Scalable Robustness Analysis Using Integral Quadratic Constraints

Anders Rantzer

LCCC Linnaeus Center, Lund University, Sweden

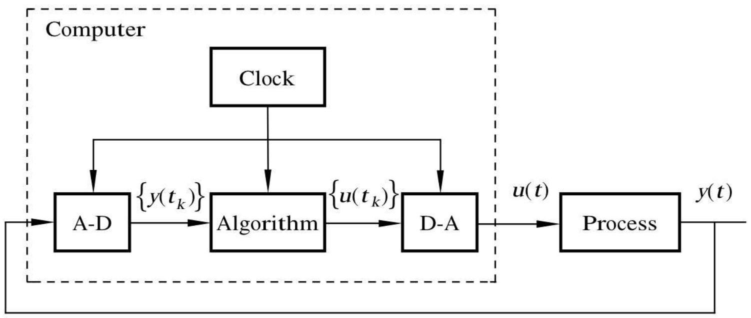
A Grand Challenge

A control system should be delivered with

1. A specification of closed loop **requirements**
2. A network of interconnected **process models** (including controller hardware)
3. A **controller code**
4. A **certificate** proving that code and processes together meet the requirements. Validation of certificates must **scale linearly** with the number of interconnected components.

Is this possible?

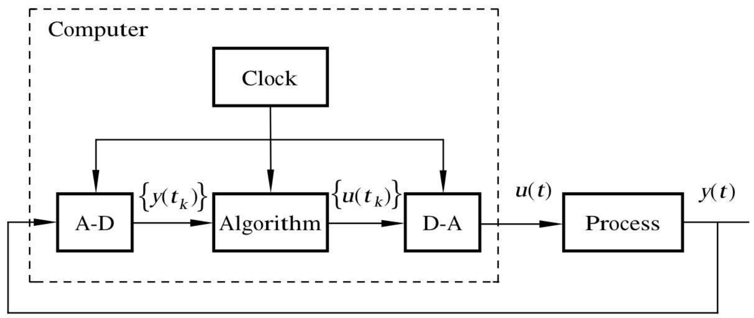
A Standard Setup



For quadratic requirements, linear process model and linear control algorithm, verification is straightforward...

... but is it scalable?

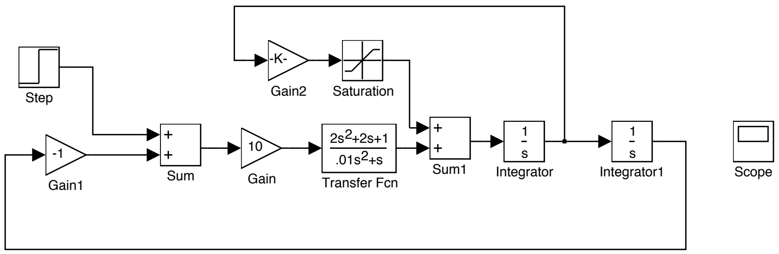
A Standard Setup



For quadratic requirements, linear process model and linear control algorithm, verification is straightforward...

... but is it scalable?

A servo with friction



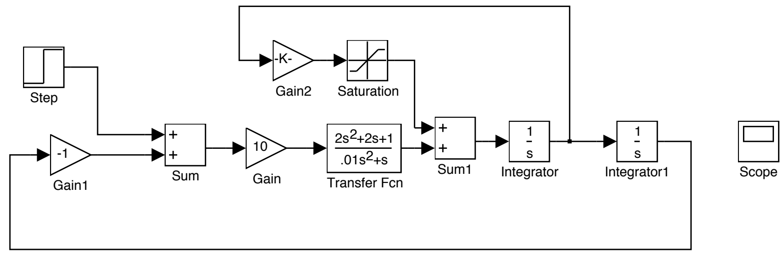
Simulations show stability.

The circle criterion can *prove* stability.

But what if the feedback controller induces time delays?

