Design of fixed-order stabilizing and $H_2$-$H_\infty$ optimal controllers

Wim Michiels  
Department of Computer Science  
KU Leuven  
Leuven, Belgium

Silviu-Iulian Niculescu  
Laboratoire des Systèmes et Signaux  
Supélec  
Gif-sur-Yvette, France

34th International Summer School of Automatic Control  
Grenoble  
1-5 July 2013
Overview lecture (1h20)

- Limitations of delays on control
  - delay margin
  - stabilizability
- Design of fixed-order / fixed structure controllers
  - Stabilization via nonsmooth, nonconvex optimization
  - Optimization of $H_\infty$ norms
  - Optimization of $H_2$ norms
  - Note on combined criteria
- Case studies
Limitations of delay on control
Limitations of delays in control loops

Delays impose fundamental limitations on stabilizability, achievable performance and robustness.

fundamental = independent of the controller used

Two illustrations
- delay margin
- stabilizability for a fixed value of the delay
Delay margin

system: $\dot{x}(t) = ax(t) + u(t), \ a > 0$

stabilizing control law: $u(t) = kx(t)$ \hspace{1cm} (1)

Delay margin: $\sup \{ \hat{\tau} : \text{null solution is asymptotically stable for all } \tau \in [0, \hat{\tau}] \}$

analysis of closed-loop system

$$\lambda - a - ke^{-\lambda \tau} = 0$$
$$\hat{\lambda} - (a \tau) - (k \tau)e^{-\hat{\lambda}} = 0, \ \hat{\lambda} = \lambda \tau.$$

$\Rightarrow$ maximum achievable delay margin with controller (1) is $1/a$
What if a more sophisticated controller is used?

Result of Middleton & Miller (IEEE TAC, 2007):

Consider a SISO plant with strictly proper rational transfer function $G(s)$ and proper controllers with transfer function $C(s)$.

If $G(s)$ has a RHP pole $re^{j\phi}$, $r > 0$, $\phi \in [0, \pi/2]$, then the maximal achievable delay margin is bounded by

$$\frac{\pi}{r} \sin \phi + \max \left( \frac{2}{r} \cos \phi, \frac{2}{r} \phi \sin \phi \right).$$

⇒ Application to $\dot{x}(t) = ax(t) + u(t)$, $y(t) = x(t)$, $a > 0$:

⇒ Maximum achievable delay margin is bounded by $2/a$

(Explicit construction of controller with margin $2/a - \epsilon$ possible for any $\epsilon > 0$)

⇒ Compared to the very simple linear state feedback controller the margin can at most be doubled!
Stabilizability for a fixed value of the delay

system: \( \dot{x}(t) = Ax(t) + Bu(t - \tau), \quad x(t) \in \mathbb{R}^2, (A, B) \) controllable
\[
\det(\lambda I - A) = \lambda^2 + a_1 \lambda + a_2
\]

control law: \( u(t) = Kx(t), \quad K \in \mathbb{R}^{2 \times 1} \quad u(t) = K(x(t) - x(t - \tau)) \)

Notice: not stabilizable for \( a_2 < 0 \): odd number limitation
Design of fixed-order / fixed structure controllers
Techniques for systems with delays in states, inputs, outputs,…

⇒ fixed-order or fixed structure controllers (including conventional state /output feedback, PID controllers,…) [this lecture]

Control design techniques specifically for systems with delays only in inputs and outputs

⇒ delay compensation / prediction based controllers [principles at the end of the lecture]
Fixed structure / fixed-order control design

Linear(ized) time-delay system

\[ \dot{x}(t) = \sum A_i x(t - \tau_i) + \sum B_i u(t - r_i) \]
\[ y(t) = C_i x(t - s_i) \]

Two mainstream approaches for controller design

1.) based on application of the infinite-dimensional system’s theory

\[ \dot{z}(t) = A \ z(t) + B u(t), \quad z(t) \equiv x(t + \theta), \quad \theta \in [-\tau_{\text{max}}, 0] \]
\[ y(t) = C \ z(t) \]

→ easy to guarantee optimality
→ typically results in a distributed-delay controller
   hard to implement in practice

example: shifting one characteristic root

\[ \dot{x}(t) = A x(t) + B u(t) \quad \lambda I - A : (\text{root, right / left null vector}) \ (\lambda_0, E_0, V^*_0) \]
→ \[ u = k V^*_0 \ x, \quad k \in \mathbb{R} \]

generalization to \[ \dot{x}(t) = \sum A_i x(t - \tau_i) + B u(t) : \]
→ \[ u(t) = k \left( V^*_0 \ x(t) + \int_{-\tau_{\text{max}}}^{0} \left( \sum_{\tau_i < \xi} V^*_0 e^{-\lambda_0 (\xi + \tau_i)} A_i \right) x(t + \xi) \ d\xi \right) \]
Fixed structure / fixed-order control design

