COMPUTATION OF OPTIMAL ACTUATOR LOCATIONS

Kirsten Morris

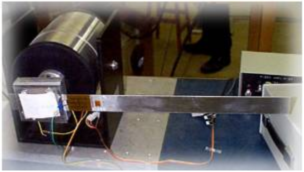
University of Waterloo
Waterloo, CANADA

February 7, 2012

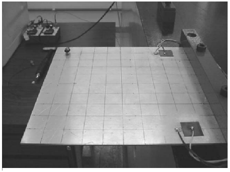
Work with: Neda Darivandi, Dhanaraja Kasinathan, Amir Khajepour, Steven Yang

Sponsored by: AFOSR, NSERC, SharcNet

Control Systems Modelled by PDE's



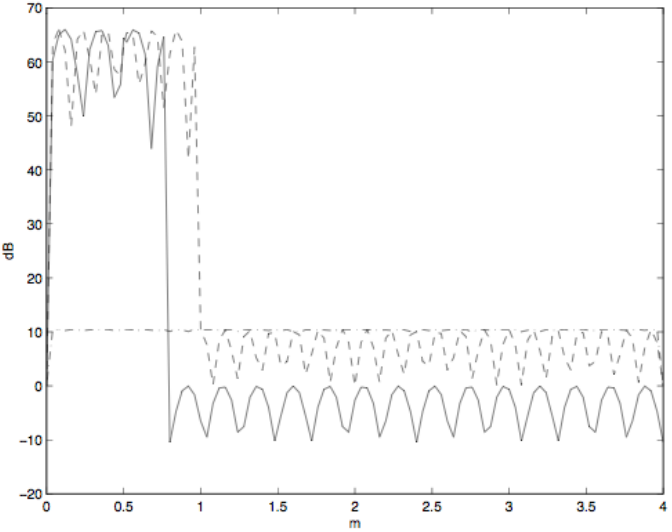
Beam



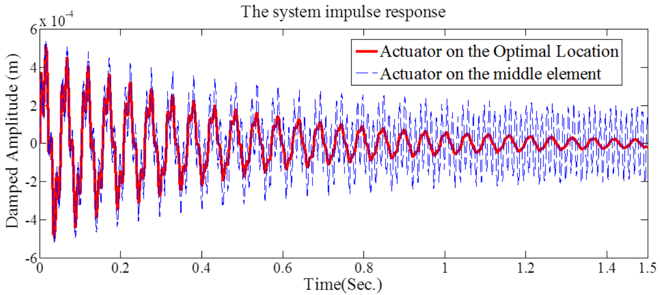
Plate

Acoustic Noise in Duct

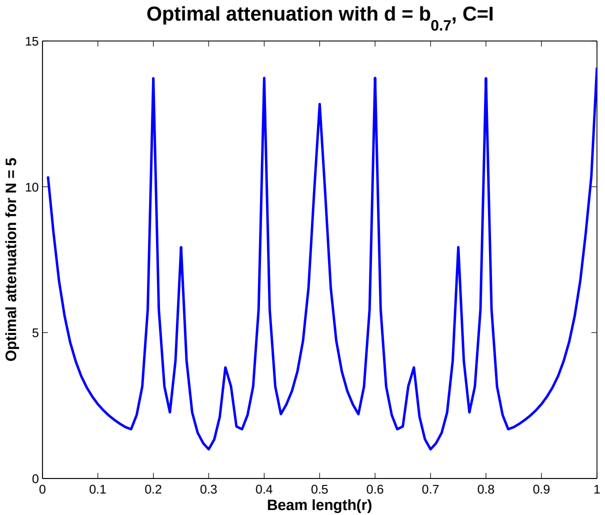
Optimal $\mathbb{H}_\infty$-performance for acoustic noise in a duct



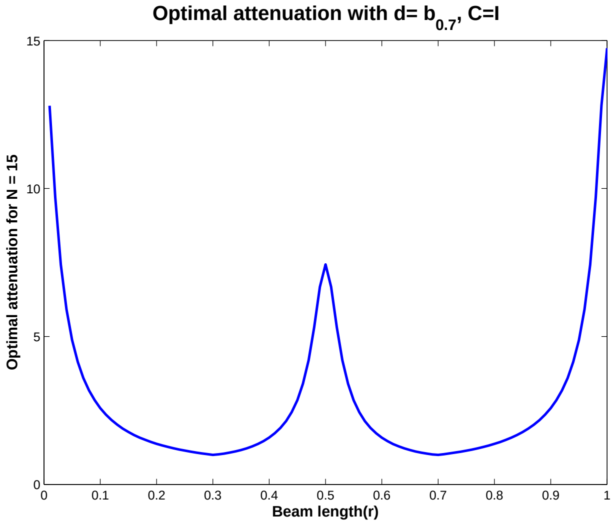
Linear-quadratic control of beam vibrations- one actuator



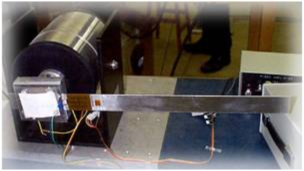
Variation of $\mathbb{H}_\infty$-optimal cost in simply supported beam - one actuator



Variation of $\mathbb{H}_\infty$-optimal cost in simply supported beam- one actuator

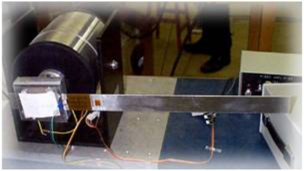


Optimal Actuator Location



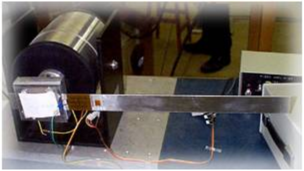
- Freedom on where to place the actuators
- Performance depends on actuator location.

Optimal Actuator Location



- Freedom on where to place the actuators
- Performance depends on actuator location.
- Different performance objectives in controller design

Optimal Actuator Location



- Freedom on where to place the actuators
- Performance depends on actuator location.

- Actuators should be located to optimize performance.

Control System Formulation

- $z(x,t)$ is temperature at point x, time t
- heat flux $u(t)$ is controlled
- $b(x)$ describes distribution of applied energy

PDE

$$\frac{\partial z}{\partial t} = \frac{\partial^2 z}{\partial x^2} + b(x)u(t), \qquad 0 < x < 1,$$

$$z(0,t) = 0, \quad z(1,t) = 0.$$

Control System Formulation

- $z(x,t)$ is temperature at point x, time t
- heat flux $u(t)$ is controlled
- $b(x)$ describes distribution of applied energy

PDE

$$\frac{\partial z}{\partial t} = \frac{\partial^2 z}{\partial x^2} + b(x)u(t), \qquad 0 < x < 1,$$

$$z(0,t) = 0, \quad z(1,t) = 0.$$

State-space Formulation

PDE

$$\frac{\partial z}{\partial t} = \frac{\partial^2 z}{\partial x^2} + b(x)u(t), \qquad 0 < x < 1,$$
$$z(0,t) = 0, \quad z(1,t) = 0.$$

$$\frac{dz}{dt} = Az(t) + Bu(t)$$

- state-space $\mathcal{H} = \mathcal{L}_2(0,1)$
- $A = \frac{\partial^2}{\partial x^2}$ with domain
  $D(A) = \{z(x) \in \mathcal{H}^2(0,1) \text{ with } z(0) = z(1) = 0\}.$
- $B = b(x)$

Strongly Continuous Semigroups

$$\dot{z}(t) = Az(t), \qquad z(0) = z_0$$

Definition: Strongly continuous semigroup $S(t)$ on Hilbert space $\mathcal{H}$

- $S(0) = I$,
- $S(t)S(s) = S(t+s)$,
- $\lim_{t\downarrow 0} S(t)z = z$, for all $z \in \mathcal{H}$.

