Robust Localization Methods for Passivity Enforcement of Linear Macromodels

Zohaib Mahmood¹, Alessandro Chinea², Giuseppe C. Calafiore³, Stefano Grivet-Talocia³ and Luca Daniel¹

¹Massachusetts Institute of Technology, USA
²IdemWorks, Italy
³Politecnico di Torino, Italy

SPI - May 13, 2013
Outline

• Motivation for Compact Dynamic Passive Modeling

• What is Passivity?

• Overview of Existing Techniques

• Enforcing Passivity using Localization Methods

• Results
Motivation for Automatic Model Generation

Linear devices are completely described by their frequency response (Scattering/Impedance parameters)

Compact Model $H(s)$

$H(s)$ must be PASSIVE

Circuit Simulator (Time Domain Simulations)
Passivity

**DEFINITION**

*Passivity is the inability of a system (or model) to generate more energy than what had been stored in it previously.*

*Often referred to as dissipativity in other scientific communities.*
Importance of Preserving Passivity

Problems with non-passive models:
- Time domain circuit simulation may “blow-up”
- Results may become completely non-physical.

\[ H(j\omega) \] (real and imaginary part)
Importance of Preserving Passivity

**Problems with non-passive models:**
- Time domain circuit simulation may “blow-up”
- Results may become completely non-physical.

\[ H(j\omega) \] (real and imaginary part)  \[ y(t) \]

![Diagram showing frequency and time domain representations of a non-passive model and a passive model. The non-passive model's output becomes unstable over time.](image)

*Output of the non-passive model, shown by green lines, becomes unstable.*
Existing Techniques
Overview of Existing Techniques

Simultaneous Pole/Residue Optimization
[Sou2005]

• Guarantees stability and passivity  
• Preserve accuracy (poles are not fixed)  
• Quasi-convex relaxation  
• Solved using localization methods  
• Could be expensive

• Models *parameterized* with 
  certificates of global stability/passivity
Simultaneous Pole/Residue Optimization

\[
\min_{p,q} \left\| \hat{H}(s) - \frac{p(s)}{q(s)} \right\|_{\infty}
\]

• Solves a relaxation to the optimal H-infinity norm problem
• In addition to stability and passivity, that framework can handle constraints such as minimizing quality factor and matching transfer function derivatives
• That framework has been extended to parameterized modeling problems
• An open-source matlab tool, available at: http://www.mit.edu/~dluca/squid/

Overview of Existing Techniques

Simultaneous Pole/Residue Optimization [Sou2005]

Stable Models
[Gustavsen1999, Deschrijver2005]
Overview of Existing Techniques

Simultaneous Pole/Residue Optimization [Sou2005]

Passive Models [Coelho2001]
- Global optimal (for fixed poles)

Stable Models [Gustavsen1999, Deschrijver2005]
Overview of Existing Techniques

Simultaneous Pole/Residue Optimization [Sou2005]

- Passive Models [Coelho2001]
  - Global optimal (for fixed poles)

- Suboptimal (for fixed poles)

- Passivity (Sufficient cond.)

Stable Models
[Gustavsen1999, Deschrijver2005]
Overview of Existing Techniques

Simultaneous Pole/Residue Optimization [Sou2005]

Passive Models [Coelho2001]

Global optimal (for fixed poles)

Suboptimal (for fixed poles)

Passivity (Sufficient cond.) [Saraswat2004, Tommasi2011, Mahmood2012]

Stable Models [Gustavsen1999, Deschrijver2005]
Overview of Existing Techniques

Simultaneous Pole/Residue Optimization [Sou2005]

Perturbation Methods
[Grivet-Talocia2004, Gustavsen2008]

Passive Models
[Coelho2001]

Global optimal (for fixed poles)

Suboptimal (for fixed poles)

Passivity (Sufficient cond.)

Stable Models
[Gustavsen1999, Deschrijver2005]

Suboptimal (for fixed poles)

Perturbation Methods
[Grivet-Talocia2004, Gustavsen2008]
Simultaneous Pole/Residue Optimization [Sou2005]

Overview of Existing Techniques

Perturbation Methods
[Grivet-Talocia2004, Gustavsen2008]

Stable Models
[Gustavsen1999, Deschrijver2005]

Suboptimal
(for fixed poles)

Passive Models
[Coelho2001]

Global optimal
(for fixed poles)

Suboptimal
(for fixed poles)

Passivity (Sufficient cond.)

