Robust Control Workshop
Delft Center for Systems and Control

Mixed Deterministic/Randomized Methods for Fixed Order Controller Design

Roberto Tempo
IEIIT-CNR
Politecnico di Torino
tempo@polito.it
Acknowledgments

- Joint work with Yasumasa Fujisaki and Yasuaki Oishi
- Thanks to Shinji Hara and Toshiharu Sugie for technical discussions and for hosting me in their respective institutions
Computationally difficult problems

- Various robust control problems are computationally intractable
- This is generally meant for NP-hard problems
- Examples:
  - stability of interval matrices
  - μ computation
  - static output feedback
Deterministic methods can be successfully used only for computationally tractable problems

Example: Convex problems

Many (difficult) control problems are not convex

Finding local solutions instead of global solutions

Use of relaxation, overbounding or other techniques
Randomized methods

- Randomized methods are often used in computer science, computational geometry, optimization, …
- Problems: Data structuring and search trees, graph algorithms, linear program, …
- Remarkably, randomized methods are not used systematically in uncertain systems and control
- Development of mathematically rigorous methods, not straightforward use of Monte Carlo simulations
Randomized algorithms (RAs)

- Randomized algorithms are *probably approximately correct* (PAC)
- They provide an approximate solution only with some given probability
- Give up a guaranteed deterministic solution
- Obtain polynomial-time complexity
Early work on the topic

- First appearance in [Stengel, 80]
- Motivations: Flight control (applications) and stochastic optimal control (theory)
- CDC paper titled “Probabilistic robust controller design” by [Djavadan, Tulleken, Voetter, Verbruggen, Olsder, 89]
- Drawback: Not much theory, development of Monte Carlo methods for analysis problems
Subsequent work on the topic

- Finite sample size bounds given in [Khargonekar and Tikku, 96; Tempo, Bai and Dabbene, 96]
- Statistical learning theory for control design [Vidyasagar, 98]
Deterministic and randomized methods

deterministic methods

randomized methods
Mixed deterministic/randomized methods
Tractable/intractable parameters

Controller

- Controller parameters $\theta$ and $\eta$
- Tractable parameters $\theta$
- (Computationally) intractable parameters $\eta$
- *Deterministic* methods for $\theta$
- *Randomized* methods for $\eta
Plant and controller

- SISO strictly proper plant

\[ P(s) = \frac{N_P(s)}{D_P(s)} \]

- Fixed order controller

\[ C(s) = \frac{N_C(s)}{D_C(s)} = \frac{X(s^2) + sY(s^2)}{Z(s^2) + sV(s^2)} \]
Controller polynomials

- The even-order controller polynomials are of the form
  \[ X(s^2) = \theta_0 + \theta_2 s^2 + \theta_4 s^4 + \ldots \]
  \[ Y(s^2) = \alpha_0 + \alpha_2 s^2 + \alpha_4 s^4 + \ldots \]
  \[ Z(s^2) = \beta_0 + \beta_2 s^2 + \beta_4 s^4 + \ldots \]
  \[ V(s^2) = \mu_0 + \mu_2 s^2 + \mu_4 s^4 + \ldots \]

- Closed-loop polynomial is given by
  \[ p(s) = N_P(s) \left( X(s^2) + s Y(s^2) \right) + D_P(s) \left( Z(s^2) + s V(s^2) \right) \]

- We assume that \( p(s) \) has invariant degree (generic subset of controller parameters)
Parameter partitioning

1. Coefficients of $X(s^2)$ are tractable; coefficients of $Y(s^2)$, $Z(s^2)$, $V(s^2)$ are computationally intractable

2. Coefficients of $Y(s^2)$ are tractable; coefficients of $X(s^2)$, $Z(s^2)$, $V(s^2)$ are computationally intractable

3. Coefficients of $Z(s^2)$ are tractable; coefficients of $X(s^2)$, $Y(s^2)$, $V(s^2)$ are computationally intractable

4. Coefficients of $V(s^2)$ are tractable; coefficients of $X(s^2)$, $Y(s^2)$, $Z(s^2)$ are computationally intractable
Tractable/intractable parameters