- A generates a strongly continuous semigroup $S(t)$ on $\mathcal{H}$: For all $z \in D(A)$,

$$Az = \lim_{t\downarrow 0} \frac{S(t)z - z}{t}$$

- If $u \equiv 0$,

$$z(t) = S(t)z_0.$$

Approaches to Controller Design

Direct

- Use PDE directly to design controller.
- Controller may be infinite-dimensional
- Infinite-dimensional controller approximated for implementation

Approaches to Controller Design

Direct

- Use PDE directly to design controller.
- Controller may be infinite-dimensional
- Infinite-dimensional controller approximated for implementation

Indirect

- PDE is approximated by system of ODE's (finite-dimensional system).
- Finite-dimensional approximation is used to design controller.
- Controller needs to work on original PDE.

Approaches to Controller Design

Direct

- Use PDE directly to design controller.
- Controller may be infinite-dimensional
- Infinite-dimensional controller approximated for implementation

Indirect

- PDE is approximated by system of ODE's (finite-dimensional system).
- Finite-dimensional approximation is used to design controller.
- Controller needs to work on original PDE.

Linear-Quadratic Optimal Control

Find controller u to achieve

$$\inf_{u \in L_2(0,\infty;\mathcal{U})} \underbrace{\int_0^\infty \langle Cz(t), Cz(t)\rangle + \langle u(t), u(t)\rangle dt}_{J(u,z_0)}$$

subject to

$$\dot{z}(t) = Az(t) + Bu(t), \qquad z(0) = z_0$$

- C is design variable

Linear Quadratic (LQ) Control

$$\inf_{u \in L_2(0,\infty;\mathcal{U})} \underbrace{\int_0^\infty \langle Cz(t), Cz(t)\rangle + \langle u(t), u(t)\rangle dt}_{J(u,z_0)}$$

Definition: (A,B) is stabilizable

There exists K so that semigroup generated by $A - BK$ is exponentially stable.

Definition: (A, C) is detectable

There exists F so that semigroup generated by $A - FC$ is exponentially stable.

Linear Quadratic (LQ) Control

$$\inf_{u\in L_2(0,\infty;\mathcal{U})} \underbrace{\int_0^\infty \langle Cz(t), Cz(t)\rangle + \langle u(t), u(t)\rangle dt}_{J(u,z_0)}$$

Theorem

If (A, B) is stabilizable and (A, C) is detectable then there exists a unique $\Pi \geq 0$ such that for all $z \in D(A)$,

$$(\Pi A + A^*\Pi + C^*C - \Pi BB^*\Pi)z = 0,$$

- Optimal cost $\inf_{u\in L_2(0,\infty;\mathcal{U})} J(u, z_0) = \langle z_0, \Pi z_0\rangle$
- Optimal control $u(t) = -Kz(t)$ where $K = B^*\Pi$
- $A - BK$ generates an exponentially stable semigroup .

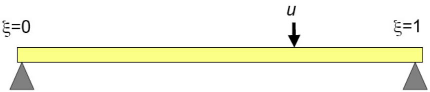
Calculation of Linear Quadratic Regulator

Operator ARE

$$A^*\Pi + \Pi A - \Pi B B^* \Pi + C^* C = 0$$

- Need to approximate solution
- Approximate A, B, C by A_n, B_n, C_n
- Let $S_n(t)$ indicate the semigroup generated by A_n.
- Approximation Π_n and hence K_n used to control original system

Example: Controller Design for Beam



PDE

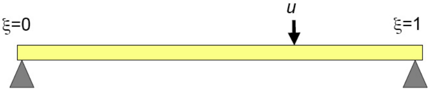
$$\frac{\partial^2 w}{\partial t^2} + \frac{\partial^4 w}{\partial x^4} = b(x)u(t), \qquad t \geq 0,\ 0 < x < 1,$$

$$b(x) = \begin{cases} 1/\delta, & |x - .5| < \frac{\delta}{2} \\ 0, & |x - .5| \geq \frac{\delta}{2} \end{cases} .$$

$$w(0,t) = 0,\ w_{xx}(0,t) = 0,\ w(1,t) = 0, w_{xx}(1,t) = 0.$$

- Use eigenfunctions as basis for approximating subspace
- Linear quadratic regulator, state weight $C = I$
- Feedback controller is $u(t) = -B^*\Pi z(t)$.

Example: Controller Design for Beam



PDE

$$\frac{\partial^2 w}{\partial t^2} + \frac{\partial^4 w}{\partial x^4} = b(x)u(t), \qquad t \geq 0,\ 0 < x < 1,$$

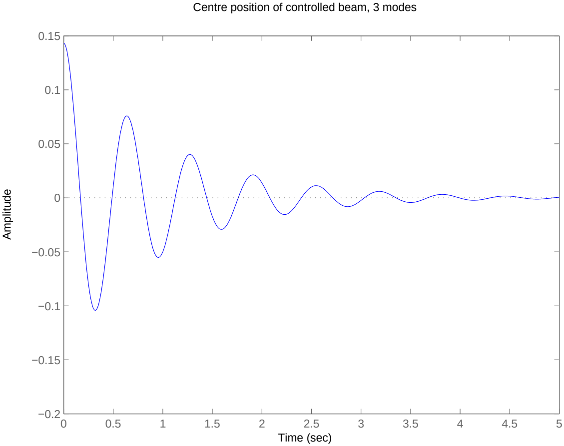
$$b(x) = \begin{cases} 1/\delta, & |x - .5| < \frac{\delta}{2} \\ 0, & |x - .5| \geq \frac{\delta}{2} \end{cases} .$$

$$w(0,t) = 0,\ w_{xx}(0,t) = 0,\ w(1,t) = 0, w_{xx}(1,t) = 0.$$

- Use eigenfunctions as basis for approximating subspace
- Linear quadratic regulator, state weight $C = I$
- Feedback controller is $u(t) = -B^*\Pi z(t)$.

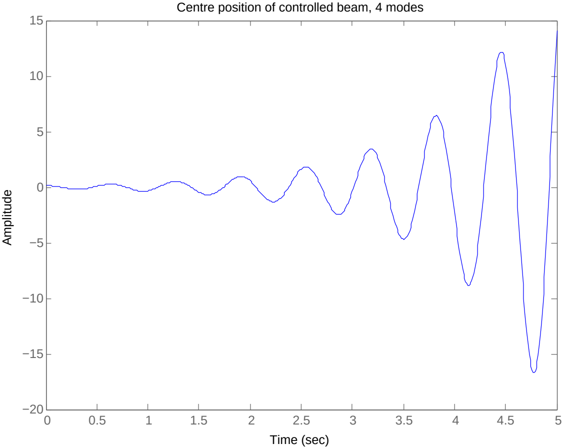
Indirect Design of Linear Quadratic Regulator for Beam

- Use first 3 modes to design controller
- Initial condition is first eigenfunction.