:

A servo with friction



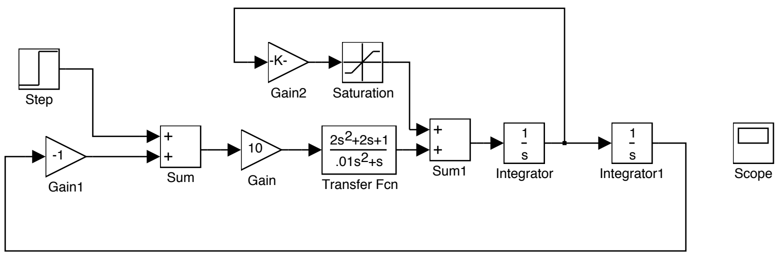
Simulations show stability.

The circle criterion can *prove* stability.

But what if the feedback controller induces time delays?

:

A servo with friction



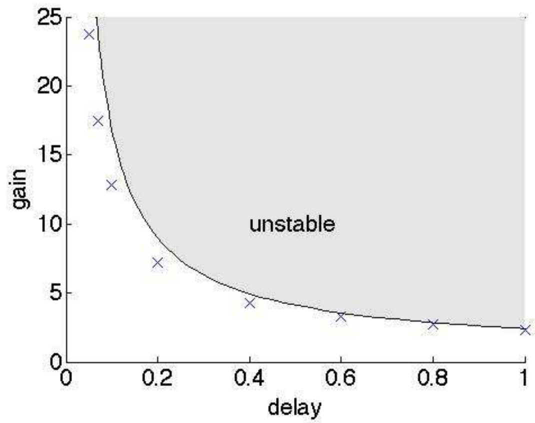
Simulations show stability.

The circle criterion can *prove* stability.

But what if the feedback controller induces time delays?

:

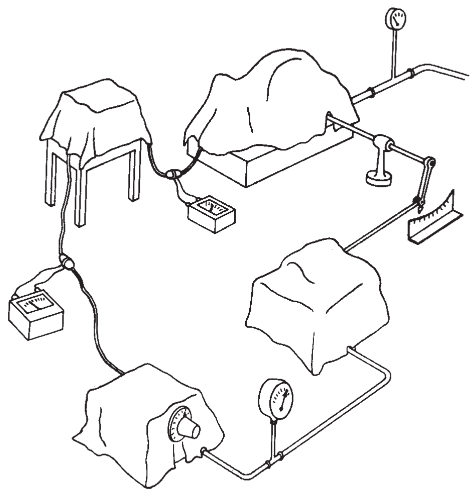
Stability by simulation



Every cross represents a stable simulation.

But what about in between?

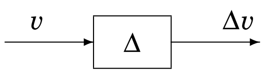
Verification of Complex Systems



Outline

- **Integral Quadratic Constraints**
- A Matlab tool for verification
- Matrix Decompostion
- Making IQC Analysis Scalable
- Conclusions

Integral Quadratic Constraint



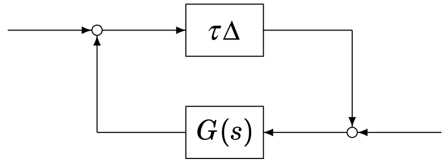
The (possibly nonlinear) operator Δ on $\mathbf{L}_2^m[0,\infty)$ is said to *satisfy the IQC defined by* Π if

$$\int_{-\infty}^{\infty} \begin{bmatrix} \widehat{v}(i\omega) \\ \widehat{(\Delta v)}(i\omega) \end{bmatrix}^* \Pi(i\omega) \begin{bmatrix} \widehat{v}(i\omega) \\ \widehat{(\Delta v)}(i\omega) \end{bmatrix} d\omega \geq 0$$

for all $v \in \mathbf{L}_2[0,\infty)$.

