Optimization-Based Iterative Learning Control for Trajectory Tracking

Angela Schoellig and Raffaello D'Andrea

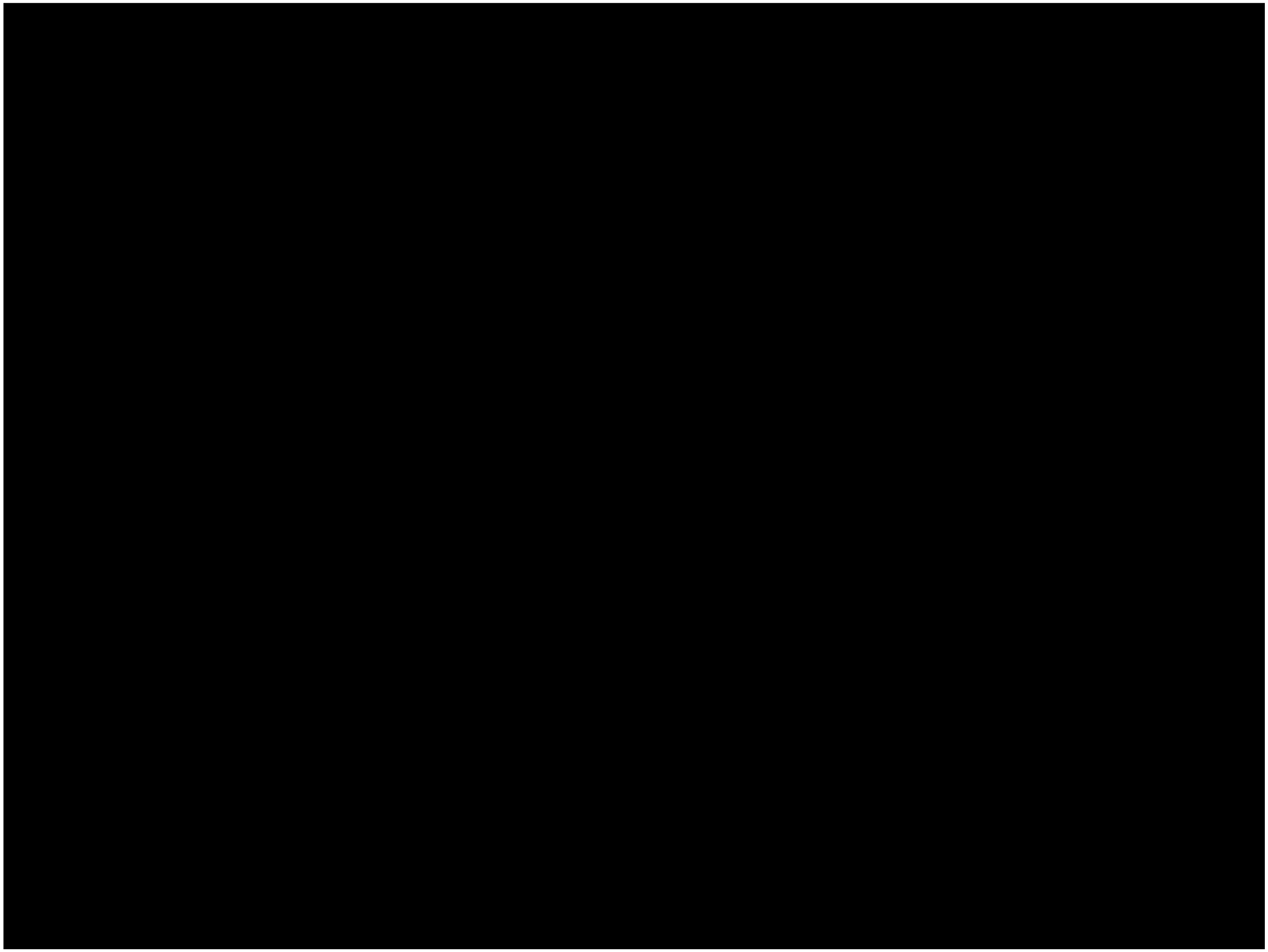
Institute for Dynamic Systems and Control