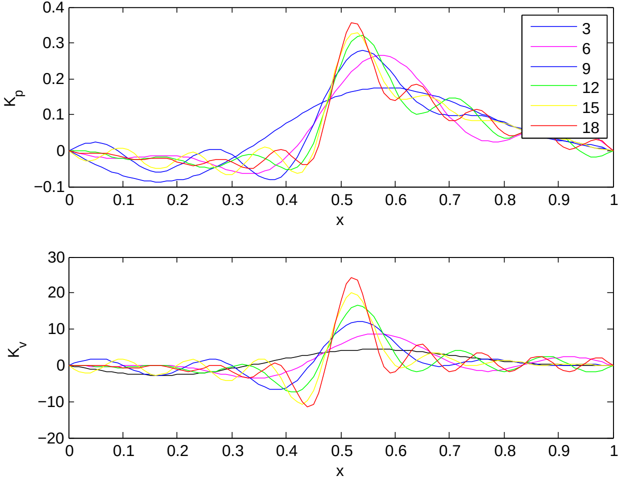


Indirect Design of Linear Quadratic Regulator for Beam

- Use first 3 modes to design controller
- Initial condition is first eigenfunction.
- Same controller, but 4 modes in approximation



Controllers for different model orders



Convergence of Π_n

Assume that for each $z \in \mathcal{H}$ $u \in \mathcal{U}$, $y \in \mathcal{Y}$,

(A1i) $\|S_n(t)P_n z - S(t)z\| \to 0$ (uniformly on $[0, T]$)

(A2i) $\|B_n u - Bu\| \to 0$, $\|C_n P_n z - Cz\| \to 0$,

Convergence of Π_n

Standard Assumptions for Controller Design

Assume that for each $z \in \mathcal{H}$ $u \in \mathcal{U}$, $y \in \mathcal{Y}$,

(A1i) $\|S_n(t)P_n z - S(t)z\| \to 0$ (uniformly on $[0, T]$)

(A1ii) $\|S_n^*(t)P_n z - S^*(t)z\| \to 0$ (same)

(A2i) $\|B_n u - Bu\| \to 0$, $\|C_n P_n z - Cz\| \to 0$,

(A2ii) $\|B_n^* z - B^* z\| \to 0$, $\|C_n^* y - C^* y\| \to 0$

Convergence of Π_n

Standard Assumptions for Controller Design

Assume that for each $z \in \mathcal{H}$ $u \in \mathcal{U}$, $y \in \mathcal{Y}$,

(A1i) $\|S_n(t)P_n z - S(t)z\| \to 0$ (uniformly on $[0, T]$)

(A1ii) $\|S_n^*(t)P_n z - S^*(t)z\| \to 0$ (same)

(A2i) $\|B_n u - Bu\| \to 0$, $\|C_n P_n z - Cz\| \to 0$,

(A2ii) $\|B_n^* z - B^* z\| \to 0$, $\|C_n^* y - C^* y\| \to 0$

(A3i) (A_n, B_n) is uniformly exponentially stabilizable:
$\exists K_n \in \mathcal{L}(\mathcal{H}_n, \mathcal{U})$, $\|K_n\| \le M$,
$\left\| e^{(A_n - B_n K_n)t} P^n z \right\| \le M_1 \, e^{-\omega_1 t} \, |z|$

(A3ii) (A_n, C_n) is uniformly exponentially detectable:
$\exists F_n \in \mathcal{L}(\mathcal{Y}, \mathcal{H}_n)$, $\|F_n\| \le M$,
$\left\| e^{(A_n - F_n C_n)t} P_n \right\| \le M_2 \, e^{-\omega_2 t}$

Convergence of Π_n

Theorem 1

If the standard assumptions for controller design hold, then for all $z \in \mathcal{H}$,

- $\|\Pi_n P_n z - \Pi z\| \to 0$
- *there exists constants $M_2 \geq 1$, $\alpha_2 > 0$, independent of n, such that*
$$\|e^{(A_n - B_n K_n)t}\| \leq M_2 e^{-\alpha_2 t}.$$

Performance Convergence

Performance arbitrarily close to optimal can be achieved:

- For sufficiently large n, semigroups generated by $A - BK_n$ are uniformly exponentially stable
- Cost with feedback K_n converges to optimal:

$$J(-K_n z(t), z_0) \longrightarrow \langle \Pi z_0, z_0 \rangle.$$

Optimal Actuator Location Problem

$$\dot{z}(t) = Az(t) + B(r)u(t), \qquad z(0) = z_0$$

- A generates a strongly continuous semigroup $S(t)$ on $\mathcal{Z}$
- Consider m actuators with locations in some closed and bounded set $\Omega \subset \mathbb{R}^N$.
- location r is a vector of length m, $r_i \in \Omega \subset \mathbb{R}^N$
- $r \in \Omega^m$
- input operator $B(r) \in \mathcal{L}(\mathcal{U}, \mathcal{Z})$.
- Choose r to minimize performance criterion
- Joint controller design/actuator location

Performance Criteria

- Different objectives lead to different controller designs
- linear-quadratic (LQ) and $\mathbb{H}_\infty$ most popular for systems with multiple inputs and outputs
- choose actuator location to optimize given design criterion
- approximations to PDE used in controller design

LQ-optimal actuator location

$$\inf_{u \in L_2(0,\infty;\mathcal{U})} \underbrace{\int_0^\infty \langle Cz(t), Cz(t)\rangle + \langle u(t), u(t)\rangle dt}_{J_r(u,z_0)}$$

$$\dot{z}(t) = Az(t) + B(r)u(t), \qquad z(0) = z_0$$

- for each r, optimal cost is $\langle \Pi(r)z_0, z_0\rangle$ where $\Pi(r)$ solves ARE.
- Choose r to minimize response to the worst initial condition:

$$\begin{aligned} \max_{\substack{z_0 \in \mathcal{H} \\ \|z_0\|=1}} \min_{u \in L_2(0,\infty;U)} J_r(u, z_o) &= \max_{\substack{z_0 \in \mathcal{H} \\ \|z_0\|=1}} \langle \Pi(r)z_0, z_0\rangle \\ &= \|\Pi(r)\| \end{aligned}$$

- Performance/Cost function for location r is $\mu(r) = \|\Pi(r)\|$

$$\hat{\mu} = \inf_{r \in \Omega^m} \|\Pi(r)\|$$

Well-posedness of LQ-optimal actuator location

Theorem 2

If

- *for any* r_0, $\lim_{r\rightarrow r_0} \|B(r)-B(r_0)\| = 0$,
- $(A, B(r))$ *are all stabilizable,* (A, C) *is detectable*
- B *is a compact operator,*

then

$$\lim_{r\rightarrow r_0} \|\Pi(r)-\Pi(r_0)\| = 0.$$

Also, there exists $\hat{r}$ *such that*

$$\|\Pi(\hat{r})\| = \inf_{r\in\Omega^m} \|\Pi(r)\| = \hat{\mu}.$$

Well-posedness of LQ-optimal actuator location

Theorem 2

If

*

- *for any* r_0, $\lim_{r\to r_0}\|B(r)-B(r_0)\|=0$,
- $(A, B(r))$ *are all stabilizable,* (A, C) *is detectable*
- B *is a compact operator,*