[This Work, Calafiore2012]
(The problem is posed as convex optimization and solved using localization methods)
Memory Comparison with [Coelho2001]

- 4-Port model with increasing order
- Run on a laptop with 4GB of main memory
Memory Comparison with [Coelho2001]

- 4-Port model with increasing order
- Run on a laptop with 4GB of main memory

![Graph showing memory comparison](image)
Problem Statement
Used Framework

• Algorithmic Flow for Perturbation based Passivity Enforcement

\{(\omega_k, S_k), k = 1..., K\}

Vector Fitting Algorithm
[Gustavsen 1999]

Hamiltonian Matrix Test
[Grivet-Tallocia 2008]

Yes

No

Data

Compute Stable Model

Is Model Passive

Perturb the model

Passive Model
The nominal ‘non-passive’ macromodel (from fitting/identification process)

\[
H(0, s) = C(sI - A)^{-1} B + D
\]

- Scattering matrix
- State-space realization (stable)
- Laplace variable

“0” here denotes “nominal”
Passivity Conditions

\[ \| H(0) \|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathbb{R}} \sigma_1(H(0, j\omega)) \leq 1 \]

Max singular value at single frequency

\[ \sigma_1(X) = \sqrt{\max \lambda(X^H X)} \]
Passivity Enforcement via Perturbation

\[ H(C_p, s) = (C + C_p)(sI - A)^{-1}B + D \]

- Perturbation term

\[ \min \|C_p\|_F, \quad \text{s.t. } \|H(C_p)\|_{\mathcal{H}_\infty} \leq 1 \]

- Minimize perturbation
- Passivity constraint

We want to find the BEST passive macromodel
not just ONE passive macromodel
Choice of Objective Function

\[
\begin{align*}
\text{minimize } & \| C_p \|_F \\
\text{for Minimal perturbation}
\end{align*}
\]

\[
\begin{align*}
\text{minimize } & \| C_p K^T \|_F \\
\text{for Minimal impulse response perturbation [Grivet-Talocia2004]}
\end{align*}
\]

‘K’ is the Cholesky factor of the controllability Grammian of the system
Choice of Objective Function

\[ \min \| C_p \|_F, \quad \text{s.t.} \quad \| H(C_p) \|_{H_\infty} \leq 1 \]

Change of notation

\[ \min_x f(x) \quad \text{s.t.} \quad h(x) \leq 0 \]

where: \( x = \text{vec}(C_p), \quad x \in \mathbb{R}^n \)

\[ f(x) = \| x \|_2 = \| C_p \|_F, \quad h(x) = \| H(C_p) \|_{H_\infty} - 1 \]
Compact Formulation of Passivity Enforcement

\[
\min_x f(x) \quad \text{s.t.} \quad h(x) \leq 0
\]

Our cost function is a norm
any norm is convex

The function \( h(x) \) is convex
(but non-smooth)\(^1\)

Compact Formulation of Passivity Enforcement

\[
\min_{x} f(x) \quad \text{s.t.} \quad h(x) \leq 0
\]

Our cost function is a norm
any norm is convex

The function \(h(x)\) is convex
(but non-smooth)\(^{1}\)

Compact Formulation of Passivity Enforcement

\[
\min_{x} f(x) \quad \text{s.t.} \quad h(x) \leq 0
\]

Our cost function is a norm
any norm is convex

\(f(x)\)

The function \(h(x)\) is convex
(but non-smooth)\(^1\)

Convex function
(finding global minimum-easy)

Non-convex function
(finding global minimum extremely difficult)

Localization Based Methods

(These methods can handle convex non-smooth functions)
Localization Based Methods

* Global optimal
Localization Based Methods

Global optimal
Localization Based Methods

Algorithm terminates when the updated search space is ‘small enough’

* Global optimal
Using Localization Methods

Problem specific details of the localization methods

- How to define the initial search space guaranteed to contain the global optimal?

- How to define a cutting plane that reduces the search space?