- We consider the first case
- Then, we have
  \[ \theta = [\theta_0 \ \theta_2 \ \theta_4 ...]^T \quad \text{tractable parameters} \]
  \[ \eta = [\alpha_0 \ \alpha_2 \ \alpha_4 ... \ \beta_0 \ \beta_2 \ \beta_4 ... \ \mu_0 \ \mu_2 \ \mu_4 ...]^T \quad \text{intractable parameters} \]

- We assume that \( \theta \in \mathcal{M} \) and \( \eta \in \mathcal{N} \)
  \[ n_{\theta} = \text{deg}(X)/2 + 1 \]
  \[ n_{\eta} = \text{deg}(Y)/2 + \text{deg}(V)/2 + \text{deg}(Z)/2 + 3 \]
Example: Second order controller

\[ N_C(s) = X(s^2) + sY(s^2) = \theta_0 + \alpha_0 s + \theta_2 s^2 \]
\[ D_C(s) = Z(s^2) + sV(s^2) = \beta_0 + \mu_0 s + \beta_2 s^2 \]

\[ \theta = [\theta_0 \ \theta_2]^T \quad \text{tractable parameters} \]
\[ \eta = [\alpha_0 \ \beta_0 \ \beta_2 \ \mu_0]^T \quad \text{intractable parameters} \]

\[ n_\theta = 2 \]
\[ n_\eta = 4 \]
Randomized Methods for Intractable Parameters
RAs for intractable parameters

**Definition** (feasibility)

- Intractable parameter $\eta \in \mathcal{N}$ is feasible if the set of stabilizing controllers is not empty.

- Take a probability measure $\mathcal{P}$ associated to $\mathcal{N}$.

- Let $\varepsilon \in (0,1)$ and $\delta \in (0,1)$ be confidence and accuracy and consider:

$$N_I = \frac{\log(\delta)}{\log(1-\varepsilon)}$$
Feasibility algorithm

Algorithm 1

1. for $i := 1, \ldots, N_f$ do
   begin
   2. draw a sample $\eta^{(i)} \in \mathcal{N}$ according to $\mathcal{P}$
   3. if $\eta^{(i)}$ is feasible then return
   end
Probability of feasibility

- Let $\mathcal{A}$ be the set of all feasible intractable parameters $\eta$ and $P(\mathcal{A})$ its measure.
- Suppose that $P(\mathcal{A}) > \epsilon$.
- **Theorem**
  The probability that no $\eta^{(i)}$ given by Algorithm 1 is feasible is less than $\delta$.
- **Proof** follows from the “log-over-log” bound.
Theorem says that Algorithm 1 gives a feasible $\eta^{(i)}$ with confidence higher than $1 - \delta$

Sample size $N_1$ depends only on confidence and accuracy

Algorithm 1 is polynomial-time because sample generation (step 2) and feasibility check (step 3) are polynomial-time

Bound $N_1$ is given in [Khargonekar and Tikku, 96; Tempo, Bai and Dabbene, 96]

Specific instance of fpras theory, see [Tempo, Calafiore and Dabbene, 05]
RAs for intractable parameters

- Let $\varepsilon \in (0,1)$ and $\delta \in (0,1)$ and consider
  \[
  N_2 = \frac{1}{2\varepsilon^2} \log \frac{2}{\delta}
  \]

- This is the well-known Chernoff bound [Chernoff, 52]

- Bound $N_2$ is independent of the number of intractable parameters, and depends only on accuracy and confidence
Algorithm for measure computation

**Algorithm 2**

1. set $N_s := 0$
2. for $i := 1, \ldots, N_2$ do
   begin
   3. draw a sample $\eta^{(i)} \in \mathcal{N}$ according to $\mathcal{P}$
   4. if $\eta^{(i)}$ is feasible then set $N_s := N_s + 1$
   end
Measure computation