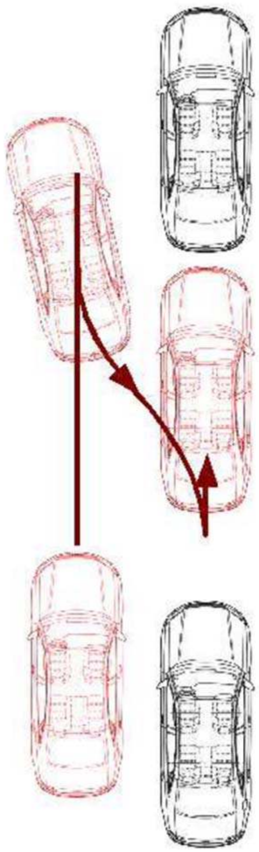
ETH Zürich, Switzerland

European Control Conference 2009 – Budapest, Hungary

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

CHALLENGING… [video]





AUTOMATED SYSTEMS

REPEATED OPERATION

INPUT AND STATE CONSTRAINTS

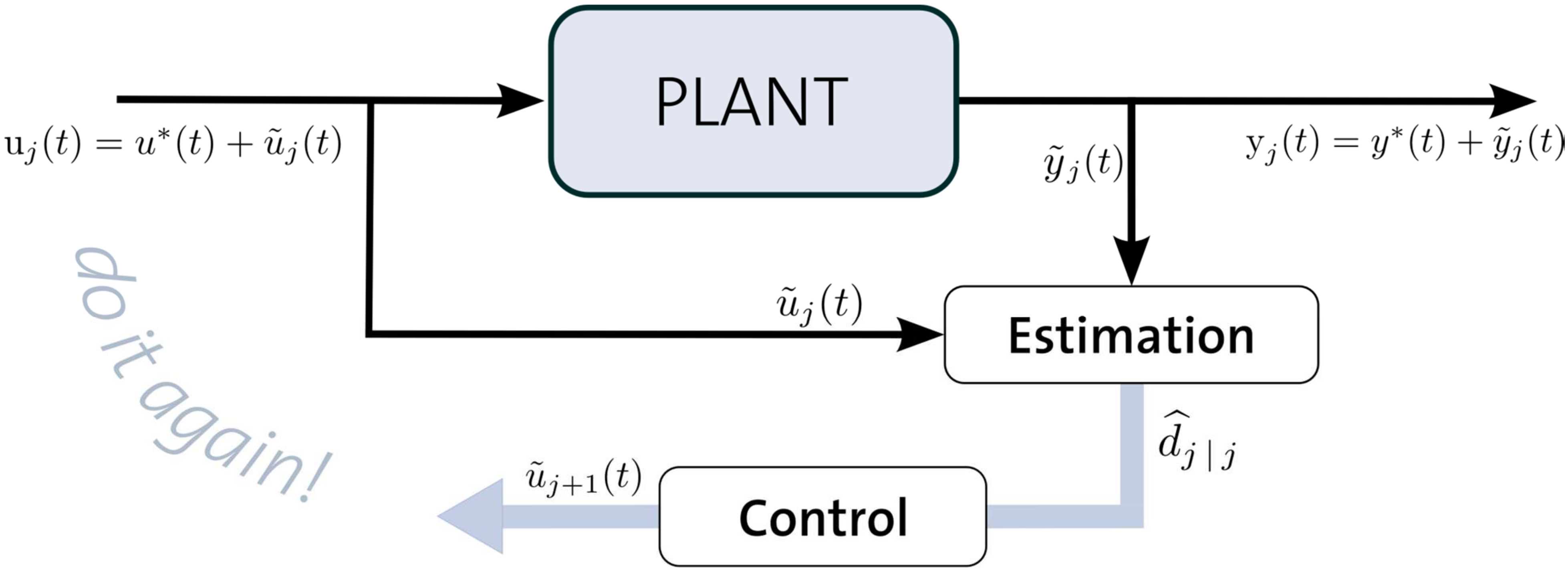
GOAL

High performance trajectory tracking through iterative learning

Taking constraints explicitly into account

… Making full use of system‘s capabilities!