Δ structure	$\Pi(i\omega)$	Condition
Δ passive	$\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$	
$\|\Delta(i\omega)\| \leq 1$	$\begin{bmatrix} x(i\omega)I & 0 \\ 0 & -x(i\omega)I \end{bmatrix}$	$x(i\omega) \geq 0$
$\delta \in [-1,1]$	$\begin{bmatrix} X(i\omega) & Y(i\omega) \\ Y(i\omega)^* & -X(i\omega) \end{bmatrix}$	$X = X^* \geq 0$ $Y = -Y^*$
$\delta(t) \in [-1,1]$	$\begin{bmatrix} X & Y \\ Y^T & -X \end{bmatrix}$	
$\Delta(s) = e^{-\theta s} - 1$	$\begin{bmatrix} x(i\omega)\rho(\omega)^2 & 0 \\ 0 & -x(i\omega) \end{bmatrix}$	$\rho(\omega) =$ $2\max_{\|\theta\|\leq\theta_0} \sin(\theta\omega/2)$

IQC Stability Theorem



Let $G(s)$ be stable and proper and let Δ be causal.

For all $\tau \in [0,1]$, suppose the loop is well posed and $\tau\Delta$ satisfies the IQC defined by $\Pi(i\omega)$. If

$$\begin{bmatrix} G(i\omega) \\ I \end{bmatrix}^* \Pi(i\omega) \begin{bmatrix} G(i\omega) \\ I \end{bmatrix} < 0 \quad \text{for } \omega \in [0,\infty]$$

then the feedback system is input/output stable.

S-procedure

The inequality

$$\sigma_0(h) \leq 0$$

follows from the inequalities

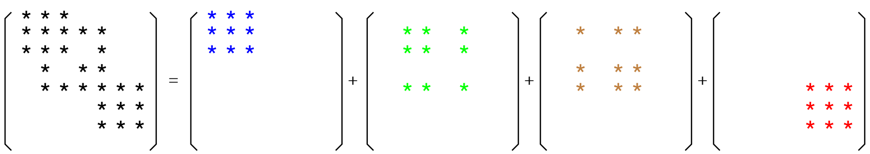
$$\sigma_1(h) \geq 0, \ldots, \sigma_n(h) \geq 0$$

if there exist $\tau_1, \ldots, \tau_n \geq 0$ such that

$$\sigma_0(h) + \sum_k \tau_k \sigma_k(h) \leq 0 \qquad \forall h$$

Underlying Math Problem

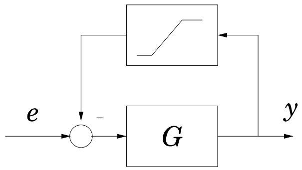
Given a number of symmetric matrices, find a convex combination that is positive definite!

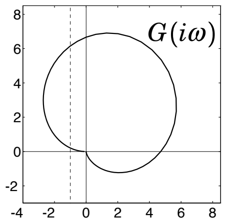


Outline

- Integral Quadratic Constraints
- **A Matlab tool for verification**
- Matrix Decompostion
- Making IQC Analysis Scalable
- Conclusions