- How to update the search space?
Challenge: How to Define the Initial Set

\[ f(x) = \|x\|_2 \quad \text{(Minimal perturbation)} \]

Feasible region

\[ h(x) \leq 0 \]

No perturbation
Observation

\[ f(x) = \|x\|_2 \]  

(Minimal perturbation)

**Observation:** Any feasible point \( x_F \) has \( f(x_F) \geq f(x^*) \)
Observation

$$f(x) = \|x\|_2$$  (Minimal perturbation)

OBSERVATION: Any feasible point $x_F$ has $f(x_F) \geq f(x^*)$
The Initial Set

\[ f(x) = \|x\|_2 \]  

(Minimal perturbation)

Initial set is a hypersphere with radius \( R = \|x_F\|_2 \)

Feasible region

\( h(x) \leq 0 \)

A feasible point

No perturbation

\( x_F \)
The Initial Set: A Feasible Point

\[ H(C_p, s) = (C + C_p)(sI - A)^{-1}B + D \]

Can we find a passive system analytically?
The Initial Set: A Feasible Point

\[ H(C_p, s) = (C + C_p)(sI - A)^{-1}B + D \]

Can we find a passive system analytically?

\[ C_p = -C \implies H(C_p, s) = D \]

*IS PASSIVE*
The Initial Set: A Feasible Point

\[ H(C_p, s) = (C + C_p)(sI - A)^{-1}B + D \]

Can we find a passive system analytically?

\[ C_p = -C \implies H(C_p, s) = D \]

**IS PASSIVE**

\[ x_F = -\text{vec}(C) \]

*Is a feasible point*
The Ellipsoid Algorithm*
The Ellipsoid Algorithm*

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{s.t.:} & \quad h(x) \leq 0
\end{align*}
\]

Feasible region

\[h(x) \leq 0\]

No perturbation

[*Bland1981]
The Ellipsoid Algorithm*

\[
\begin{align*}
\text{minimize } & \quad f(x) \quad \text{s.t.: } h(x) \leq 0 \\
\end{align*}
\]

Feasible region

\[ h(x) \leq 0 \]

\[ x^{(0)} \]

\[ h(x^{(0)}) > 0 \]

\[ g \in \partial h \]

\[ \mathcal{E}^{(0)} \]

[*Bland1981]
The Ellipsoid Algorithm*

minimize $f(x)$ s.t.: $h(x) \leq 0$

Feasible region

$h(x) \leq 0$

$\mathcal{E}^{(0)}$

$g \in \partial h$

$\mathbf{x}^{(0)}$

[*Bland1981]
The Ellipsoid Algorithm*

minimize \( f(x) \) s.t.: \( h(x) \leq 0 \)

Feasible region

\[ h(x) \leq 0 \]

\[ h(x^{(0)}) > 0 \]

\[ g \in \partial h \]

\[ \mathcal{E}^{(1)} \in \mathcal{E}^{(0)} \cap \{x \mid g^T (x - x^{(0)}) \leq 0\} \]

[*Bland1981]
The Ellipsoid Algorithm*

minimize $f(x)$ \text{s.t.:} $h(x) \leq 0$

Feasible region

$\mathcal{E}^{(1)}$

$\mathcal{E}^{(0)}$

$h(x) \leq 0$
The Ellipsoid Algorithm\textsuperscript{*}

\[
\text{minimize } f(x) \quad \text{s.t. } h(x) \leq 0
\]

\[\mathcal{E}^{(1)}\]

Feasible region

\[h(x) \leq 0\]

\textsuperscript{*Bland1981}
The Ellipsoid Algorithm*

\[
\text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0
\]

\[E^{(1)}\]

Feasible region

\[h(x) \leq 0\]

\[h(x^{(1)}) < 0\]

\[*Bland1981*]
The Ellipsoid Algorithm*

\[ \text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0 \]

Feasible region

\[ h(x) \leq 0 \]

\[ h(x^{(1)}) < 0 \]

\[ g = \nabla f \]

\[ x^{(1)} \]

[*Bland1981]
The Ellipsoid Algorithm*

\[
\begin{align*}
\text{minimize } & \quad f(x) \\
\text{s.t.} & \quad h(x) \leq 0 \\
\end{align*}
\]

Feasible region

\[h(x) \leq 0\]

\[h(x^{(1)}) < 0\]

\[g = \nabla f\]

\[x^{(1)}\]

[*Bland1981*]
The Ellipsoid Algorithm

\[
\text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0
\]

\[\mathcal{E}^{(1)}\]

