

A Pole-Residue Framework for Optimal \mathcal{H}_2 Model Reduction

Patrick Sheridan

Harold Metz

May 8, 2009

1 Abstract

In this paper, we describe our attempts to develop an algorithm for solving the optimal \mathcal{H}_2 model reduction problem in a pole-residue framework using Newton's method. To this end, we convert the moment matching problem into a form where it requires finding the zero of a function. This function takes in the poles and residues of a reduced system as independent variables and is zero at the optimum reduced poles and residues. Newton's method is then applied to the function to numerically find the zero. We discovered that this problem does not seem well suited to be solved using Newton's method but converges much faster using the fixed point method of the iteratively corrected rational Krylov algorithm.

2 Introduction

The model reduction problem aims to replace a dynamical system of first order ordinary differential equations with another smaller system of differential equations which gives similar outputs but is easier to use due to its reduced size. For our purposes here, we only consider stable linear systems with a single input and single output of the form:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t)$$

$$y(t) = \mathbf{c}^T \mathbf{x}(t)$$

where \mathbf{A} is an $n \times n$ real matrix, \mathbf{b} and \mathbf{c} are real n -tuples and $u(t)$ is the input signal. The transfer function $G(s)$ is found by taking the Laplace transform of the state equations and solving for $\frac{Y(s)}{U(s)}$ and is given in the Laplace domain as:

$$G(s) = \mathbf{c}(s\mathbf{I}_n - \mathbf{A})^{-1}\mathbf{b}$$

The goal is to replace this system with a new system

$$G_r(s) = \mathbf{c}_r^T (s\mathbf{I} - \mathbf{A}_r)\mathbf{b}_r$$

where \mathbf{A}_r is an $r \times r$ real matrix, \mathbf{b}_r and \mathbf{c}_r are real r -tuples and r is less than n while maintaining a close approximation of the input-output relationship. From this reduction process, there are three criteria which we desire to meet:

1. The difference in the \mathcal{H}_2 norm of $G - G_r$ is small.
2. Defining characteristics of the original system, such as bounded-input - bounded-output (BIBO) stability, are preserved.
3. The reduction process can be achieved in a way which is tractable and numerically stable.

We thus began literature review process to examine methods currently in use.

3 Background

3.1 Dynamical Systems

A dynamical system is a mapping which describes how a dependent variable changes with time. For the purposes of this paper, we are concerned with dynamical systems described by a system of first order linear differential equations with a single input and a single output (hereafter called SISO). Such dynamical systems can be described by the following state equations:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}u(t)$$

$$y(t) = \mathbf{c}^T \mathbf{x}(t)$$

where \mathbf{A} is an $n \times n$ real matrix, $u(t)$ and $y(t)$ are real valued functions, and \mathbf{b} , $\mathbf{x}(t)$, and \mathbf{c} are real n-tuples. The variables $u(t)$, $\mathbf{x}(t)$, and $y(t)$ are called the input, state, and output, respectively. So, at any time t , the system is given a real value for $u(t)$ as an input. Solving the set of linear first order differential equations $\mathbf{x}' = \mathbf{A}\mathbf{x} + \mathbf{b}u$ yields a value for $\mathbf{x}(t)$, the state of the system. Given the state of the system, one can solve for the output variable $y(t)$ by $y(t) = \mathbf{c}^T \mathbf{x}$.