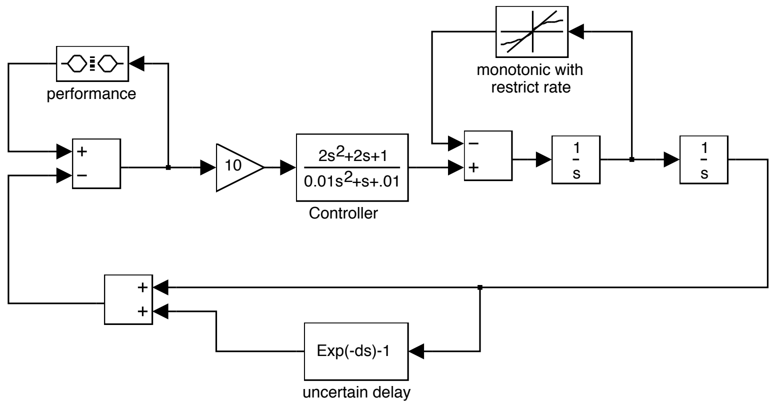
A Matlab toolbox for system analysis





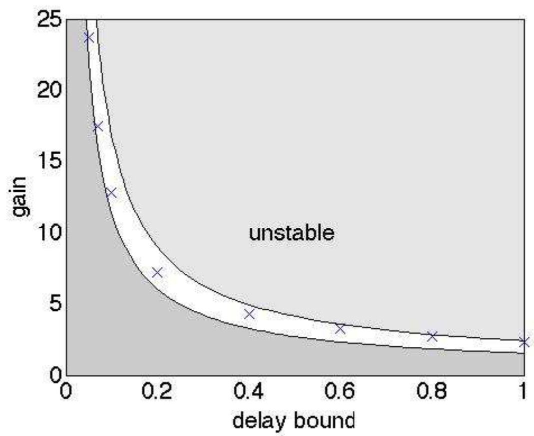

```
>> abst_init_iqc;
>> G = tf([10 0 0],[1 2 2 1]);
>> e = signal
>> w = signal
>> y = -G*(e+w)
>> w==iqc_monotonic(y)
>> iqc_gain_tbx(e,y)
```

An analysis model defined graphically



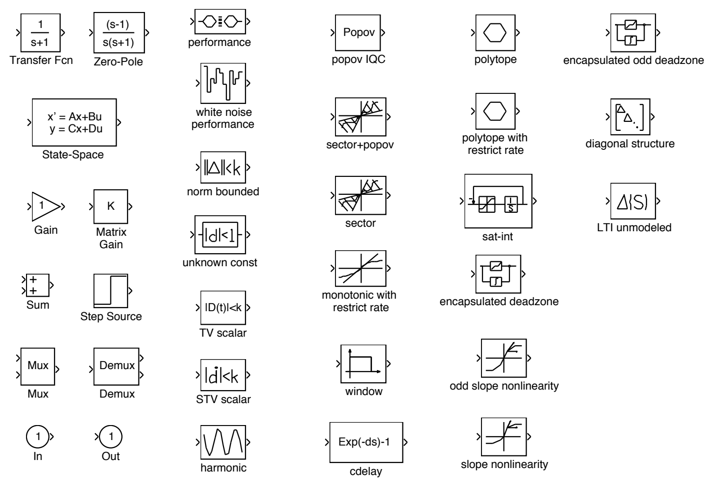

```
>> iqc_gui('fricSYSTEM')

extracting information from fricSYSTEM ...

    scalar inputs:  5
    states:         10
    simple q-forms: 7

    LMI #1    size = 1    states: 0
    LMI #2    size = 1    states: 0
    LMI #3    size = 1    states: 0
    LMI #4    size = 1    states: 0
    LMI #5    size = 1    states: 0

  Solving with 62 decision variables ...

  ans =     4.7139
```

Verification by IQCs



IQCs prove stability below the lower line.

A library of analysis objects



The friction example in text format

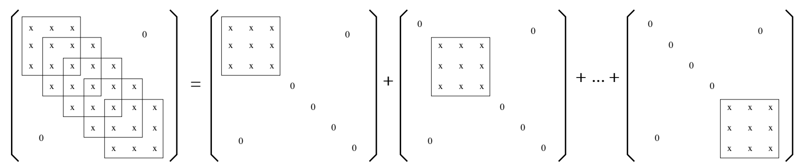
```
d=signal;                              % disturbance signal
e=signal;                              % error signal
w1=signal;                             % friction force
w2=signal;                             % delay perturbation
u=signal;                              % control force
v=tf(1,[1 0])*(u-w1)                   % velocity
x=tf(1,[1 0])*v;                       % position
e==d-x-w2;
u==10*tf([2 2 1],[0.01 1 0.01])*e;
w1==iqc_monotonic(v,0,[1 5],10)
w2==iqc_cdelay(x,.01)
iqc_gain_tbx(d,e)
```


Outline

- Integral Quadratic Constraints
- A Matlab tool for verification
- **Matrix Decompostion**
- Making IQC Analysis Scalable
- Conclusions

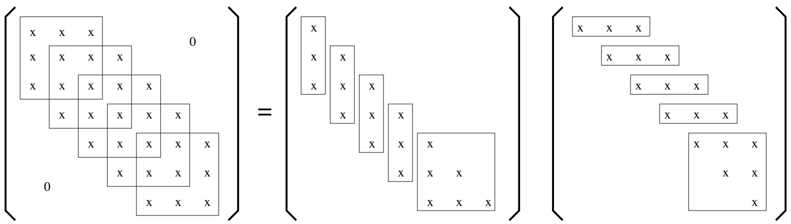
A Matrix Decomposition Theorem

A banded matrix is positive semi-definite if and only if it can be written as a sum of positive semi-definite matrices with the structure on the right.