ITERATIVE LEARNING CONTROL



EXECUTE – ESTIMATE – CONTROL

SYSTEM DYNAMICS

G I V E N

Model of the real-world system

$$\dot{\mathrm{x}}(t) = \mathrm{f}(\mathrm{x}(t), \mathrm{u}(t))$$
$$\mathrm{y}(t) = \mathrm{g}(\mathrm{x}(t), \mathrm{u}(t))$$

Input and state constraints

$$\mathrm{u}_{\min} \leq \mathrm{u}(t) \leq \mathrm{u}_{\max}$$
$$\mathrm{x}_{\min} \leq \mathrm{x}(t) \leq \mathrm{x}_{\max}\,, \quad \forall\, t \geq 0$$

Desired trajectory

$$(u^*(t),\, x^*(t),\, y^*(t))\,, \quad t \in [0,\, t_f]$$

LINEARIZE AND DISCRETIZE

Small deviations from nominal trajectory

$$\tilde{u}(t) = \mathrm{u}(t) - u^*(t), \quad \tilde{x}(t) = \mathrm{x}(t) - x^*(t), \quad \tilde{y}(t) = \mathrm{y}(t) - y^*(t)$$

LIFTED-SYSTEM REPRESENTATION

Linear, time-varying difference equations

$$\tilde{x}(k+1) = A_D(k)\tilde{x}(k) + B_D(k)\tilde{u}(k)$$
$$\tilde{y}(k) = C_D(k)\tilde{x}(k) + D_D(k)\tilde{u}(k), \quad k \in \{0, \ldots, N\}$$

! LIFT IT

$$\underbrace{\begin{bmatrix} \tilde{x}(0) \\ \tilde{x}(1) \\ \tilde{x}(2) \\ \vdots \\ \tilde{x}(N) \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ B_D(0) & 0 & \cdots & 0 & 0 \\ \Phi_{(1,1)}B_D(0) & B_D(1) & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \Phi_{(N-1,1)}B_D(0) & \Phi_{(N-1,2)}B_D(1) & \cdots & B_D(N) & 0 \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} \tilde{u}(0) \\ \tilde{u}(1) \\ \tilde{u}(2) \\ \vdots \\ \tilde{u}(N) \end{bmatrix}}_{u}$$

with $\Phi_{(l,m)} = A_D(l)A_D(l+1)\cdots A_D(m), \quad l < m$

and assuming that $\tilde{x}(0) = 0$.

ITERATION-TIME DOMAIN

For trial $j, \; j \in \{1, 2, \dots\}$:

$$\begin{aligned} x_j &= \; F u_j + \boldsymbol{d_j} \qquad + \boldsymbol{N_\xi \, \xi_j} \\ y_j &= \; G x_j \; + \; H u_j \; + \boldsymbol{N_\upsilon \, \upsilon_j} \end{aligned}$$

- Model error along the trajectory

$$d_j = d_{j-1} + \omega_{j-1}$$

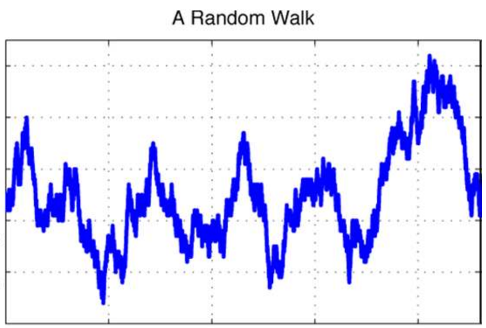
! DESIGN PARAMETER

$$\omega_j \sim \mathcal{N}(0, \Omega_j), \quad \Omega_j = \epsilon_j I, \quad \epsilon_j > 0$$