- Let \( \mathcal{A} \) be the set of all feasible intractable parameters \( \eta \) and \( P(\mathcal{A}) \) its measure

- **Theorem**
  
  The probability that
  
  \[ | N_s/N_2 - P(\mathcal{A}) | > \varepsilon \]

  holds is less than \( \delta \)

- **Proof** follows easily from the Chernoff bound
Theorem says that Algorithm 2 gives an estimate of the feasibility measure $P(\mathcal{A})$ of $\mathcal{A}$ with accuracy at least $\varepsilon$ and confidence higher than $1 - \delta$

- Sample size $N_2$ depends only on confidence and accuracy
- Algorithm 2 is polynomial-time because sample generation (step 2) and feasibility check (step 3) are polynomial-time
- “Best” proof of the Chernoff bound is based on Hoeffding inequality
Deterministic Methods for Tractable Parameters
Set of stabilizing controllers

- Suppose that an intractable parameter vector $\eta$ is selected according to Algorithm 1

- **Lemma**
  The set of all tractable parameter vectors $\theta$ giving a stabilizing controller is either empty or is a union of a finite number of polyhedral sets
Tractable parameter space

$\Theta$ space

Number of unstable roots

Stability boundary

(2 unstable roots and 2 imaginary roots)
Set of stabilizing controllers

no unstable roots
Related literature

- Previous lemma is a minor extension of results first developed for PID controllers [Ho, Datta, Bhattacharyya, 97]
- Linear programming approach proposed, even if there are only two gains $K_P$ and $K_D$
- Additional results in [Ackermann, Kaesbauer, 03; Soylemez, Munro, Baki, 03]
- Extensions to lead-lag in [Blanchini, Lepsky, Miani, Viaro, 04]
Consider the closed-loop polynomial

\[ p(s) = p_0(s) + p_1(s) X(s^2) \]

where

\[ p_0(s) = sN_P(s)Y(s^2) + D_P(s) (Z(s^2) + s V(s^2)) \]
\[ p_1(s) = N_P(s) \]

For \( s=j\omega \), we have

\[ p_0(j\omega) = R_0(\omega^2) + j\omega I_0(\omega^2) \]
\[ p_1(j\omega) = R_1(\omega^2) + j\omega I_1(\omega^2) \]
Critical frequencies

- The set of critical frequencies are values of $\omega$ such that 
  
  \[ p(j\omega) = 0 \text{ for all } X(-\omega^2) \]

- This set is given by 

  \[ \Omega = \{ \omega_1, \omega_2, \ldots, \omega_{nf} \} \]

  where $\omega_i$ are solutions of the polynomial equation 

  \[ I_0(\omega_i^2) R_I(\omega_i^2) - I_I(\omega_i^2) R_0(\omega_i^2) = 0 \]
Number of critical frequencies

- **Lemma**: The number of critical frequencies is a polynomial function of $n_N$, $n_D$, $n_Y$, $n_Z$ and $n_V$

- **Remark**: Number of critical frequencies does not depend on the number of tractable parameters $\theta$
Stability boundaries

- For each critical frequency we have an hyperplane which defines part of the stability boundary
- Each hyperplane has the form
  \[
  \psi_0(\omega) \theta_0 + \psi_2(\omega) \theta_2 + \ldots + \psi_{n_\theta}(\omega) \theta_{n_\theta} = \upsilon(\omega)
  \]
- In vector form we write
  \[
  \psi_i = [ \psi_0(\omega) \; \psi_2(\omega) \; \ldots \; \psi_{n_\theta}(\omega) ]
  \]
  \[
  \upsilon = [ \upsilon(\omega) \; \upsilon(\omega) \; \ldots \; \upsilon(\omega) ]^T
  \]
  and
  \[
  \Psi = [ \psi_1 \; \psi_2 \; \ldots \; \psi_{n_\theta} ]^T
  \]
Use of linear programming to determine a stabilizer