In these types of settings, one often looks at the transfer function of the system as a convenient means of defining the system. If we take the Laplace transform of the system, we see the following set of equations:

$$s\mathbf{X}(s) = \mathbf{A}\mathbf{X}(s) + \mathbf{b}U(s)$$

$$Y(s) = \mathbf{c}\mathbf{X}(s)$$

where $\mathbf{X}(s)$, $Y(s)$ and $U(s)$ are the Laplace transforms of the functions $\mathbf{x}(t)$, $y(t)$ and $u(t)$, respectively. Solving for $\mathbf{X}(s)$ in the first equation gives $\mathbf{X}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}U(s)$. Substituting this into the second equation gives $Y(s) = \mathbf{c}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}U(s)$. The transfer function of the system is defined as the function $G(s) = Y(s)/U(s)$. So in this case, the system is described by $G(s) = \mathbf{c}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}$

As a final remark, we note that it is well known that this transfer function of a system with simple roots in the denominator, can be expressed in the following form, found by using partial fraction expansion:

$$G(s) = \sum_{i=1}^n \frac{\phi_i}{s - \lambda_i}$$

where each ϕ_i and λ_i are called residues and poles of the system, respectively. This form is thus known as pole-residue form.

3.2 Model Order Reduction

The goal of the model reduction problem is as follows: given a dynamical system $G(s) = \mathbf{c}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}$ with $A \in \mathbb{R}^{n \times n}$ and $b, c \in \mathbb{R}^n$, find another dynamical system $G_r(s) = \mathbf{c}_r(s\mathbf{I} - \mathbf{A}_r)^{-1}\mathbf{b}_r$ with $A \in \mathbb{R}^{r \times r}$ and $b, c \in \mathbb{R}^r$, and r less than n where $|y - y_r|$ is small for all inputs $u(t)$. G_r should be able to be computed in an efficient manner.

To do this, we first consider the \mathcal{H}_2 norm of a dynamical system defined by

$$\|G(s)\|_{\mathcal{H}_2}^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} |G(j\omega)|^2 d\omega \stackrel{\text{Parseval's Theorem}}{=} \int_0^{\infty} |g(t)|^2 dt$$

where $g(t)$ is the impulse response of $G(s)$ in the time domain Using the definition of the \mathcal{H}_2 norm, we can redefine the problem by finding G_r such that the \mathcal{H}_2 norm of $G - G_r$ is minimized. Antoulas [1] was able to give the following characterization of the \mathcal{H}_2 norm:

$$\|G\|_{\mathcal{H}_2}^2 = \sum_{i=1}^n \phi_i G(-\lambda_i)$$

This yields the following result stated by Gugercin [3]:

$$\mathcal{E} := \|G(s) - G_r(s)\|_{\mathcal{H}_2}$$

$$\mathcal{E} = \sum_{i=1}^n \phi_i (G(-\lambda_i) - G_r(-\lambda_i)) + \sum_{j=1}^r \hat{\phi}_j (G(-\hat{\lambda}_j) - G_r(-\hat{\lambda}_j))$$

Therefore, $G_r(s)$ is the optimum system when (the error) is minimized. Gugercin [3] then goes on to show that

$$\left. \begin{array}{l} \frac{\partial \mathcal{E}}{\partial \phi_i} = 0 \\ \frac{\partial \mathcal{E}}{\partial \hat{\lambda}_i} = 0 \end{array} \right\} \Rightarrow \begin{array}{l} G(-\hat{\lambda}_i) = G_r(-\hat{\lambda}_i) \\ G'(-\hat{\lambda}_i) = G'_r(-\hat{\lambda}_i) \end{array}$$

This result will be what we use to compute $G_r(s)$.

4 Hermite Interpolation

Recall that in order to satisfy the \mathcal{H}_2 optimality conditions it is necessary that $G(-\hat{\lambda}_i) = G_r(-\hat{\lambda}_i)$ and $G'(-\hat{\lambda}_i) = G'_r(-\hat{\lambda}_i)$. In order to accomplish this, we list the following method found in Gugercin's paper [3], given without proof:

Given $\{\sigma\}_{i=1}^r$ interpolation points:

- $\mathbf{V} = [(\sigma_1 \mathbf{I} - \mathbf{A})^{-1} \mathbf{b}, (\sigma_2 \mathbf{I} - \mathbf{A})^{-1} \mathbf{b}, \dots, (\sigma_r \mathbf{I} - \mathbf{A})^{-1} \mathbf{b}]$
- $\mathbf{Z} = [(\overline{\sigma}_1 \mathbf{I} - \mathbf{A}^T)^{-1} \mathbf{c}^T, (\overline{\sigma}_2 \mathbf{I} - \mathbf{A}^T)^{-1} \mathbf{c}^T, \dots, (\overline{\sigma}_r \mathbf{I} - \mathbf{A}^T)^{-1} \mathbf{c}^T]$
- $\mathbf{A}_r = (\mathbf{Z}^T \mathbf{V})^{-1} \mathbf{Z}^T \mathbf{A} \mathbf{V}$
- $\mathbf{b}_r = (\mathbf{Z}^T \mathbf{V})^{-1} \mathbf{Z}^T \mathbf{b}$
- $\mathbf{c}_r = \mathbf{c} \mathbf{V}$

Choosing $\sigma_i = -\hat{\lambda}_i$ for each reduced pole would interpolate $G(s)$ and $G'(s)$ and thus satisfy the optimality conditions. Because $\hat{\lambda}_i$ are unknown however, we must use an iterative method to them.

4.1 Iterative Rational Krylov Algorithm (IRKA)

In his paper, Gugercin [3] proposes an iterative algorithm in projection framework that converges to the optimal reduced poles. One first chooses r initial shift values, $\{\sigma\}_{i=1}^r$, and constructs a reduced system using the method outline above. One can then easily determine the poles of the reduced system $G_r(s) = \mathbf{c}_r^T (s \mathbf{I} \mathbf{A}_r) \mathbf{b}_r$. IRKA then sets the new shift values at the negative poles of the reduced system. By repeating this process, the shifts converge to the values of the negative poles of the optimally reduced system. This results in a reduced system which interpolates the original transfer thus satisfying the conditions for an optimal solution.

As a final note, we point out that this method is a fixed point iteration in the projection framework. That is, given an initial guess of the shifts, the algorithm produces a new shift selection and stops when the algorithm converges to a fixed point. From a theoretical standpoint, this algorithm should converge linearly.

5 Newton Iteration Scheme

In contrast to the projection framework outlined by Gugercin et al. we approached the model reduction problem from a pole-residue framework. We formulate the \mathcal{H}_2 optimal reduction scheme as a root finding problem and solve it using Newtons method of iteration.

5.1 A Root Finding Problem

Again, recall that to achieve the \mathcal{H}_2 optimality, we need the following conditions to hold: $G(-\hat{\lambda}_i) = G_r(-\hat{\lambda}_i)$ and $G'(-\hat{\lambda}_i) = G'_r(-\hat{\lambda}_i)$

We therefore define a function $\mathbf{F}(\mathbf{Q}) : \mathbb{C}^{2r} \rightarrow \mathbb{C}^{2r}$ by $\mathbf{F}(\mathbf{Q}) = \begin{cases} G(-\hat{\lambda}_i) - G_r(-\hat{\lambda}_i) & \text{if } 1 \leq i \leq r \\ G'(-\hat{\lambda}_i) - G'_r(-\hat{\lambda}_i) & \text{if } (r+1) \leq i \leq 2r \end{cases}$

Clearly, when $\mathbf{F} = \mathbf{0}$, the reduced transfer function satisfies the optimality conditions. For a given dynamical system, $G(s)$ and $G_r(s)$ are fixed. The function $\mathbf{F}(\mathbf{Q})$ takes in as variables the reduced poles and the reduced residues:

$$\mathbf{Q}_i = \begin{cases} \hat{\phi}_i & \text{if } 1 \leq i \leq r \\ \hat{\lambda}_{i-r} & \text{if } (r+1) \leq i \leq 2r \end{cases}$$

5.2 Newton Step

The Newton iteration step on \mathbf{Q} is:

$$\mathbf{Q}^{(i+1)} = \mathbf{Q}^{(i)} - (\mathbf{J}^{(i)})^{-1} \mathbf{F}^{(i)}$$