- Process and measurement noise

trial uncorrelated

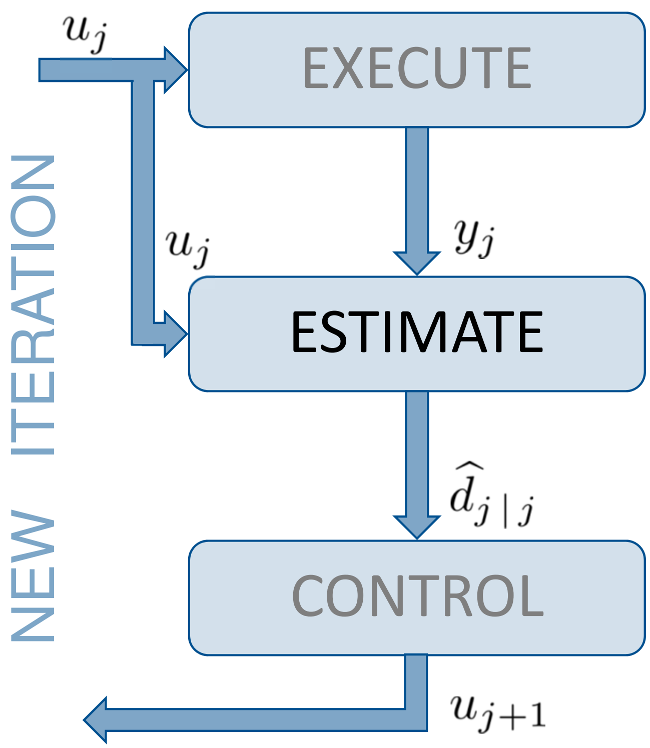
$$\xi_j \sim \mathcal{N}(0, \Xi_j)$$
$$\upsilon_j \sim \mathcal{N}(0, \Upsilon_j)$$



LINEAR, TIME-INVARIANT, DISCRETE

ESTIMATION

$$x_j = F u_j + d_j + N_\xi \xi_j$$
$$y_j = G x_j + H u_j + N_\upsilon \upsilon_j$$
$$d_j = d_{j-1} + \omega_{j-1}$$



$$d_j = d_{j-1} + \omega_{j-1}$$
$$y_j = G\, d_j + (GF + H)\, u_j + \mu_j$$

Error estimate $\widehat{d}_{j\,|\,j} = g\left(\widehat{d}_{j-1\,|\,j-1},\ P_{j\,|\,j-1},\ y_j,\ u_j\right)$

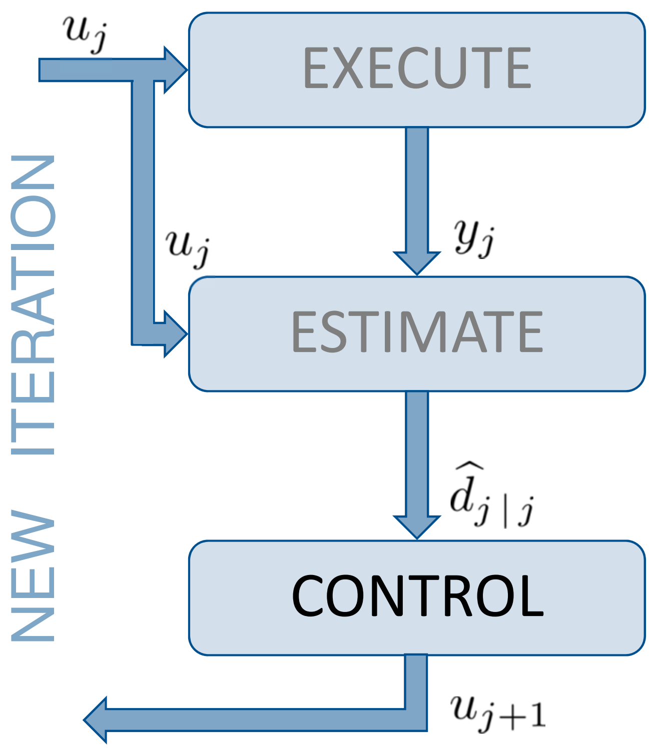
Minimizing $P_{j\,|\,j} = E[(d_j - \widehat{d}_{j\,|\,j})\,(d_j - \widehat{d}_{j\,|\,j})^T]$