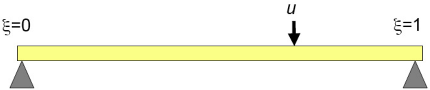
then

$$\lim_{r\to r_0}\|\Pi(r)-\Pi(r_0)\|=0.$$

Also, there exists $\hat{r}$ *such that*

$$\|\Pi(\hat{r})\| = \inf_{r\in\Omega^m}\|\Pi(r)\| = \hat{\mu}.$$

Optimal actuator location for simply supported beam with viscous damping



PDE

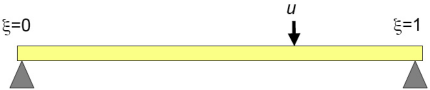
$$\frac{\partial^2 w}{\partial t^2} + c_d \frac{\partial w}{\partial t} + \frac{\partial^4 w}{\partial x^4} = b(x)u(t), \qquad t \geq 0, 0 < x < 1,$$

$$w(0,t) = 0,\ w_{xx}(0,t) = 0,\ w(1,t) = 0, w_{xx}(1,t) = 0.$$

$$b(r) = \begin{cases} 1/\delta, & |x-r| < \frac{\delta}{2} \\ 0, & |x-r| \geq \frac{\delta}{2} \end{cases}.$$

- reduce state uniformly: $C = I$
- eigenfunction approximations

Optimal actuator location for simply supported beam with viscous damping



PDE

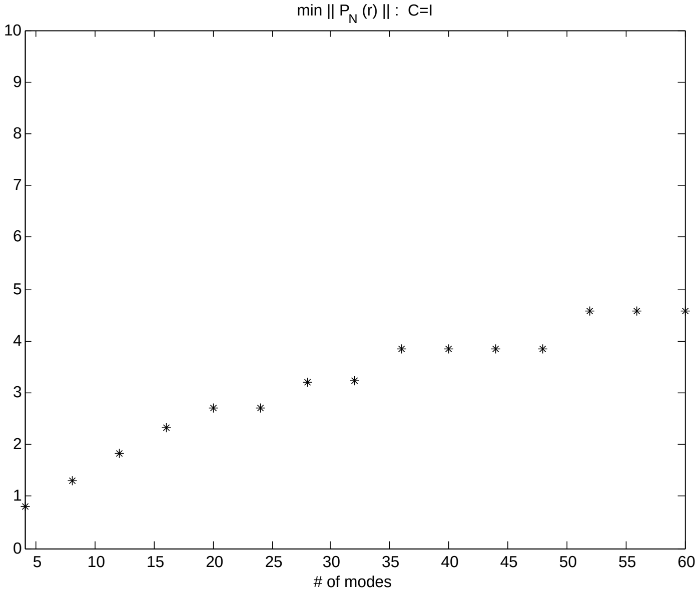
$$\frac{\partial^2 w}{\partial t^2} + c_d \frac{\partial w}{\partial t} + \frac{\partial^4 w}{\partial x^4} = b(x)u(t), \qquad t \geq 0, 0 < x < 1,$$

$$w(0,t) = 0,\ w_{xx}(0,t) = 0,\ w(1,t) = 0, w_{xx}(1,t) = 0.$$

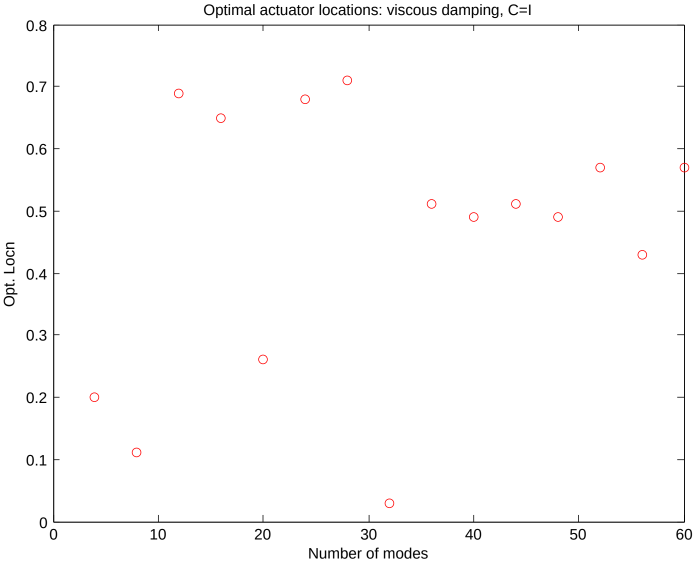
$$b(r) = \begin{cases} 1/\delta, & |x - r| < \frac{\delta}{2} \\ 0, & |x - r| \geq \frac{\delta}{2} \end{cases} .$$

- reduce state uniformly: $C = I$
- eigenfunction approximations

Optimal performance ($\|\Pi_n\|$), $C = I$



Optimal actuator location ($\|\Pi_n\|$), $C = I$



Convergence of $LQ-$Optimal Actuator Location

Theorem 3

In addition to (*) assume that

- $(A_n, B_n(r), C_n)$ *satisfies standard assumptions on approximations for controller design*
- C *is a compact operator.*

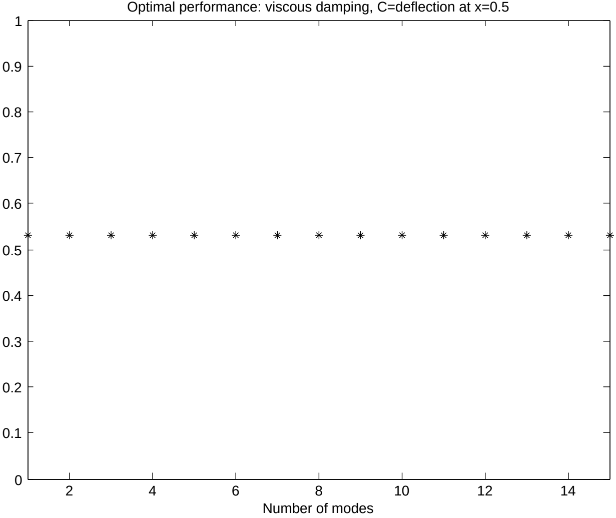
Then

$$\hat{\mu} = \lim_{n\to\infty} \hat{\mu}_n,$$

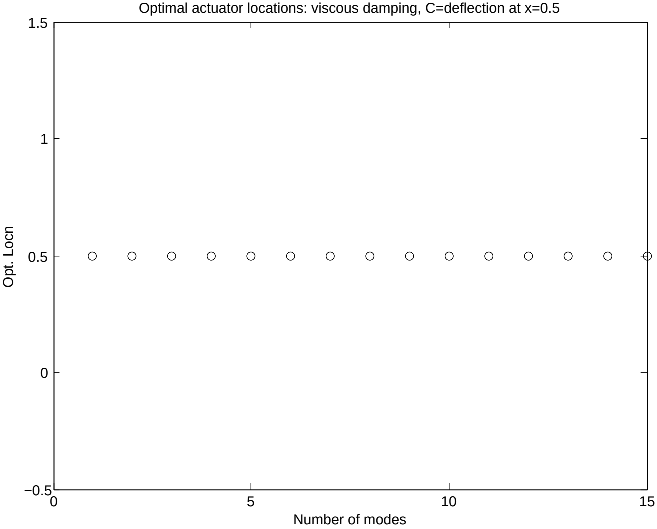
and there exists a subsequence $\{\hat{r}_m\}$ *of* $\{\hat{r}_n\}$ *such that*