Proof idea

The decomposition follows immediately from the band structure of the Cholesky factors:



[Martin and Wilkinson, 1965]

Example: Vehicle formation

The first vehicle is controlled to maintain a constant speed:

$$x_1 = G_1 C_1 x_1 + w_1$$

Every other vehicle controls the distance to preceeding vehicle:

$$x_k = G_k C_k (x_{k-1} - x_k) + w_k \qquad\qquad k = 1, \dots, N$$

Is it true that $|x| \le \gamma |w|$ for all w?

(Other requirements can be handled similarly.)

Example: Vehicle formation

$$w = \underbrace{\begin{bmatrix} 1 + G_1 C_1 & & & 0 \\ -G_2 C_2 & \ddots & & \\ & \ddots & & \\ 0 & & -G_N C_N & 1 + G_N C_N \end{bmatrix}}_{M(s)} x$$

Hence $|x| \le \gamma |w|$ if and only if $(1 + G_k C_k)^{-1}$ stable for all k and

$$M(i\omega)^* M(i\omega) - \gamma^{-2} = \begin{bmatrix} * & * & & & \\ * & * & * & & \\ & * & * & \ddots & \\ & & \ddots & * & * \\ & & & * & * \end{bmatrix}$$

is positive semi-definite for all ω.

Example: Vehicle formation

The vehicle formation satisfies $|x| \le \gamma |w|$ for all w if and only if there exist $K_1, \dots, K_N$ with $K_N = 0$, $K_1 = |1 + G_1 C_1|^2$ such that

$$\begin{bmatrix} |G_k C_k|^2 + K_{k-1} - \gamma^{-2} & C_k^* G_k^* (1 + G_k C_k) \\ (1 + G_k C_k)^* G_k C_k & |1 + G_k C_k|^2 - K_k - \gamma^{-2} \end{bmatrix} \succeq 0$$

for $k = 2, \dots, N$.

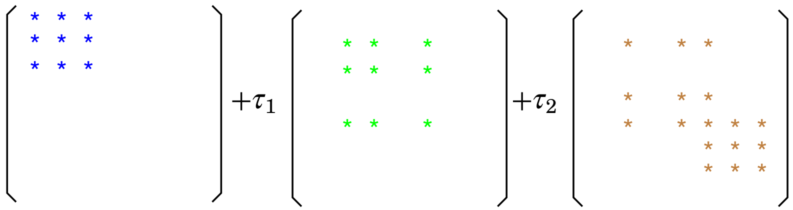
- **Complexity:** Separate test for each vehicle $2, \dots, N$.
- **Confidentiality:** Distributed seach for $K_2, \dots, K_{N-1}$.
- **Transparency:** Use H_∞ optimization to improve C_k.
- If C_k, G_k transfer functions, then K_k frequency dependent.
- Simplify K_k by model reduction.

Outline

- Integral Quadratic Constraints
- A Matlab tool for verification
- Matrix Decompostion
- **Making IQC Analysis Scalable**
- Conclusions