Feasible region

\[h(x) \leq 0\]

\[h(x^{(1)}) < 0\]

\[\mathcal{E}^{(2)} \in \mathcal{E}^{(1)} \cap \{x \mid g^T (x - x^{(1)}) \leq 0\}\]

[*Bland1981]
The Ellipsoid Algorithm*

\[
\text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0
\]

\[x \in \mathcal{E}^{(1)}\]

[Feasible region]

\[h(x) \leq 0\]

\[h(x^{(1)}) < 0\]

[\*Bland1981]
The Ellipsoid Algorithm*

\[
\text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0
\]

[Feasible region]

\[
h(x) \leq 0
\]

[*Bland1981]
The Ellipsoid Algorithm*

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{s.t.} & \quad h(x) \leq 0
\end{align*}
\]

Feasible region

\[h(x) \leq 0\]

Converge when the ellipsoid is small enough

\[*\text{Bland1981}\]
The Ellipsoid Algorithm

An ellipsoid is described as
\[ \mathcal{E}(x, P) = \{ z \mid (z - x)^T P^{-1} (z - x) \leq 1 \} \]

Given: \( f, h \) and \( \mathcal{E}^{(0)}(x^{(0)}, P^{(0)}) \) containing \( x^* \)

Gradient selection: \( g_k = \begin{cases} \nabla f(x^{(k)}) & \text{if } h(x^{(k)}) \leq 0 \\ \partial h(x^{(k)}) & \text{if } h(x^{(k)}) > 0 \end{cases} \)

Convergence check:
\[
\begin{align*}
    h(x^{(k)}) &\leq 0 \\
    \sqrt{g_k^T P^{(k)} g_k} &\leq \delta \quad \text{return } x^{(k)} \quad \text{QUIT}
\end{align*}
\]
The Ellipsoid Algorithm

An ellipsoid is described as
\[ \mathcal{E}(x, P) = \{ z \mid (z - x)^T P^{-1} (z - x) \leq 1 \} \]

Given: \( f, h \) and \( \mathcal{E}^{(0)}(x^{(0)}, P^{(0)}) \) containing \( x^* \)

Gradient selection: \( g_k = \begin{cases} \nabla f(x^{(k)}) & \text{if } h(x^{(k)}) \leq 0 \\ \partial h(x^{(k)}) & \text{if } h(x^{(k)}) > 0 \end{cases} \)

Update: \( x^{(k+1)} = x^{(k)} - \frac{1}{n+1} \frac{P^{(k)}}{\sqrt{g_k^T P^{(k)} g_k}} \),

\[
P^{(k+1)} = \frac{n^2}{n^2 - 1} \left[ P^{(k)} - \frac{2}{n+1} \frac{P^{(k)} g_k^T g_k P^{(k)}}{g_k^T P^{(k)} g_k} \right]
\]
The Ellipsoid Algorithm

An ellipsoid is described as
\[ \mathcal{E}(x, P) = \{ z \mid (z - x)^T P^{-1} (z - x) \leq 1 \} \]

Given: \( f, h \) and \( \mathcal{E}^{(0)}(x^{(0)}, P^{(0)}) \) containing \( x^* \)

Gradient selection: \( g_k = \begin{cases} \nabla f(x^{(k)}) & \text{if } h(x^{(k)}) \leq 0 \\ \partial h(x^{(k)}) & \text{if } h(x^{(k)}) > 0 \end{cases} \)

Update: \( x^{(k+1)} = x^{(k)} - \frac{1}{n+1} \frac{P^{(k)}}{\sqrt{x^T P^{(k)} g_k}} g_k \)

\[ P^{(k+1)} = \frac{n^2}{n^2 - 1} \left( P^{(k)} - \frac{1}{n+1} \frac{P^{(k)} g_k g_k^T P^{(k)}}{g_k^T P^{(k)} g_k} \right) \]

\( O(n^2) \)
Properties of The Ellipsoid Algorithm

• Modest storage  \( O(n^2) \quad x \in \mathbb{R}^n \)

• Modest computation per iteration  \( O(n^2) \quad x \in \mathbb{R}^n \)

• Volume reduction  \( \text{vol}(\mathcal{E}^{(k+1)}) < e^{\frac{-1}{2n}} \text{vol}(\mathcal{E}^{(k)}) \)