Since previous equations define only part of the stability boundary, this results into a combinatorial problem

We need to consider *inequality* constraints rather than equalities

\[
\psi_0(\omega_i) \theta_0 + \psi_2(\omega_i) \theta_2 + \ldots + \psi_n(\omega_i) \theta_n \leq \nu(\omega_i)
\]

\[
\psi_0(\omega_i) \theta_0 + \psi_2(\omega_i) \theta_2 + \ldots + \psi_n(\omega_i) \theta_n \geq \nu(\omega_i)
\]

**Drawback:** worst case number of linear programs \(N_{LP}(n_f)\) is exponential
Algorithms for fixed order stabilization

- We propose a different approach
- Use of *vertices* of the stability boundary instead of hyperplanes
- This method is based on two stages
  - computation of *marginal stabilizer*
  - computation of *fixed order stabilizer*
- First step is polynomial-time
- Second step requires one-parameter optimization
(Candidate) marginal stabilizer
(Candidate) marginal stabilizer
All (candidate) marginal stabilizers
Definition

A marginal stabilizer is a controller $C(s, \theta)$ with the property that closed-loop polynomial $p(s, \theta)$ has a fixed number of roots on the imaginary axis and no roots in the open right half plane.
Marginal stabilizer computation

- Recall that we have a matrix \( \Psi \) and a vector \( v \)
- Solution of the combinatoric problem using vertices of the stability boundary
- Construct a square matrix \( \Psi_i \) with \( n \) rows of \( \Psi \) and a vector \( v_i \) with \( n \) corresponding elements of \( v \)
- Obtain a linear system of the form
  \[
  \Psi_i \theta = v_i
  \]
- Marginal stabilizer is immediately given by
  \[
  \theta^{(i)} = \Psi_i^{-1} v_i
  \]
- Number of vertices \( i \) of stability boundary is \( N_{MI}(n_f,n_\theta) \)
Invertibility condition

- Suppose that $n_f \geq n_\theta$

- **Lemma**
  The square matrices $\Psi_i$ are full rank for $i=1, 2, \ldots, N_{MI}$

- **Proof** based on the Vandermonde-like structure of the matrix $\Psi$
Marginal stabilizer computation

- For fixed $i$, $\theta^{(i)}$ is found by matrix inversion
- This requires $O(n_\theta^3)$ operations
- Once a candidate marginal stabilizer $\theta^{(i)}$ is computed, we check with Routh test if it is a marginal stabilizer
- Routh test has polynomial (quadratic) complexity
- Worst case number of matrix inversions is given by

$$N_{Mf}(n_f, n_\theta) = n_f!/(n_\theta! (n_f - n_\theta)!)$$
Suppose that \( n_f \geq n_\theta \)

**Theorem**

1. We have

\[
N_{MI}(n_f, n_\theta) = O(n_f^{n_\theta})
\]

2. Furthermore, we obtain

\[
N_{MI}(n_f, n_\theta) \leq N_{LP}(n_f)
\]

where equality is attained only if \( n_f = 0 \)

**Proof** is based on the so-called bimodal theorem and properties of the factorial
Study fixed order stabilization, hence the number $n_\theta$ of controller parameters $\theta$ is fixed

- We have $N_{MI}(n_f, n_\theta) = O(n_f^{n_\theta})$

- Polynomial-time complexity in the number of critical frequencies and, consequently, in $n_N, n_D, n_Y, n_Z$ and $n_V$

- Number of worst case matrix inversions is much smaller than the number of LPs, see next table
### Comparison of $N_{LP}$ and $N_{MI}$