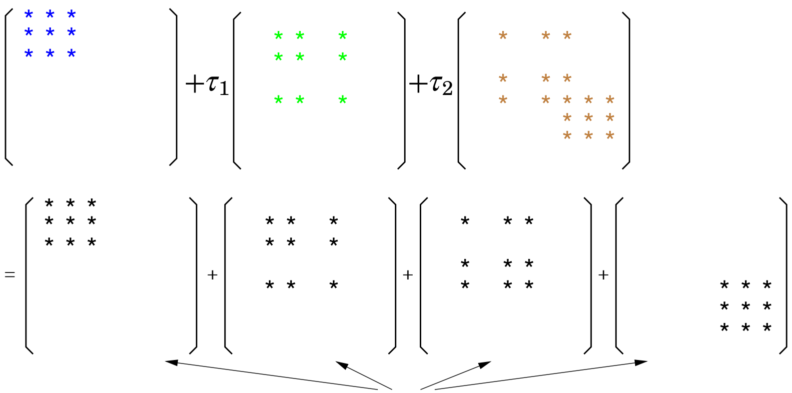
S-procedure for IQC Analysis

Find $\tau_1, \ldots, \tau_n \geq 0$ such that $\sigma_0(h) + \sum_k \tau_k \sigma_k(h)$ becomes negative semi-definite:



Decomposing IQC Analysis

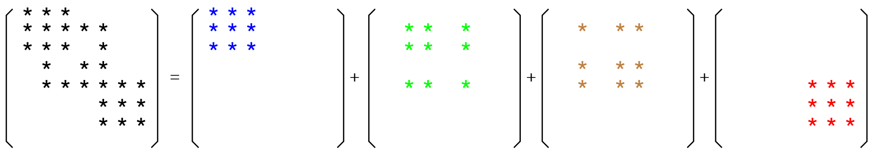
Find $\tau_1, \ldots, \tau_n \geq 0$ such that $\sigma_0(h) + \sum_k \tau_k \sigma_k(h)$ has a negative semi-definite decomposition:

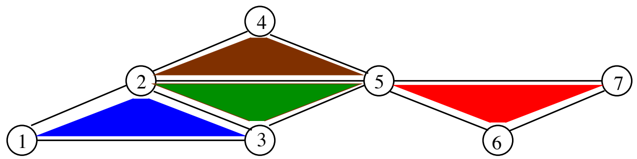


Distributed certificates!

Chordal Decompositions

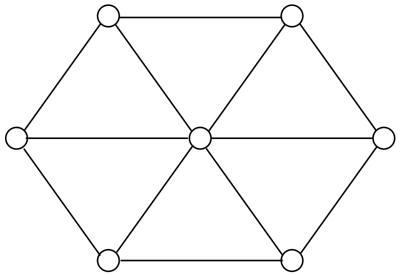
Cholesky factors inherit the sparsity structure of the symmetric matrix if and only if the sparsity pattern corresponds to a "chordal" graph.





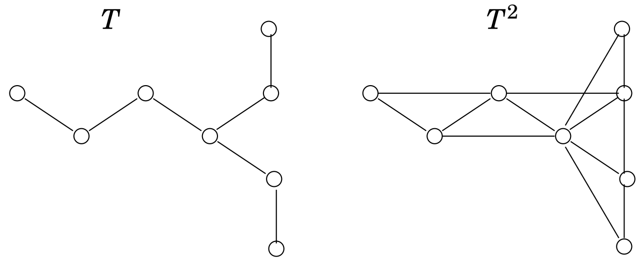
[Blair & Peyton, An introduction to chordal graphs and clique trees, 1992]

Example: Non-chordal graph



Example: Chordal graphs

If T is a tree, then T^k is chordal for every $k \geq 1$.



Network congestion control

Maximize $\sum_i U_i(x)$ over $x_i \geq 0$ subject to $\sum_i R_{li} x_i \leq c_l$

Alternatively: $\min_{p_l \geq 0} \max_{x_i \geq 0} \sum_i \left[U_i(x_i) - \sum_l p_l \left(R_{li} x_i - c_l\right)\right]$

A model for Internet dynamics can look like this:

$$\dot{x}_i(t) = k_i x_i(t) \left(1 - \frac{\sum_l R_{li} p_l(t - \tau_{li})}{U_i'(x_i(t))}\right)$$

$$\beta_l \dot{p}_l(t) + p_l(t) = \sum_i x_i(t - \tau_{il})$$

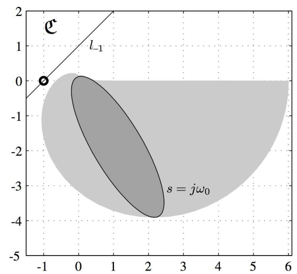
Scalable stabilty conditions by Low, Paganini, Doyle, Papachristodolou, Vinnicombe, Lestas, Pates, . . .