• \textit{ROBUST}, could take more iterations
Example 1: Large Passivity Violation (2 Port)

Singular values of the original non-passive model
Example 1: Large Passivity Violation (2 Port)

Singular values of the original non-passive model

Large passivity violation
Example 1: Large Passivity Violation (2 Port)

Singular values of the original non-passive model
Singular values of the perturbed model using [Gustavsen2008] (after 5 iterations)
Example 1: Large Passivity Violation (2 Port)

Singular values of the original non-passive model
Singular values of the perturbed passive model
Example 1: Large Passivity Violation (2 Port)

Standard Vector Fitting: model with N=36 states

Normalized Frequency

\[ \Re S_{1,1} \]

Original

Passive

Normalized Frequency

\[ \Im S_{1,1} \]

Original

Passive

Normalized Frequency
Example 1: Large Passivity Violation (2 Port)

Convergence plots for difference radii of the initial hyper-sphere

![Graph showing convergence plots for different radii of the initial hyper-sphere. The x-axis represents the number of iterations, the y-axis represents the objective function |Cp|, and the graph includes lines for different initial hypersphere radii: 0.06, 0.10, and 0.15. The lines are distinguished by colors: red, green, and blue, respectively.](image-url)
Example 1: Large Passivity Violation (2 Port)

Convergence plots for difference radii of the initial hyper-sphere

Objective function $|C_p|$

- Radius of Initial Hypersphere = 0.06
- Radius of Initial Hypersphere = 0.10
- Radius of Initial Hypersphere = 0.15

Took only 55 seconds

[On a laptop with 4GB main memory]
Example 2: 4-Port Models

- 4-Port model with increasing order
- Run on a laptop with 4GB of main memory

```
Example 2: 4-Port Models

- 4-Port model with increasing order
- Run on a laptop with 4GB of main memory
```

```
[Coelho2001]
This work

Model Order
Allocated Memory (GB)

2 GB !!!
2 MB
203 sec.

```

```
Example 2: 4-Port Models

- 4-Port model with increasing order
- Run on a laptop with 4GB of main memory
```

```
[Coelho2001]
This work

Model Order
Allocated Memory (GB)

2 GB !!!
2 MB
203 sec.

```
The Cutting Plane Method*

[*Kelley1960]
A Polyhedron

Intersection of a finite number of half-spaces

\[ P = \{ x \mid Ax \leq b \} \]

\[
A = \begin{bmatrix}
  a_1^T \\
  \vdots \\
  a_5^T
\end{bmatrix}
\]
The Cutting Plane Method*

\[
\text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0
\]

Feasible region

\[h(x) \leq 0\]

\[x^{(0)}\]

[*Kelley1960]
The Cutting Plane Method*

minimize $f(x)$ s.t.: $h(x) \leq 0$

Feasible region

$h(x) \leq 0$

$x^{(0)}$

[*Kelley1960*]
The Cutting Plane Method*

minimize $f(x)$  s.t.: $h(x) \leq 0$

Feasible region

$h(x) \leq 0$

[*(Kelley1960]*
The Cutting Plane Method*

minimize $f(x)$ s.t.: $h(x) \leq 0$

[$*$Kelley1960]
The Cutting Plane Method*

minimize \( f(x) \)  s.t.: \( h(x) \leq 0 \)

Feasible region

\( h(x) \leq 0 \)

[*Kelley1960]
The Cutting Plane Method*

minimize $f(x)$  s.t.: $h(x) \leq 0$

Feasible region

$h(x) \leq 0$

Where to define the next cut?

[*Kelley1960]
The Cutting Plane Method

minimize $f(x)$ s.t.: $h(x) \leq 0$

Feasible region

$h(x) \leq 0$

Where to define the next cut?

In principle we can define a cut anywhere in the polyhedron

[*Kelley1960]*
The Cutting Plane Method*

minimize \( f(x) \) s.t. \( h(x) \leq 0 \)

In principle we can define a cut anywhere in the polyhedron

\[ h(x) \leq 0 \]

[*)Kelley1960]
The Cutting Plane Method*

In principle we can define a cut anywhere in the polyhedron.

Not a good cut because it eliminates only a small part of the polyhedron.