$$\hat{\mu} = \lim_{m\to\infty} \|\Pi(\hat{r}_m)\|.$$

Optimal performance $\|\Pi_n\|$, state weight is $C = [I \ 0]$



Optimal actuator location, state weight is $C = [I\ 0]$



Algorithm for LQ-Optimal Actuator Location

- discretize region into M possible actuator locations
- replace physical locations by $\bar{r} = (0, 1, 0..)$
- $\#$ 1's $= \#$ actuators (m)
- $\bar{r} \in \Phi = \{\bar{r} \in \mathbb{R}^M,\ \bar{r}_j \in \{0, 1\},\ \sum_{j=1}^{M} \bar{r}_j = m\}$.
- convex in $\mathbb{R}^M$ [Geromel, 1989]

$$\mu(\bar{r}) \geq \mu(\bar{r}^0) + \langle d(\bar{r}^0), \bar{r} - \bar{r}^0 \rangle$$

where d is a subgradient.

LQ-optimal actuator location algorithm

$\mu(\bar{r})$ is replaced with θ and the optimization problem relaxed to

$$\begin{aligned} &\min\nolimits_{\bar{r},\theta} \theta \\ s.t. \quad &\theta \geq \mu(\bar{r}^0) + \langle d(\bar{r}^0), \bar{r} - \bar{r}^0 \rangle \end{aligned}$$

- Since this is a linear optimization problem the solution falls on the boundary of the inequality constraint and

$$\theta^1 = \min_{\bar{r}} \mu(\bar{r}^0) + \langle d(\bar{r}^0), \bar{r} - \bar{r}^0 \rangle.$$

- If the solution of this problem, $\bar{r}^1$, has $\theta = \mu(\bar{r}^1)$ then by convexity, $\bar{r}^1$ is a minimizer
- Otherwise, introduce another constraint

$$\begin{aligned} &\min\nolimits_{\bar{r}} \theta \\ s.t. \quad &\theta \geq \mu(\bar{r}^i) + \langle d(\bar{r}^i), \bar{r} - \bar{r}^i \rangle \quad i = 0, 1 \end{aligned}$$

- and so on

A Set $k = 0$ and choose tolerance $\varepsilon > 0$.

B Choose an initial location for actuators $\bar{r}^0 \in \Phi$ and calculate $\mu(\bar{r}^0)$ and $d(\bar{r}^0)$.

C

$$\min_{\theta, \bar{r}} \theta$$
$$s.t. \quad \theta \geq \mu(\bar{r}^i) + \langle d(\bar{r}^i), \bar{r} - \bar{r}^i \rangle \quad i = 1, \ldots, k$$

using a branch and bound algorithm

D Calculate $\mu(\bar{r}^{k+1})$. If $\mu(\bar{r}^{k+1}) - \theta^{k+1} \leq \varepsilon$, done. If not; return to step C.

Comparision to Genetic Algorithm

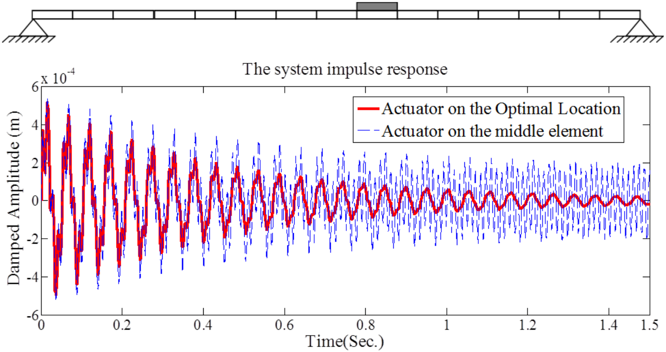
	Objective Value	Elapsed time(sec.)
Current Method	71.98	4.78e2
GA	72.17	4.14e4

Optimal location of 10 actuators, pinned beam

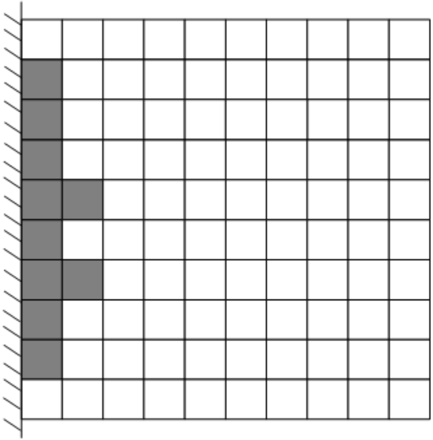
	Objective Value	Elapsed time(sec.)
Current Method	1.584	4.920e2
Genetic algorithm	1.748	4.443e4

Optimal location of 10 actuators, cantilevered plate

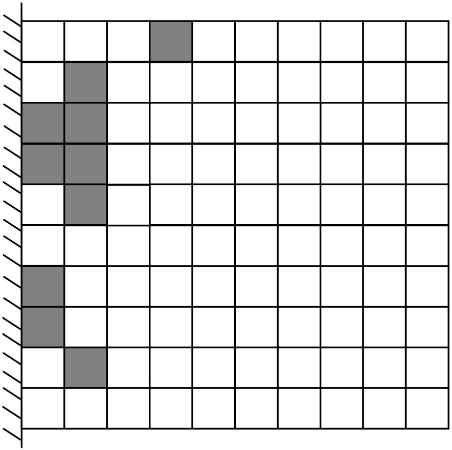
Optimal location for single actuator on a pinned beam



Optimal location for 10 actuators on a cantilevered plate



Genetic Algorithm- cantilevered plate



$\mathbb{H}_\infty$- Controller Design

$$\dot{z}(t) = Az(t) + B(r)u(t) + Dd(t), \quad z(0) = 0$$

$\mathbb{H}_\infty$- Controller Design

$$\dot{z}(t) = Az(t) + B(r)u(t) + Dd(t), \quad z(0) = 0$$

$\mathbb{H}_\infty$- Controller Design

$$\dot{z}(t) = Az(t) + B(r)u(t) + Dd(t), \quad z(0) = 0$$

- A generates a strongly continuous semigroup $S(t)$ on $\mathcal{Z}$
- $B(r) \in \mathcal{L}(\mathcal{U}, \mathcal{Z})$, $D \in \mathcal{L}(\mathcal{V}, \mathcal{Z})$
- full information: input to the controller is

$$y_2(t) = \begin{bmatrix} z(t) \\ d(t) \end{bmatrix}$$

Problem Formulation

Cost : $y_1(t) = \begin{bmatrix} Cz(t) \\ u(t) \end{bmatrix}$

Find, for given $\gamma > 0$, a stabilizing controller so that

$$\int_0^\infty \|y_1(t)\|^2 dt < \gamma^2 \int_0^\infty \|d(t)\|^2 dt$$

The system is stabilizable with attenuation $\gamma(r)$ if there is a stabilizing controller so that this inequality holds.