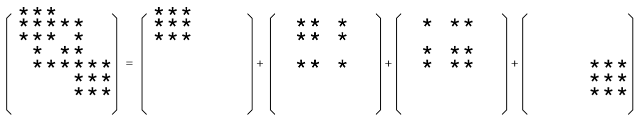
Is there a connection to scalable IQC analysis?

Network congestion control

Yes!
[Pates/Vinnicomb 2012]:

Separate the ellipse $\left\{\frac{z^* L(i\omega) z}{z^* z} : z \in \mathbb{C}^n\right\}$ from -1





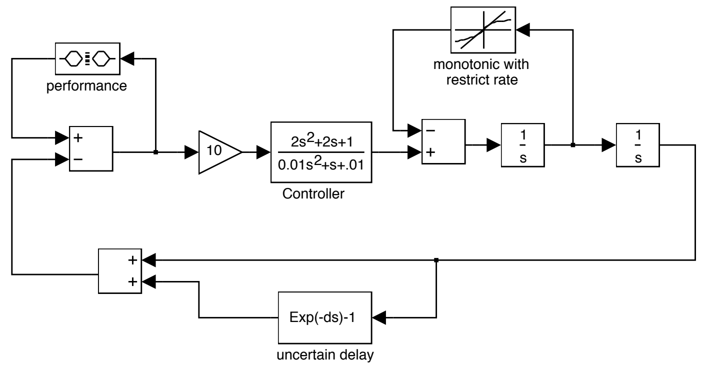
Sources for conservatism: fixed decomposition
fixed separating hyperplane

Outline

- Integral Quadratic Constraints
- A Matlab tool for verification
- Matrix Decompostion
- Making IQC Analysis Scalable
- **Conclusions**

:

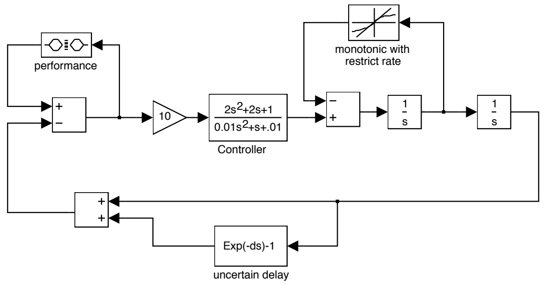
Distributed Verification



[Feron (2010)]: "The credible autocoder produces not only a target code that implements control-system specifications but also documents the target code with its properties and their proofs."

:

Summary



IQC analysis scales using positive definite decompositions !

$$\begin{pmatrix} \star & \star & \star & & & & \\ \star & \star & \star & \star & \star & & \\ \star & \star & \star & & \star & & \\ & \star & & \star & \star & & \\ & \star & \star & \star & \star & \star & \star \\ & & & & \star & \star & \star \\ & & & & \star & \star & \star \end{pmatrix} = \begin{pmatrix} \star & \star & \star & & & & \\ \star & \star & \star & & & & \\ \star & \star & \star & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \end{pmatrix} + \begin{pmatrix} & & & & & & \\ & \star & \star & & \star & & \\ & \star & \star & & \star & & \\ & & & & & & \\ & \star & \star & & \star & & \\ & & & & & & \\ & & & & & & \end{pmatrix} + \begin{pmatrix} & & & & & & \\ & \star & & \star & \star & & \\ & & & & & & \\ & \star & & \star & \star & & \\ & \star & & \star & \star & & \\ & & & & & & \\ & & & & & & \end{pmatrix} + \begin{pmatrix} & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & & & \\ & & & & \star & \star & \star \\ & & & & \star & \star & \star \\ & & & & \star & \star & \star \end{pmatrix}$$

Scalability comes from monotonicity.