<table>
<thead>
<tr>
<th></th>
<th>$n_f$</th>
<th>18</th>
<th>16</th>
<th>32</th>
<th>64</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_{MI}(n_f, 2)$</td>
<td>28</td>
<td>120</td>
<td>496</td>
<td>2,016</td>
<td></td>
</tr>
<tr>
<td>$N_{MI}(n_f, 4)$</td>
<td>70</td>
<td>1,820</td>
<td>35,960</td>
<td>$6.8 \cdot 10^5$</td>
<td></td>
</tr>
<tr>
<td>$N_{LP}(n_f)$</td>
<td>256</td>
<td>65,536</td>
<td>$4.2 \cdot 10^9$</td>
<td>$1.8 \cdot 10^{19}$</td>
<td></td>
</tr>
</tbody>
</table>
Theorem

Let \( p(s, \theta^{(i)}) \) be the closed-loop polynomial corresponding to tractable parameter \( \theta^{(i)} \)

- **Theorem**

  There exists a marginal stabilizer if and only if there exists \( \theta^{(i)}, i = 1, 2 \ldots, N_{MI} \), such that \( p(s, \theta^{(i)}) \) has its zeros within the closed left half plane

- **Proof** easily follows from previous discussions
Vertices of stabilizing controller set

- The controller parameter vector $\theta^{(i)}$, if it exists, is a vertex of a polyhedral set of stabilizing controllers.
- The $n_\theta$ rows of the matrix $\Psi_i$ and the $n_\theta$ elements of $\Psi_i$ and of $v_i$ define some of the hyperplanes of the boundary of a polyhedral set of stabilizing controllers.
Marginal stabilizer
Fixed order stabilizer

marginal stabilizer

fixed order stabilizer
Computation of stabilizing controller

- Once a marginal stabilizing controller is determined, we need to compute a fixed-order stabilizing controller.
- Sensitivity method for the zeroes of $p(s,\theta)$ for perturbations of $\theta$.
- Suppose that $\theta$ is a marginal stabilizer so that $p(s,\theta)$ has $k$ simple roots on the imaginary axis (and the remaining in the open left half plane).
- Study how the root $j\omega_i$ moves when perturbing $\theta$ by $\Delta\theta$.
- Consider the analytic function $z_i = z_i(\Delta\theta)$. 
Computation of stabilizing controller:

- Use implicit function theorem to obtain $\frac{\partial z_i}{\partial \theta_k}$
- Want to find $\Delta \theta$ moving all imaginary zeroes inside the open left half plane
- This leads to the solution of a linear system of the kind
  
  $A \Delta \theta = b$

  where $A$ and $b$ are given data and
  
  $\Delta \theta = [\Delta \theta_0 \ \Delta \theta_2 \ \Delta \theta_4 \ \ldots ]^T$

- If $A$ is full rank, we can immediately compute $\Delta \theta$
- Consider a parameter $\theta(i) + \alpha \Delta \theta$, where $\alpha > 0$
- Obtain the optimal $\alpha$ using bisection
Algorithm 3

1. construct $\Psi$ and $\nu$ for given $\eta^{(i)}$
2. for $j := 1, \ldots, N_{MI}(n_f, n_\theta)$ do
   
   begin
   3. compute $\theta^{(j)}$ with matrix inversion
   4. if $\theta^{(j)}$ is a marginal stabilizer then
      
      begin
      5. compute $\Delta \theta$ with matrix inversion
      6. if a stabilizing parameter $\theta^{(j)} + \alpha \Delta \theta$ is found then stop
      
      end
   
   end
Another approach

- The sensitivity method requires two conditions
  - zeroes of $p(s, \theta)$ need to be simple
  - a full rank condition should be satisfied
- If these conditions are not met, we use randomization
- Take a ball (for example, Euclidean) of radius $r > 0$, generate $N$ samples within the ball until we find a fixed order stabilizer
- This procedure is guaranteed to converge because a stabilizer exists within the ball
Randomization again…

marginal stabilizer
Randomization again…
Randomization again…
Randomization again…
Fixed order stabilizer
Example: stabilization