Linear(ized) time-delay system
\[ \dot{x}(t) = \sum A_i x(t - \tau_i) + \sum B_i u(t - r_i) \]
\[ y_i(t) = C_i x(t - s_i) \]

Two mainstream approaches for controller design

1.) based on application of the infinite-dimensional system’s theory
\[ \dot{z}(t) = A z(t) + Bu(t), \quad z(t) \equiv x(t + \theta), \quad \theta \in [-\tau_{\text{max}}, 0] \]
\[ y(t) = C z(t) \]
→ easy to guarantee optimality
→ typically results in a distributed-delay controller
   hard to implement in practise

2.) based on a finite-dimensional approximation and the application of standard controller synthesis methods (e.g. pole placement, $H_\infty$ control)
→ order (dimension) of the controller typically $\geq$ dimension plant
→ trade-off between accuracy of the system and implementability of the controller
Fixed structure control design

1. (infinite-dimensional) time-delay system

\[
\begin{align*}
\dot{x}(t) &= \sum A_i x(t - \tau_i) + \sum B_i u(t - r_i) \\
y_i(t) &= \sum C_i x(t - s_i)
\end{align*}
\]

2. any type of controller characterized by finite number of parameters, \( p=(p_1,\ldots,p_m) \)

⇒ closed loop system of the form

\[
\dot{z}(t) = \sum_{i=1}^{m} \tilde{A}_i(p) \ z(t - \tau_i)
\]

⇒ control design = parameter tuning

= optimization of design specifications over the parameters

Motivation

• in applications the structure of the controller is mostly fixed or restricted
• a low order controller often perform well compared to full order controllers (a full order controller is infinite-dimensional!)
• easy to implement
Stabilization via nonsmooth, nonconvex optimization

\[ \dot{x}(t) = A_0(p)x(t) + \sum_{i=1}^{m} A_i(p)x(t - \tau_i) \]

spectral abscissa function:

\[ c(p) = \max_{\lambda \in \mathbb{C}} \left\{ \Re(\lambda) : \det \left( \lambda I - A_0(p) - \sum_{i=1}^{m} A_i(p)e^{-\lambda \tau_i} \right) = 0 \right\} \]

- characterizes the exponential decay of solutions
- the systems is stabilizable if and only if \( \inf_p c(p) < 0 \)

⇒ stabilization approach: minimize \( c \) until the objective function becomes strictly negative
Properties of the spectral abscissa function

\[ c(p) = \max_{\lambda \in \mathbb{C}} \left\{ \Re(\lambda) : \det \left( \lambda I - A_0(p) - \sum_{i=1}^{m} A_i(p) e^{-\lambda \tau_i} \right) = 0 \right\} \]

• not everywhere differentiable

• not locally Lipschitz continuous
• but ... smooth almost everywhere
Generalization of the steepest descent method
takes steps along the *nonsmooth steepest descent direction*:

\[
- \arg \min_{z \in \delta_c \phi(p)} \|z\|,
\]

\[
\partial_c \phi(p) = \text{conv} \left\{ \lim_{q \to p} \nabla c(q) \right\},
\]

> generalized gradient (Clarke subdifferential) at \( p \)
The gradient sampling algorithm (Burke et al, SIOMP 2005)

• approximates the nonsmooth steepest descent direction by randomly sampling gradients in a neighborhood of the current iterate
The gradient sampling algorithm (Burke et al, SIOPT 2005)

- approximates the nonsmooth steepest descent direction by randomly sampling gradients in a neighborhood of the current iterate
- leads to a monotone decrease of the objective function towards a Clarke stationary point: \( \bar{0} \in \partial_c \phi(p) \)

- The algorithm relies on routines to compute the objective function and its gradient, whenever it exists.
  - objective function: via computation of characteristic roots
  - gradient: analytically or numerically (finite differences)

\[
\frac{\partial \lambda(p)}{\partial p_k} = \frac{u^* \left( \frac{\partial A_0(p)}{\partial p_k} + \sum_{i=1}^{m} \frac{\partial A_i(p)}{\partial p_k} e^{-\lambda \tau_i} \right)}{u^* (I + \sum_{i=1}^{m} A_i(p) \tau_i e^{-\lambda \tau_i})} v
\]

- acceleration by BFGS (Broyden-Fletcher-Goldfarb-Shanno quasi-Newton method)
Example

\[ \dot{x}(t) = Ax(t) + Bu(t - \tau), \quad u(t) = K^T x(t) \implies p \equiv K \]

\[
A = \begin{bmatrix}
-0.08 & -0.03 & 0.2 \\
0.2 & -0.04 & -0.005 \\
-0.06 & 0.2 & -0.07
\end{bmatrix}, \quad B = \begin{bmatrix}
-0.1 \\
-0.2 \\
0.1
\end{bmatrix}, \quad \tau = 5.
\]

Spectral abscissa for \( K = 0 \) satisfies \( \alpha \approx 0.108 > 0 \).