Theorem 1

$$\mathbf{J} = \begin{bmatrix} \frac{\partial G}{\partial \hat{\phi}} - \frac{\partial G_r}{\partial \hat{\phi}} & \left| \frac{\partial G}{\partial \hat{\lambda}} - \frac{\partial G_r}{\partial \hat{\lambda}} \right. \\ \frac{\partial G'}{\partial \hat{\phi}} - \frac{\partial G'_r}{\partial \hat{\phi}} & \left| \frac{\partial G'}{\partial \hat{\lambda}} - \frac{\partial G'_r}{\partial \hat{\lambda}} \right. \end{bmatrix}$$

where

$$\begin{aligned} \frac{\partial G(-\hat{\lambda}_i)}{\partial \hat{\phi}_j} &= 0 \\ \frac{\partial G_r(-\hat{\lambda}_i)}{\partial \hat{\phi}_j} &= \frac{-1}{\hat{\lambda}_i + \hat{\lambda}_j} \\ \frac{\partial G'(-\hat{\lambda}_i)}{\partial \hat{\phi}_j} &= 0 \\ \frac{\partial G'_r(-\hat{\lambda}_i)}{\partial \hat{\phi}_j} &= \frac{-1}{(\hat{\lambda}_i + \hat{\lambda}_j)^2} \\ \frac{\partial G(-\hat{\lambda}_i)}{\partial \hat{\lambda}_j} &= \begin{cases} 0 & \text{if } i \neq j \\ \sum_{k=1}^n \frac{\hat{\phi}_k}{(\hat{\lambda}_j + \hat{\lambda}_k)^2} = -G'(-\hat{\lambda}_i) & \text{if } i = j \end{cases} \\ \frac{\partial G_r(-\hat{\lambda}_i)}{\partial \hat{\lambda}_j} &= \begin{cases} \frac{\hat{\phi}_j}{(\hat{\lambda}_j + \hat{\lambda}_k)^2} & \text{if } i \neq j \\ \frac{\hat{\phi}_i}{2\hat{\lambda}_i^2} + \sum_{\substack{k=1 \\ k \neq i}}^r \frac{\hat{\phi}_k}{(\hat{\lambda}_i + \hat{\lambda}_k)^2} = \frac{\hat{\phi}_i}{4\hat{\lambda}_i^2} - G'_r(-\hat{\lambda}_i) & \text{if } i = j \end{cases} \\ \frac{\partial G'(-\hat{\lambda}_i)}{\partial \hat{\lambda}_j} &= \begin{cases} 0 & \text{if } i \neq j \\ \sum_{k=1}^n \frac{2\hat{\phi}_k}{(\hat{\lambda}_j + \hat{\lambda}_k)^3} = -G''(-\hat{\lambda}) & \text{if } i = j \end{cases} \\ \frac{\partial G'_r(-\hat{\lambda}_i)}{\partial \hat{\lambda}_j} &= \begin{cases} \frac{2\hat{\phi}_j}{(\hat{\lambda}_i + \hat{\lambda}_k)^3} & \text{if } i \neq j \\ \frac{\hat{\phi}_j}{2\hat{\lambda}_i^3} + \sum_{\substack{k=1 \\ k \neq j}}^r \frac{2\hat{\phi}_k}{(\hat{\lambda}_i + \hat{\lambda}_k)^3} = \frac{2\hat{\phi}_i}{4\hat{\lambda}_i^3} - G''_r(-\hat{\lambda}_i) & \text{if } i = j \end{cases} \end{aligned}$$

We quickly found that convergence was not obtained in most cases, and so modified the approach.