- equivalent to the $\mathbb{H}_\infty$-norm of the transfer function being less than γ

Solution to fixed attenuation problem

The full-information system is stabilizable with disturbance attenuation $\gamma(r)$ if and only if there exists a nonnegative, self-adjoint operator Π on $\mathcal{Z}$ solving the algebraic Riccati equation (ARE)

$$\left(A^*\Pi+\Pi A-\Pi\left(B(r)B(r)^*-\frac{1}{\gamma^2}DD^*\right)\Pi+C^*C\right)z=0, \qquad z\in D(A), \tag{1}$$

where $A-B(r)B(r)^*\Pi+\frac{1}{\gamma^2}DD^*\Pi$ generates an exponentially stable semigroup on $\mathcal{Z}$.

$$u(t)=-\underbrace{B(r)^*\Pi}_{K}z(t)$$

achieves $\gamma(r)$-attenuation.

Solution to fixed attenuation problem

The full-information system is stabilizable with disturbance attenuation $\gamma(r)$ if and only if there exists a nonnegative, self-adjoint operator Π on $\mathcal{Z}$ solving the algebraic Riccati equation (ARE)

$$\left(A^*\Pi + \Pi A - \Pi\left(B(r)B(r)^* - \frac{1}{\gamma^2}DD^*\right)\Pi + C^*C\right)z = 0, \qquad z \in D(A), \tag{1}$$

where $A - B(r)B(r)^*\Pi + \frac{1}{\gamma^2}DD^*\Pi$ generates an exponentially stable semigroup on $\mathcal{Z}$.

$$u(t) = -B(r)^*\Pi$$

achieves $\gamma(r)$-attenuation.

$\mathbb{H}_\infty$-optimal actuator location

- m actuators
- Control operator B varies with r, $r \in \Omega^m$.
- optimal attenuation with actuator location r is

$$\mu(r) = \inf \gamma(r)$$

  over all $\gamma(r)$ for which the system is stabilizable with attenuation $\gamma(r)$.
- indicate optimal attenuation over all r by $\hat{\mu}$.

Well-posedness of $\mathbb{H}_\infty$-optimal actuator problem

Theorem 4

Consider a family of control systems with full information. If

*

- *for any* r_0, $\lim_{r \to r_0} \|B(r) - B(r_0)\| = 0$,
- $(A, B(r))$ *are all stabilizable,* (A, C) *is detectable*
- B *and* D *are compact operators,*

then

$$\lim_{r \to r_0} \mu(r) = \mu(r_0). \tag{2}$$

Furthermore, there exists an optimal actuator location $\hat{r}$ *so that*

$$\hat{\mu} = \mu(\hat{r}) = \inf_{r \in \Omega^m} \mu(r).$$

Proof (outline)

- First show that if system at r_0 is stabilizable with attenuation $\gamma(r_0)$ then for every $\epsilon > 0$, for small $\|r - r_0\|$, $(A, [B(r)\ D], C)$ stabilizable with attenuation $\gamma(r_0) + \epsilon$.
- Also show can choose stabilizing feedback operators $K(r)$ to be continuous at r_0.
- Regard systems at r as approximations/perturbations to system at r_0.
- Show $\lim_{r \to r_0} \mu(r) = \mu(r_0)$.
- existence of an optimal actuator location then follows from compactness of Ω^M.

Convergence of $\mathbb{H}_\infty$-optimal actuator location using approximations

Theorem 5

In addition to (*) assume that

- $(A_n, [B_n(r)\ D_n], C_n)$ *where* $B_n = P_n B$ *satisfies standard assumptions on approximations for controller design.*

Letting $\hat{r}$ *be an optimal actuator location for* $(A, [B(r)\ D], C)$ *with optimal cost* $\hat{\mu}$ *and defining similarly* $\hat{r}_n$, $\hat{\mu}_n$, *it follows that*

$$\hat{\mu} = \lim_{n\to\infty} \hat{\mu}_n,$$

and there exists a subsequence $\{\hat{r}_m\}$ *of* $\{\hat{r}_n\}$ *such that*

$$\hat{\mu} = \lim_{m\to\infty} \mu(\hat{r}_m).$$

Convergence of $\mathbb{H}_\infty$-optimal actuator location using approximations

Theorem 5

In addition to (*) assume that

- $(A_n, [B_n(r)\ D_n], C_n)$ *where* $B_n = P_n B$ *satisfies standard assumptions on approximations for controller design.*

Letting $\hat{r}$ *be an optimal actuator location for* $(A, [B(r)\ D], C)$ *with optimal cost* $\hat{\mu}$ *and defining similarly* $\hat{r}_n$, $\hat{\mu}_n$, *it follows that*

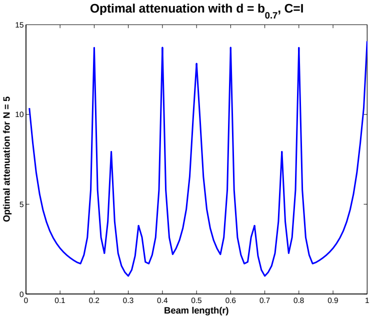
$$\hat{\mu} = \lim_{n\to\infty} \hat{\mu}_n,$$

and there exists a subsequence $\{\hat{r}_m\}$ *of* $\{\hat{r}_n\}$ *such that*

$$\hat{\mu} = \lim_{m\to\infty} \mu(\hat{r}_m).$$

C doesn't need to be compact.

Calculation of optimal attenuation



$\mathbb{H}_\infty$-cost function with respect to actuator location for simply supported beam with viscous damping

Calculation of optimal attenuation

- $\mathbb{H}_\infty$ performance $\mu(r)$ is nonconvex
- $\mu(r)$ likely not differentiable
- construction of derivative of $\mu(r)$?
- function $\mu(r)$ evaluation is time-consuming

Calculation of optimal attenuation

- $\mathbb{H}_\infty$ performance $\mu(r)$ is nonconvex
- $\mu(r)$ likely not differentiable
- construction of derivative of $\mu(r)$?
- function $\mu(r)$ evaluation is time-consuming

- use derivative-free optimization
- directional direct search algorithm
- parallel implementation

Calculation of optimal attenuation

- $\mathbb{H}_\infty$ performance $\mu(r)$ is nonconvex
- $\mu(r)$ likely not differentiable
- construction of derivative of $\mu(r)$?
- function $\mu(r)$ evaluation is time-consuming

- use derivative-free optimization
- directional direct search algorithm
- parallel implementation

Algorithm

Current iterate: r_k, α_k, $\mu(r_k)$, set of positive bases $\mathcal{D}$

1. Compute r_{k+1} such that $\mu(r_{k+1}) < \mu(r_k)$
   1. **Search step:** Evaluate μ at a finite number of (random) points If a better point r is found, set $r_{k+1} := r$; iteration successful and skip poll step
   2. **Poll step:** Choose (randomly) $\mathcal{D}_k \in \mathcal{D}$. Order the poll set $P_k := \{r_k + \alpha_k d : d \in \mathcal{D}_k\}$. Start evaluating μ at the poll points. If a better poll point is found, set $r_{k+1} := r_k + \alpha_k d_k$; iteration successful. Otherwise, set $r_{k+1} := r_k$ and declare iteration unsuccessful.