Application of optimization algorithm:
Minimization of $\mathcal{H}_\infty$ norms

$$G(j\omega; \ p) := C(p) \left( j\omega I - A_0(p) - \sum_{i=1}^{m} A_i(p) e^{-j\omega \tau_i} \right)^{-1} B(p) + D(p)$$

• The function

$$p \mapsto \beta(p) := \|G(j\omega; \ p)\|_{\mathcal{H}_\infty}$$

has similar properties as the spectral abscissa function (not everywhere differentiable, smooth almost everywhere)

$\Rightarrow$ gradient sampling algorithm, accelerated by BFGS

Expression for gradient (whenever it exists): via expression for sensitivity of an (isolated) singular value:

$$\begin{align*}
\begin{array}{ll}
G(j\omega; \ p)v &= \sigma_i u \\
u^* G(j\omega; \ p)) &= \sigma_i v^*
\end{array}
\Rightarrow \frac{\partial \sigma_i(G(j\omega; \ p))}{\partial p_k} = \Re\left(u^* \frac{\partial G(j\omega; \ p)}{\partial p_k} v\right)
\end{align*}$$

• A stabilizing starting value can be generated by minimizing spectra abscissa

$\Rightarrow$ 2 step approach
Example (continued)

\[ \dot{x}(t) = Ax(t) + Bu(t - \tau), \quad u(t) = K^T x(t) \]

\[
A = \begin{bmatrix}
-0.08 & -0.03 & 0.2 \\
0.2 & -0.04 & -0.005 \\
-0.06 & 0.2 & -0.07
\end{bmatrix}, \quad B = \begin{bmatrix}
-0.1 \\
-0.2 \\
0.1
\end{bmatrix}, \quad \tau = 5.
\]

Objective: optimizing stability robustness

Consider uncertainty on \( A \):
\[ \delta A \in \mathbb{C}^{n \times n}, \quad \| \Delta \|_{\text{glob}} = \| \delta A \|_2 \]

\( \rightarrow \) complex stability radius \( r_{\mathbb{C}}(\| \cdot \|_{\text{glob}}) = \left( \left\| \left( j\omega I - A - BK^T e^{-j\omega \tau} \right)^{-1} \right\|_{\mathcal{H}_\infty} \right)^{-1} \)

\( \rightarrow \) maximizing stability radius \( \equiv \) minimizing a \( \mathcal{H}_\infty \) norm
Result of algorithm starting from minimizer of spectral abscissa function:

<table>
<thead>
<tr>
<th>optimal</th>
<th>$c(K^*)$</th>
<th>$r_C(| \cdot |_{\text{glob}}; K^*)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c$</td>
<td>-0.15</td>
<td>0.0938</td>
</tr>
<tr>
<td>$r_C(| \cdot |_{\text{glob}})$</td>
<td>-0.073</td>
<td>0.287</td>
</tr>
</tbody>
</table>

→ Notice usual trade-off between performance and robustness
Optimization of $\mathcal{H}_2$ norms

$$G(j\omega) = C(p) \left( j\omega I - A_0(p) - \sum_{k=1}^{m} A_k(p)e^{-j\omega \tau_k} \right)^{-1} B(p)$$

- in contrast to the $\mathcal{H}_\infty$ norm, the $\mathcal{H}_2$ norm of $G$ smoothly depends on $p$, provided that the system matrices do so

- expressions for derivatives available

$\rightarrow$ embedding in a derivative based optimization framework
$\rightarrow$ second order methods applicable

<table>
<thead>
<tr>
<th>spectral abscissa, $\mathcal{H}_\infty$ norm</th>
<th>nonsmooth function of parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{H}_2$ norm</td>
<td>smooth function of parameters</td>
</tr>
</tbody>
</table>
Optimization of $\mathcal{H}_2$ norms

$$G(j\omega) = C(p) \left( j\omega I - A_0(p) - \sum_{k=1}^{m} A_k(p)e^{-j\omega\tau_k} \right)^{-1} B(p)$$

• in contrast to the $\mathcal{H}_\infty$ norm, the $\mathcal{H}_2$ norm of $G$ *smoothly* depends on $p$, provided that the system matrices do so