- Unstable plant of order 4 and second order controller

- $N_P(s) = s (1 + 2s + s^2)$
- $D_P(s) = 1 + s + s^2 + s^3 + s^4$
- $N_C(s) = X(s^2) + sY(s^2) = \theta_0 + \alpha_0 s + \theta_2 s^2$
- $D_C(s) = Z(s^2) + sV(s^2) = \beta_0 + \mu_0 s + \beta_2 s^2$
Tractable/intractable parameters

\[ \theta = [\theta_0 \ \theta_2]^T \quad \text{tractable parameters} \]
\[ \eta = [\alpha_0 \beta_0 \ \beta_2 \ \mu_0]^T \quad \text{intractable parameters} \]

- Set confidence and accuracy to
  \[ \varepsilon = 0.01 \text{ and } \delta = 0.01 \]
  and compute
  \[ N_I = 459 \]
Uniform random generation

- Randomly uniformly generate $\beta_2$ and $\mu_0$ in the interval $[0, 5]$.

- Set $\beta_0 = 1$ to obtain a stable controller.

- Since we may need a “non-minimum phase” controller, we randomly uniformly generate $\alpha_0$ in $[-5, 5]$.
Running the algorithms

- We obtained intractable parameters
  \[ \eta = [3.6095 \ 1 \ 4.5845 \ 3.3292]^T \]

- We computed two critical frequencies
  \[ \Omega = \{0.4059, 1.2697\} \]

- Marginal stabilizing (tractable) controller parameters are obtained solving a linear equation
  \[ \theta = [-2.3082 \ -0.3063]^T \]
Stabilizing controller

- A marginal stabilizing controller is
  \[ N_C(s) = -2.3082 + 3.6095 s - 0.3063 s^2 \]
  \[ D_C(s) = 1 + 3.3292 s + 4.5845 s^2 \]

- Then, we computed the desired direction of the perturbation
  \[ \Delta \theta = [13.2513 \, 15.3654]^T \]

- With the step parameter \( \alpha = 0.1 \), we obtained
  \[ \theta + \alpha \Delta \theta = [ -0.9831 \, 1.2303 ]^T \]

- A stabilizing controller is given by
  \[ N_C(s) = -0.9831 + 3.6095 s + 1.2303 s^2 \]
  \[ D_C(s) = 1 + 3.3292 s + 4.5845 s^2 \]
Extensions: $\mathcal{H}_\infty$ performance
\( \mathcal{H}_\infty \) performance

- \( \mathcal{H}_\infty \) performance of sensitivity/complementary sensitivity

\[
S(s) = \frac{1}{1 + P(s) C(s)}
\]

\[
T(s) = 1 - S(s)
\]

- Consider also a (stable) weighting function

\[
W(s) = \frac{N_W(s)}{D_W(s)}
\]
Suppose that an intractable parameter vector $\eta$ is selected.

**Theorem**

The set $\theta$ of all tractable parameter vectors $\theta$ giving an $\mathcal{H}_\infty$ controller satisfying

$$\| W(s) S(s) \| \leq 1$$

is either empty or is given by

$$\theta = \bigcap \Gamma(\varphi)$$

$\varphi \in [0,2\pi)$

where $\Gamma(\varphi)$ is union of finite number of polyhedral sets for fixed $\varphi$. 
(Convex) set of $\mathcal{H}_\infty$ controllers

stability boundary for fixed $\varphi$

$\mathcal{H}_\infty$ stabilizer
Comments

- **Proof** makes use of transformation of $H_\infty$ problem into a stability problem with the additional parameter $\varphi$
- The set $\theta$ is the union of *convex sets* (not necessarily polyhedral)
- This result is a minor extension of [Saeki, Aimoto, 00; Ho, 03; Blanchini, Lepsky, Miani, Viaro, 04]
Computation of critical frequencies