Algorithm

Current iterate: r_k, α_k, $\mu(r_k)$, set of positive bases $\mathcal{D}$

1. Compute r_{k+1} such that $\mu(r_{k+1}) < \mu(r_k)$
   1. **Search step:** Evaluate μ at a finite number of (random) points If a better point r is found, set $r_{k+1} := r$; iteration successful and skip poll step
   2. **Poll step:** Choose (randomly) $\mathcal{D}_k \in \mathcal{D}$. Order the poll set $P_k := \{r_k + \alpha_k d : d \in \mathcal{D}_k\}$. Start evaluating μ at the poll points. If a better poll point is found, set $r_{k+1} := r_k + \alpha_k d_k$; iteration successful. Otherwise, set $r_{k+1} := r_k$ and declare iteration unsuccessful.

Algorithm

Current iterate: r_k, α_k, $\mu(r_k)$, set of positive bases $\mathcal{D}$

1. Compute r_{k+1} such that $\mu(r_{k+1}) < \mu(r_k)$
    1. **Search step:** Evaluate μ at a finite number of (random) points If a better point r is found, set $r_{k+1} := r$; iteration successful and skip poll step
    2. **Poll step:** Choose (randomly) $\mathcal{D}_k \in \mathcal{D}$. Order the poll set $P_k := \{r_k + \alpha_k d : d \in \mathcal{D}_k\}$. Start evaluating μ at the poll points. If a better poll point is found, set $r_{k+1} := r_k + \alpha_k d_k$; iteration successful. Otherwise, set $r_{k+1} := r_k$ and declare iteration unsuccessful.

Algorithm

Current iterate: r_k, α_k, $\mu(r_k)$, set of positive bases $\mathcal{D}$

1. Compute r_{k+1} such that $\mu(r_{k+1}) < \mu(r_k)$
   1. **Search step:** Evaluate μ at a finite number of (random) points If a better point r is found, set $r_{k+1} := r$; iteration successful and skip poll step
   2. **Poll step:** Choose (randomly) $\mathcal{D}_k \in \mathcal{D}$. Order the poll set $P_k := \{r_k + \alpha_k d : d \in \mathcal{D}_k\}$. Start evaluating μ at the poll points. If a better poll point is found, set $r_{k+1} := r_k + \alpha_k d_k$; iteration successful. Otherwise, set $r_{k+1} := r_k$ and declare iteration unsuccessful.

Algorithm

Current iterate: r_k, α_k, $\mu(r_k)$, set of positive bases $\mathcal{D}$

1. Compute r_{k+1} such that $\mu(r_{k+1}) < \mu(r_k)$
   1. **Search step:** Evaluate μ at a finite number of (random) points If a better point r is found, set $r_{k+1} := r$; iteration successful and skip poll step
   2. **Poll step:** Choose (randomly) $\mathcal{D}_k \in \mathcal{D}$. Order the poll set $P_k := \{r_k + \alpha_k d : d \in \mathcal{D}_k\}$. Start evaluating μ at the poll points. If a better poll point is found, set $r_{k+1} := r_k + \alpha_k d_k$; iteration successful. Otherwise, set $r_{k+1} := r_k$ and declare iteration unsuccessful.
2. **Step size update:** If iteration is successful, then $\alpha_{k+1} := \alpha_k$. Otherwise, decrease the step size parameter, *e.g.* $\alpha_{k+1} := \alpha_k/2$.

Algorithm

Current iterate: r_k, α_k, $\mu(r_k)$, set of positive bases $\mathcal{D}$

1. Compute r_{k+1} such that $\mu(r_{k+1}) < \mu(r_k)$
    1. **Search step:** Evaluate μ at a finite number of (random) points If a better point r is found, set $r_{k+1} := r$; iteration successful and skip poll step
    2. **Poll step:** Choose (randomly) $\mathcal{D}_k \in \mathcal{D}$. Order the poll set $P_k := \{r_k + \alpha_k d : d \in \mathcal{D}_k\}$. Start evaluating μ at the poll points. If a better poll point is found, set $r_{k+1} := r_k + \alpha_k d_k$; iteration successful. Otherwise, set $r_{k+1} := r_k$ and declare iteration unsuccessful.
2. **Step size update:** If iteration is successful, then $\alpha_{k+1} := \alpha_k$. Otherwise, decrease the step size parameter, *e.g.* $\alpha_{k+1} := \alpha_k/2$.
3. Repeat the above steps till convergence.

Surrogate Model

- Function evaluations $\mu(r)$ are expensive
- Use a surrogate model $sm(\cdot)$ to replace evaluations of $\mu(\cdot)$.
- $sm(\cdot)$ is less accurate but cheaper to evaluate than f.
- Order the points by evaluating $sm(\cdot)$ and evaluate $f(\cdot)$ in this order.
- useful if several evaluations of surrogate model cheaper than one evaluation of f.

Surrogate Model

- Function evaluations $\mu(r)$ are expensive
- Use a surrogate model $sm(\cdot)$ to replace evaluations of $\mu(\cdot)$.
- $sm(\cdot)$ is less accurate but cheaper to evaluate than f.
- Order the points by evaluating $sm(\cdot)$ and evaluate $f(\cdot)$ in this order.
- useful if several evaluations of surrogate model cheaper than one evaluation of f.

Surrogate Model

- Function evaluations $\mu(r)$ are expensive
- Use a surrogate model $sm(\cdot)$ to replace evaluations of $\mu(\cdot)$.
- $sm(\cdot)$ is less accurate but cheaper to evaluate than f.
- Order the points by evaluating $sm(\cdot)$ and evaluate $f(\cdot)$ in this order.
- useful if several evaluations of surrogate model cheaper than one evaluation of f.

Surrogate Model

- Function evaluations $\mu(r)$ are expensive
- Use a surrogate model $sm(\cdot)$ to replace evaluations of $\mu(\cdot)$.
- $sm(\cdot)$ is less accurate but cheaper to evaluate than f.
- Order the points by evaluating $sm(\cdot)$ and evaluate $f(\cdot)$ in this order.
- useful if several evaluations of surrogate model cheaper than one evaluation of f.

Surrogate Model

- Each calculation of $\mu(\cdot)$ requires multiple solutions of a (possibly large-scale) Riccati equation

Surrogate Model

- Each calculation of $\mu(\cdot)$ requires multiple solutions of a (possibly large-scale) Riccati equation
- For an arbitrary attenuation γ, solve the Riccati equation for Π to obtain a controller

$$u(t) = -B(r)^*\Pi$$

  and calculate the actual attenuation $\tilde{\gamma}(r)$ in controlled system
- Calculation relatively cheap requiring check on imaginary eigenvalues of an associated matrix.