In principle we can define a cut anywhere in the polyhedron.

\[
\text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0
\]

\[h(x) \leq 0\]

Feasible region

[*Kelley1960]
The Cutting Plane Method*

minimize $f(x)$ s.t.: $h(x) \leq 0$

Search for the center of the polyhedron (Chebychev center or Analytic center)

Feasible region

$h(x) \leq 0$

$h(x^{(1)}) < 0$

$x^{(1)}$

$g = \nabla f$

[1960]
The Cutting Plane Method*

minimize $f(x)$ \ s.t.: $h(x) \leq 0$

*$[Kelley1960]$

Feasible region

$h(x) \leq 0$

$h(x^{(1)}) < 0$

$x^{(1)}$

$g = \nabla f$

Search for the center of the polyhedron (Chebychev center or Analytic center)
The Cutting Plane Method*

minimize $f(x)$ s.t.: $h(x) \leq 0$

[*Kelley1960]
The Cutting Plane Method*

minimize $f(x)$  s.t.: $h(x) \leq 0$

Feasible region

$h(x) \leq 0$

[*Kelley1960]
minimize $f(x)$  s.t.: $h(x) \leq 0$

[*Kelley1960*]
The Cutting Plane Method*

\[
\text{minimize } f(x) \quad \text{s.t.: } h(x) \leq 0
\]

[Feasible region]

\[
h(x) \leq 0
\]

[*Kelley1960]
The Cutting Plane Method*

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{s.t.:} & \quad h(x) \leq 0
\end{align*}
\]

Feasible region

\[h(x) \leq 0\]

Converge when the polyhedron is small enough

[*Kelley1960]
The Cutting Plane Method

A polyhedron is described as

\[ P = \{ z \mid Az \leq b \} \]

Given: \( f, h \) and \( P^{(0)} \) containing \( x^* \), \( f_{LB}^{(0)} = 0 \)

Gradient selection: \( g_k = \begin{cases} \nabla f(x^{(k)}) & \text{if } h(x^{(k)}) \leq 0 \\ \partial h(x^{(k)}) & \text{if } h(x^{(k)}) > 0 \end{cases} \)

Convergence check:

\[ h(x^{(k)}) \leq 0 \]

\[ \left| f(x^{(k)}) - f_{LB}^{(k)} \right| \leq \delta \quad \text{return } x^{(k)} \quad \text{QUIT} \]
The Cutting Plane Method

A polyhedron is described as

\[ P = \{ z \mid Az \leq b \} \]

Given: \( f, h \) and \( P^{(0)} \) containing \( x^* \), \( f_{LB}^{(0)} = 0 \)

Gradient selection: \( g_k = \begin{cases} \nabla f(x^{(k)}) & \text{if } h(x^{(k)}) \leq 0 \\ \partial h(x^{(k)}) & \text{if } h(x^{(k)}) > 0 \end{cases} \)

Update:

\[ P^{(k+1)} = P^{(k)} \cap \{ z \mid a_{k+1}^T z \leq b \} \quad \text{Adding a new row to the A matrix} \]

\[ x^{(k+1)} \in P^{(k+1)} \quad \text{Chebychev or Analytic center} \]

\[ f_{LB}^{(k+1)} \quad \text{Piecewise Linear Lowerbound} \]
Properties of The Cutting Plane Method

- **LOWER BOUND on the global minimum:**
  - Computed as a by-product during the update
  - Lower bound gets tighter as the algorithm progresses

- Modest complexity per iteration $O(n^2 m)$ \(x \in \mathbb{R}^n\)
- **ROBUST**, could take more iterations

- **Extensions**
  - Multiple Cuts
  - Dropping Constraints
Example 1: Large Passivity Violation (2 Port)

Singular values of the original non-passive model
Singular values of the perturbed passive model
Example 1: Large Passivity Violation (2 Port)

Standard Vector Fitting: model with N=36 states
Example 1: Large Passivity Violation (2 Port)

Convergence plots for difference side length of the initial hyper-cube
Conclusions

- Convergence is always guaranteed…
  - … to the \textit{optimal} passive macromodel…
  - … not just one passive macromodel…
  - … the best macromodel (in the preferred norm)

- Lower bound on the Global Optimal

- Modest storage \(O(n^2)\) or \(O(n^2m)\)

- Issues and future work
  - Many iterations may be required
  - Work in progress to speed up the algorithm