- Consider a closed-loop polynomial which depends on the additional parameter $\varphi$
  \[ p(s,\varphi) = p_0(s,\varphi) + p_1(s,\varphi) X(s^2) \]
- Setting $s = j\omega$ for fixed $\varphi \in [0, 2\pi)$, the critical frequencies are values of $\omega$ such that
  \[ p(j\omega,\varphi) = 0 \text{ for all } X(-\omega^2) \]
Computation of critical frequencies

- **Lemma**
  For fixed $\varphi \in [0, 2\pi)$, the critical frequencies are given by the solution of the equation
  \[ f_0(\omega) + \sin \varphi f_s(\omega) + \cos \varphi f_c(\omega) = 0 \]
  where $f_0(\omega), f_s(\omega)$ and $f_c(\omega)$ are given polynomials

- **Proof** based on lengthy but straightforward computations
Algorithm 4

1. set $\varepsilon > 0$, $n_{\varphi}$, $\Delta n_{\varphi}$, and $n^u$
2. while $n_{\varphi} \leq n^u$ do
   begin
3. construct $\psi$ and $\nu$ for given $\eta^{(i)}$
4. for $j := 1, \ldots, N_MI(n_f, n_{\theta})$ do
   begin
5. compute $\theta^{(j)}$ by matrix inversion
6. if $\theta^{(j)}$ gives an $\varepsilon$-almost stabilizer then
   begin
7. compute $\Delta \theta$ by matrix inversion
8. if a stabilizing parameter $\theta^{(j)} + \alpha \Delta \theta$ is found then stop
   end
   end
9. $n_{\varphi} := n_{\varphi} + \Delta n_{\varphi}$
end
Comments

- Comments similar to those made for Algorithm 3 can be stated here.
- Algorithm 4 is based on combination of matrix inversions and sensitivity methods.
- Main difference with Algorithm 3 if the presence of the additional parameter $\varphi$.
- Grid points for this parameter are needed.
- This increases the number of matrix inversions, which is a function of the number of grid points.
Extensions: stabilization of interval plants
Interval plants

- SISO strictly proper interval plant

\[ P(s,q,r) = \frac{N_P(s,q)}{D_P(s,r)} \]

where \( N_P(s,q) \) and \( D_P(s,r) \) are interval polynomials

- Fixed order controller \( C(s) \)
One-parameter stabilization problems

Transform the fixed-order stabilization problem with interval plant into a number of one-parameter stabilization problems of the form

\[ p(s, \lambda) = N_{i_1}(s) (X(s^2) + sY(s^2)) + \]
\[ D_{i_2, i_3}(s, \lambda) (Z(s^2) + sV(s^2)) \]

and

\[ D_{i_2, i_3}(s, \lambda) = \lambda D_{i_2}(s) + (1 - \lambda D_{i_2}(s)) \]

where \( N_{i_1}(s), D_{i_2}(s), D_{i_2}(s) \) are fixed polynomials associated to the interval plant and \( \lambda \in [0,1] \) is a parameter.
Comments

- Number of one-parameter stabilization problems is 32
- Consider a closed-loop polynomial which depends on the additional parameter $\lambda$
  \[ p(s, \lambda) = p_0(s, \lambda) + p_1(s, \lambda) X(s^2) \]
- Critical frequencies can be computed solving a quadratic equation in $\lambda$
- Approach similar to $\mathcal{H}_\infty$ stabilization, details are tedious
- Polynomial-time algorithm can be designed
Conclusions
Conclusions

- A number of years ago, closed-form meant writing down a “good looking” equation on a piece of paper.

- Due to the increasing computational power, the notion of closed-form solution has changed substantially.

- Various experts (rightfully) convinced us that a new notion of closed-form is to re-write (if possible) a control problem as convex optimization.
Conclusions

• Unfortunately, many important control problems are not convex

• In such cases, we can either introduce relaxation or use randomization

• Perhaps the control community should pay more attention to the latter option accepting a different and weaker notion of problem solution