Initial conditions $\widehat{d}_{0\,|\,0},\ P_{0\,|\,0}$

KALMAN FILTER IN THE ITERATION DOMAIN

CONTROL

$$x_j = F u_j + d_j + N_\xi \xi_j$$
$$\boldsymbol{x_{j+1} \approx F u_{j+1} + \widehat{d}_{j\,|\,j}}$$



$$\min_{u_{j+1}} \left\| F u_{j+1} + \widehat{d}_{j\,|\,j} \right\|_\ell$$

subject to

$$u_{\min} \leq u_{j+1} \leq u_{\max}$$
$$x_{\min} \leq x_{j+1} \leq x_{\max}$$

Different norms $\ell \in \{1, 2, \infty\}$

Weighting $x^s = W_x x$

! DESIGN PARAMETER

CONVEX PROGRAMMING PROBLEM

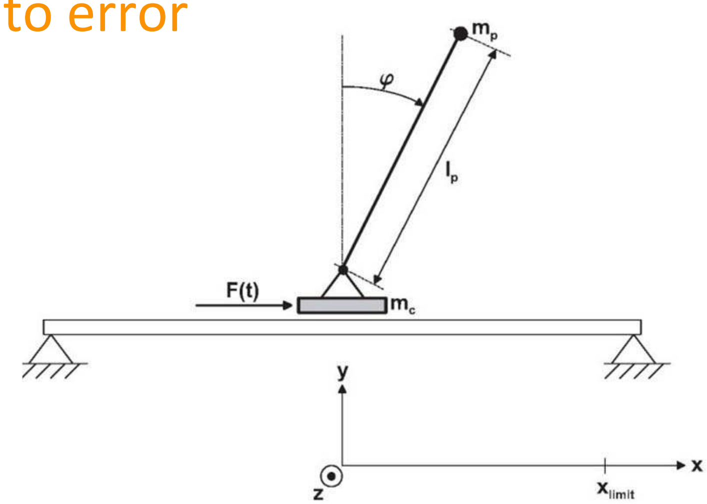
EXPERIMENT

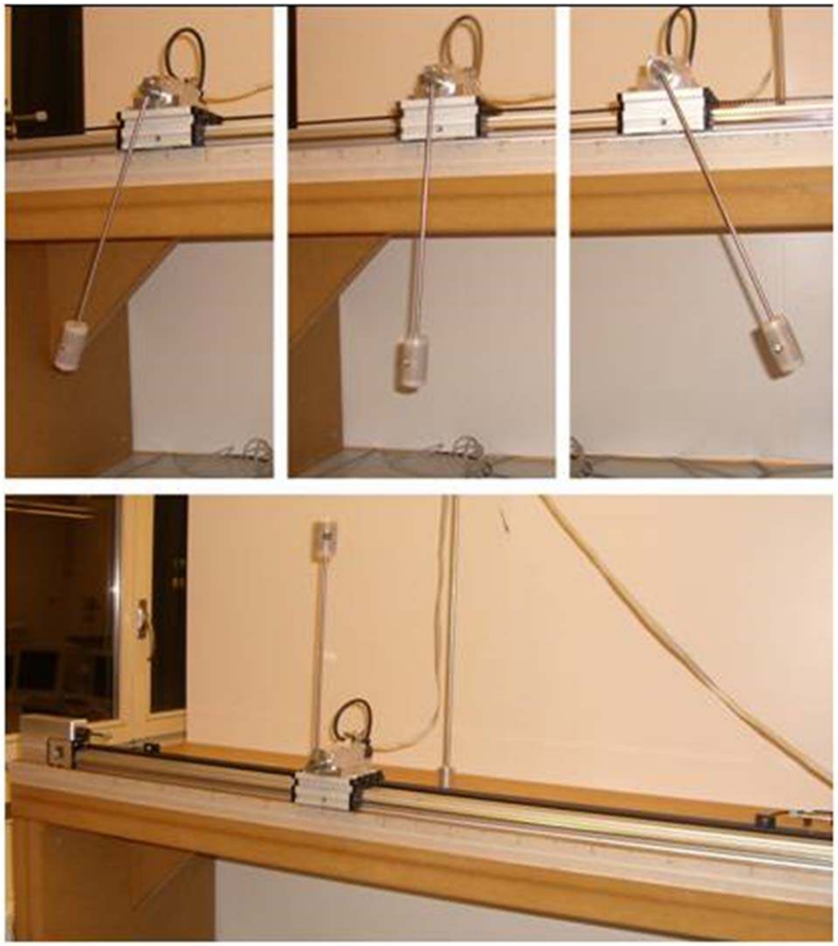
GOAL

Open-loop swing up

CHARACTERISTICS

- Nonlinear, unstable dynamics
- Coarse model
- State and input constraints
- Very sensitive to error



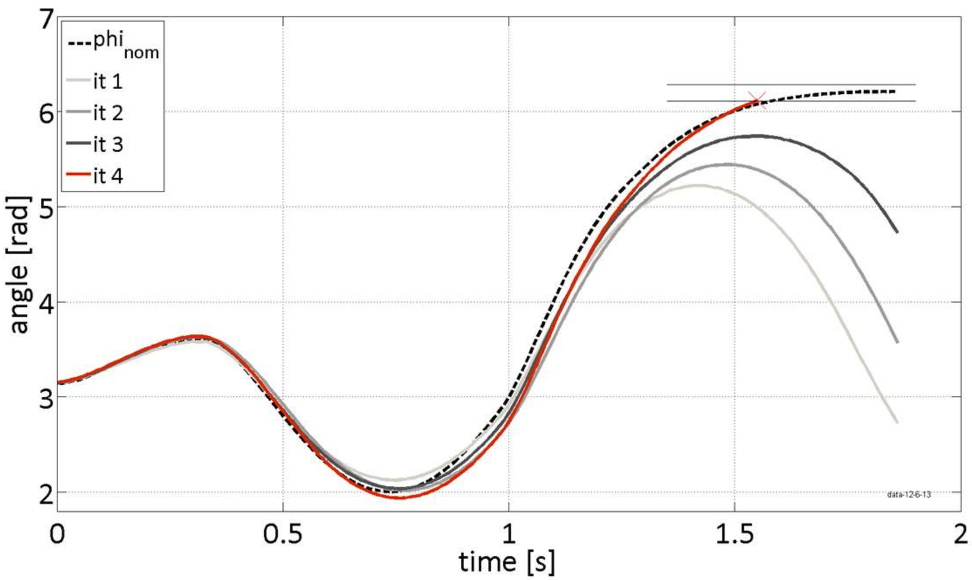


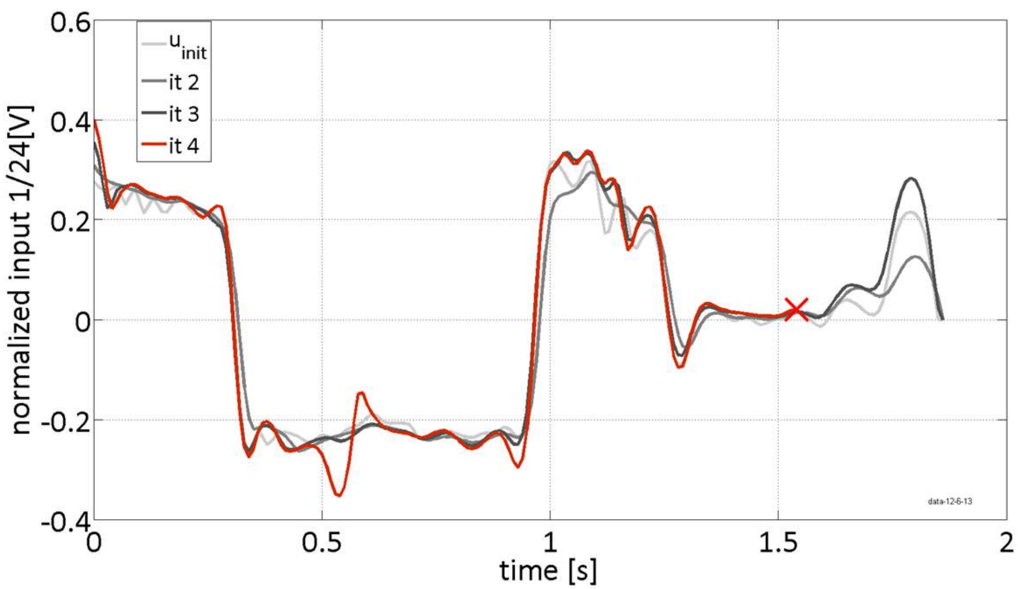
SWING IT UP!