Surrogate Model

- Each calculation of $\mu(\cdot)$ requires multiple solutions of a (possibly large-scale) Riccati equation
- For an arbitrary attenuation γ, solve the Riccati equation for Π to obtain a controller

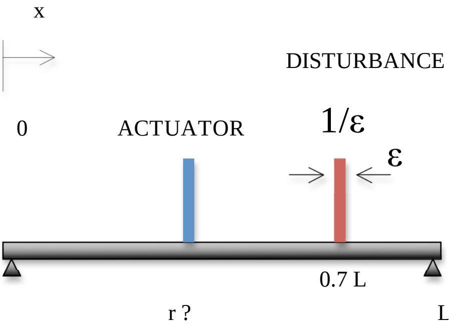
$$u(t) = -B(r)^*\Pi$$

  and calculate the actual attenuation $\tilde{\gamma}(r)$ in controlled system
- Calculation relatively cheap requiring check on imaginary eigenvalues of an associated matrix.
- Actual attenuation is close to the optimal attenuation:

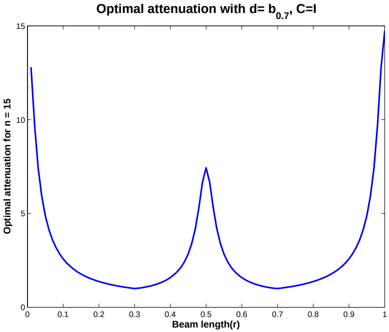
$$\mu(r) \approx \tilde{\gamma}(r)$$

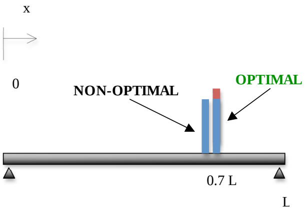
- $sm(r) = \tilde{\gamma}(r)$.

Simply supported beam with Kelvin-Voigt damping



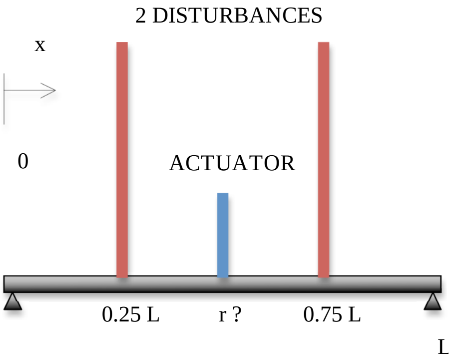
Simply supported beam with Kelvin-Voigt damping





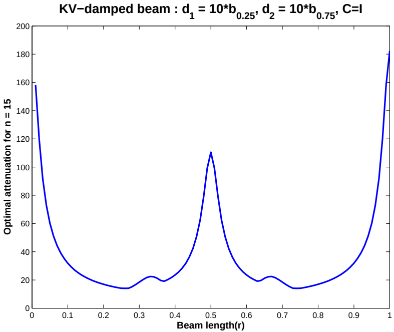
OPTIMAL COST: 0.99 NON-OPTIMAL COST: 1.37

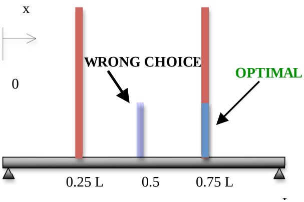
Beam with 2 disturbances and 1 actuator



GUESS: CENTER!!

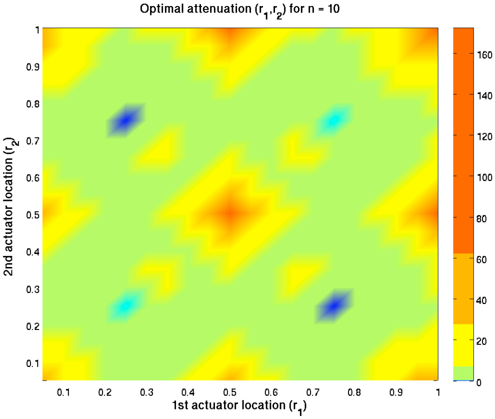
Beam with 2 disturbances and 1 actuator



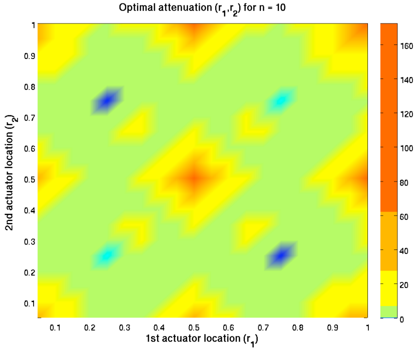


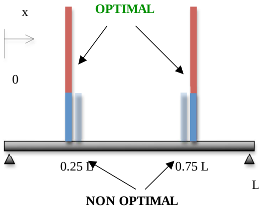
PERFORMANCE: 118 15
EVALUATIONS OF f(.) : 10
EVALUATIONS OF sm(.) : 100

Beam with 2 disturbances and 2 actuators



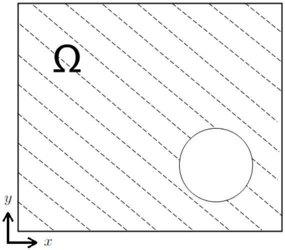
Beam with 2 Point disturbances and 2 actuators

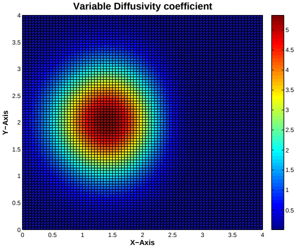




OPTIMAL COST: 9.9 NON-OPTIMAL COST: 16.0

Diffusion





Diffusivity $\kappa(x_1, x_2)$

Diffusion

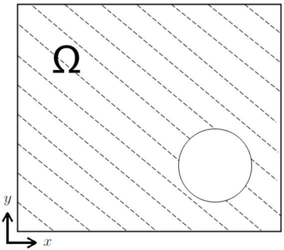
$$\begin{aligned}\frac{\partial z}{\partial t}(x_1, x_2, t) &= \nabla \cdot (\kappa(x_1, x_2)\nabla z(x_1, x_2, t)) + b(x_1, x_2)u(t) + v(t),\\ z(x_1, x_2, \cdot) &= 0 \text{ on } \partial\Omega,\\ y(t) &= \int\int_{\Omega} z(x_1 x_2, t)\ dx,\end{aligned} \qquad (3)$$

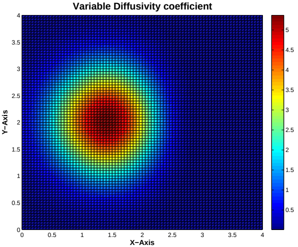
where

$$b(x_1, x_2) = \begin{cases} \frac{1}{\epsilon}, & (x_1, x_2) \in \square(r_1, r_2), \\ 0, & \text{otherwise,} \end{cases} ,$$

with $\square(r_1, r_2, \epsilon)$ is a square centered at (r_1, r_2) and side $\epsilon = 0.2$.

Diffusion





Diffusivity $\kappa(x_1, x_2)$

Optimal Actuator location is $(3.1, 3.35)$

Algorithm Performance

Property	**Coarse**	**Fine**
Order	200	700
# iterations	15	4
Overall time	1h 10m	5h 50 m
# $\hat{\gamma}(\cdot)$ evaluations	10	1
# $sm(\cdot)$ evaluations	208	64

Summary

- Developed algorithms for optimal actuator placement using LQ and $\mathbb{H}_\infty$-cost criteria.
- Original problem needs to be properly formulated
- For optimal actuator location, compactness is important for well-posedness of problem and for convergence of approximations.
- Strong degradation of performance if actuators not placed optimally

QUESTIONS?