• expressions for derivatives available

→ embedding in a derivative based optimization framework
→ second order methods applicable

<table>
<thead>
<tr>
<th>spectral abscissa, $\mathcal{H}_\infty$ norm</th>
<th>nonsmooth function of parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>smoothed spectral abscissa (Vanbiervliet et al, SIOPT 2009)</td>
<td></td>
</tr>
<tr>
<td>$\mathcal{H}_2$ norm</td>
<td>smooth function of parameters</td>
</tr>
</tbody>
</table>
Case studies
Control of a heating system

Model Vyhlídal, et al. (2009)

System
linear system, dimension 10, 7 delays

Extended state vector

\[ \bar{x}(t) = [x(t)^T \ I(t)]^T \]
\[ I(t) = \int_\tilde{\eta}^t (y_{SET}(\eta) - y(\eta)) d\eta \]

- controlled variable and its setpoint

Controller

\[ u_c(t) = -K^T \bar{x}(t) \]

- control input

→ 11 free parameters
Objective of the control

- acceleration of the set-point response
- achieving a proper damping of the step and disturbance response

Approach

- minimizing the spectral abscissa
- subject to: pole location constraints

\[
\dot{x}(t) = A_0(p)x(t) + \sum_{i=1}^{m} A_i(p) x(t - \tau_i)
\]

assigning a real pole \(c\):

\[
\det \left( cI - A_0(p) - \sum_{i=1}^{m} A_i(p)e^{-c\tau_i} \right) = 0
\]

assigning a pair of complex conjugate poles \(c \pm dj\):

\[
\det \left( (c \pm dj)I - A_0(p) - \sum_{i=1}^{m} A_i(p)e^{-(c \pm dj)\tau_i} \right) = 0
\]

Matrices \(A_i\) linear in \(p\)

\(\rightarrow\) polynomial constraints on parameters \(p\)

In addition: 1 control input

\(\rightarrow\) linear constraints, that can be eliminated
Stability optimization

spectrum of the open loop system

result of minimizing the spectral abscissa
Assigned poles:

\[ \lambda_1 = -0.025 \]
\[ \lambda_2 = -0.035 \]
\[ \lambda_{3,4} = -0.03 \pm 0.03i \]

Evolution of the objective function, and the gain values:
Set-point and disturbance responses
Coupled PDE-DDE model for a semiconductor laser

Spatial discretization: time-delay system of retarded type
dimension n=123, 1 delay
Coupled PDE-DDE model for a semiconductor laser

Spatial discretization: time-delay system of retarded type
dimension $n=123$, 1 delay

Invariant characteristic root at zero: due to symmetry
Note: invariant roots may also appear due to a transformation

Example:

\[
\dot{x}(t) = A_0 x(t) + \int_{t-\tau}^{t} A_1 x(s) \, ds, \quad x(t) \in \mathbb{R}^n
\]

\[
\begin{align*}
\dot{x}(t) &= A_0 x(t) + y(t) \\
y(t) &= \int_{t-\tau}^{t} A_1 x(s) \, ds
\end{align*}
\]

\[
\Rightarrow \begin{align*}
\dot{x}(t) &= A_0 x(t) + y(t) \\
\dot{y}(t) &= A_1 x(t) - A_1 x(t - \tau)
\end{align*}
\]

distributed delay \quad point-wise delay

Transformation (differentiation) introduces \( n \) characteristic roots at zero

Similar techniques apply to distributed delays,

\[
\int_{t-\tau}^{t} K(t-s) x(s) \, ds,
\]

with a Gamma-distribution kernel \( K \)

Introduced or non-physically meaningful roots are known and can be removed from the eigenvalue computations, ...

\[ \Rightarrow \] methods for point-wise delays extend to classes of distributed delays!
Software and references
Software

Overview of software for delay equations available at

MATLAB software for control design problems ( $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms, pseudospectral abscissa), available at
http://twr.cs.kuleuven.be/research/software/delay-control/

Software for corresponding problems without delay: HIFOO
(HIFOO= $H_\infty$-$H_2$ Fixed Order Optimization)
http://www.cs.nyu.edu/overton/hifoo/

controller dimension $m$ \[ m < n \]
plant dimension $n$

\[ \text{delay system} = \text{limit case for } n \rightarrow \infty \]

Additional references


Conclusions of Lecture 3

- Overview of numerical methods to solve synthesis problems: stabilization, optimization of system norms,…

- Based on a direct optimization approach
  
  + not conservative
  + generally applicable
  - local method for usually nonconvex problems

- Targeted methods for systems with input and output delays: prediction based controllers