MOVIE https://youtu.be/W2gCn6aAwz4?list=PLC12E387419CEAFF2

SWING IT UP!

MORE RESULTS AND FEATURES (1)





ROBUSTNESS

DOUBLE THE MASS

KEEP SAME MODEL & UPDATE RULE

MORE RESULTS AND FEATURES (2)

Weight on angle rate	Swing up in...
0.006	---
0.012	4th iteration
0.05	6th iteration
0.1	7th iteration

Epsilon	Swing up in...
0.01	6th iteration
0.1	5th iteration
10	4th iteration
100	4th iteration

WEIGHTING

INFLUENCES
LEARNING BEHAVIOR

SPEED OF LEARNING

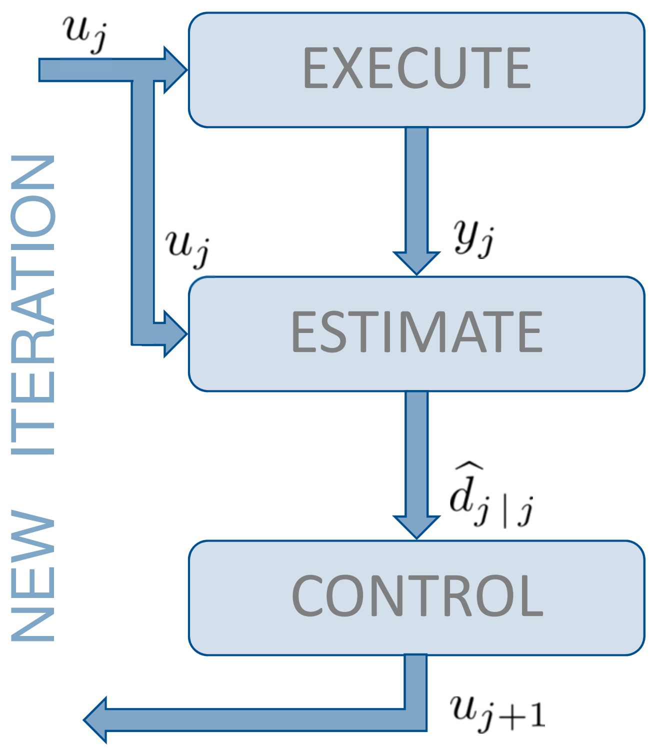
$$d_j = d_{j-1} + \omega_{j-1}$$

$$\omega_j \sim \mathcal{N}(0, \Omega_j), \quad \Omega_j = \epsilon_j I, \quad \epsilon_j > 0$$

NORM

$$\min_{u_{j+1}} \left\| F u_{j+1} + \widehat{d}_{j\,|\,j} \right\|_{\ell}$$

SUMMARY



Repetitive process
Trajectory to be followed
Input and state constraints

OPTIMAL FILTERING: Kalman Filter

CONVEX OPTIMIZATION: Cplex

Fast learning taking constraints explicitly into account.
High tracking performance tapping the system‘s full potential.

FINALLY… [video]