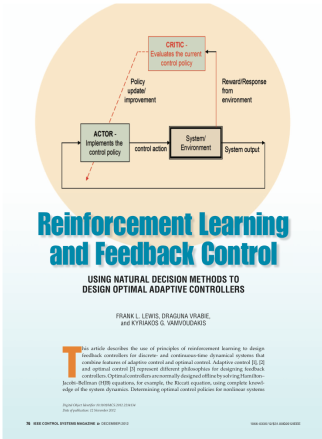
P. A. Ioannou, A. M. Annaswamy, K. S. Narendra, S. Jafari, L. Rudd, R. Ortega, and J. Boskovic

(IEEE TAC 2014)

3: Reinforcement Learning

Basic idea: If an action is followed by some kind of reward or improvement, then there is a tendency to repeat this action (cf. Pavlov's dogs).

- Markov decision processes (MDPs): Select action u when the system is in state x. Based on x and u, the system randomly switches to a new state x' that corresponds to a particular cost (or reward).
- Gives a framework for adaptive optimal control



(IEEE CSM 2012)

3: Reinforcement Learning...

One approach: Q-learning

- Define a Q function such that it represents the expected return for taking the action u in state x and thereafter following an optimal policy. (Q = quality, but the function could also represent a cost.)
- The Q function contains information about control actions in every state, such that the optimal action can be selected knowing only Q.
- The Q function can be estimated online in real time directly form data without knowing the system dynamics.

3: Reinforcement Learning: LQR Example

Example: Online solution of discrete-time LQR using Q-learning (without knowing A and B)

Q function:

$$\tilde{Q}(x_k, u_k) = \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) + V(x_{k+1}),$$

where $V(x) = x^T P x/2$ and P is the solution of the Riccati equation.

Alternative form:

$$\tilde{Q}(x_k, u_k) = \frac{1}{2}\begin{pmatrix} x_k \\ u_k \end{pmatrix}^T \begin{pmatrix} A^T PA + Q & A^T PB \\ B^T PA & B^T PB + R \end{pmatrix} \begin{pmatrix} x_k \\ u_k \end{pmatrix}$$

Let S denote the kernel matrix in $\tilde{Q}$ such that

$$\tilde{Q}(x_k, u_k) = \frac{1}{2}\begin{pmatrix} x_k \\ u_k \end{pmatrix}^T S \begin{pmatrix} x_k \\ u_k \end{pmatrix}$$

3: Reinforcement Learning: LQR Example...

Control action that minimizes $\tilde{Q}(x_k, u_k)$ given x_k:

$$u_k = -S_{uu}^{-1} S_{ux} x_k = -(B^T PB + R)^{-1} B^T PA x_k$$

Main idea: Estimate S online from measured data.

$\tilde{Q}$ can be written as

$$\tilde{Q}(x, u) = \tilde{Q}(z) = W^T \phi(z),$$

where $z = (x^T,\ u^T)^T$ and W contains the elements of S. Now we get:

$$W^T(\phi(z_k) - \phi(z_{k+1})) = \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k) \quad (*)$$

(A parameter estimation problem!)

3: Reinforcement Learning: LQR Example...

Q-learning of LQR:

1. Apply $u_k = -L_k x_k$ at time k (L_k is the current feedback gain). Measure x_{k+1} and compute $u_{k+1} = -L_k x_{k+1}$. Compute $\phi(z_k)$, $\phi(z_{k+1})$ and the updated estimate $\hat{W}_{k+1}$ using $(*)$ and RLS.
2. Unpack the vector $\hat{W}_{k+1}$ into the kernel matrix $\hat{S}_{k+1}$. Define the new feedback gain as

$$L_{k+1} = \hat{S}_{uu,k+1}^{-1} \hat{S}_{ux,k+1}$$

This algorithm solves the Riccati equation online without using any knowledge or estimates of A and B. (N.B. Make sure that there is enough excitation.)

Project Ideas

Ideas for the one-week (2hp) project (optional):

- Make a more complex simulation study of some method for adaptive control and or recursive system identification
- Test some method on a real process (or, in the recursive system identification case, on real data)
- Make a more detailed investigation of the L_1 adaptive control framework.
- Study some theoretical aspect.