6 Modified Newton Framework

6.1 A Reformulation

Having abandoned iteration on both poles and residues, we decided to attempt an iteration on poles alone. We use the work in Gaier's paper [2] who showed that, given fixed poles $\lambda_1 \dots \lambda_r$, we can calculate the optimal reduced residues, $\sigma_1 \dots \sigma_r$, by solving the following linear system:

$$\begin{bmatrix} \frac{-1}{\hat{\lambda}_1 + \lambda_1} & \cdots & \frac{-1}{\hat{\lambda}_1 + \lambda_r} \\ \vdots & & \vdots \\ \frac{-1}{\hat{\lambda}_r + \lambda_1} & \cdots & \frac{-1}{\hat{\lambda}_r + \lambda_r} \end{bmatrix} \begin{bmatrix} \hat{\phi}_1 \\ \vdots \\ \hat{\phi}_r \end{bmatrix} = \begin{bmatrix} G(-\hat{\lambda}_1) \\ \vdots \\ G(-\hat{\lambda}_r) \end{bmatrix}$$

Proposition 1 *The condition $G(-\hat{\lambda}) = G_r(-\hat{\lambda})$ is satisfied if $\{\hat{\phi}_i\}_i^r$ are determined in this way.*

Proof 1 *This follows directly from the definition of*

$$G_r(s) = \sum_{k=1}^r \frac{\hat{\phi}_k}{s - \hat{\lambda}_k}$$

with s evaluated at $-\hat{\lambda}_i$

6.2 Algorithm

Define $\mathbf{H} : \mathbb{C}^r \rightarrow \mathbb{C}^r$ as follows:

$$\mathbf{H}(\mathbf{Q}) = G'(-\hat{\lambda}) - G'_r(-\hat{\lambda})$$

with

$$\mathbf{Q}_j = \hat{\lambda}_j$$

Finding a zero of this function will ensure that $G'(-\hat{\lambda}_i) = G'_r(-\hat{\lambda}_i)$ for each reduced pole, satisfying the second optimality condition. We simultaneously force $G(-\hat{\lambda}_i) = G_r(-\hat{\lambda}_i)$ by our choice of the reduced residues as in proposition.

Note: the derivative of \mathbf{H} will be the lower right $r \times r$ matrix of the Jacobian of \mathbf{F} shown earlier.

$$\mathbf{J}_H = \frac{\partial G'}{\partial \hat{\lambda}} - \frac{\partial G'_r}{\partial \hat{\lambda}}$$

The proposed algorithm proceeds as follows:

1. Make an initial selection of the $\hat{\lambda}_i$'s and populate \mathbf{Q}
2. Find optimal residues as shown in proposition.
3. Calculate \mathbf{H} and its Jacobian, \mathbf{J}_H
- 4.

$$\mathbf{Q}^{(i+1)} = \mathbf{Q}^{(i)} - (\mathbf{J}_H^{(i)})^{-1} \mathbf{H}^{(i)}$$

5. Repeat steps 2 - 4 until convergence is obtained

6.3 Results

While this algorithm obtained convergence to the same solutions produced by IRKA, it took many more iterations, changing the poles slowly as it approached the zero.

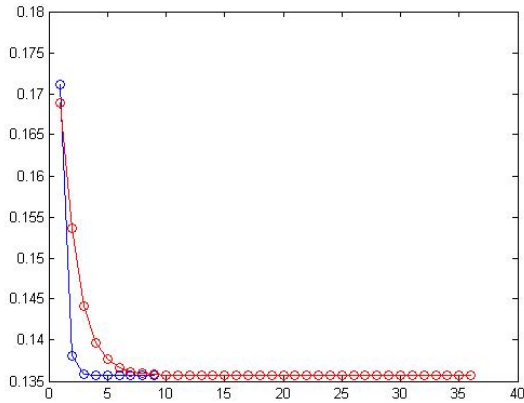


Figure 1: A comparison of the \mathcal{H}_2 Error at each iteration of IRKA (blue) and SISO Newton (red)

We believe the comparatively slow convergence to be the result of a wide flat spot in \mathbf{H} found in nearly every transfer function we examined. The following shows the values of \mathbf{H} for a system reduced to a single pole-residue term as $\hat{\lambda}$ is varied:

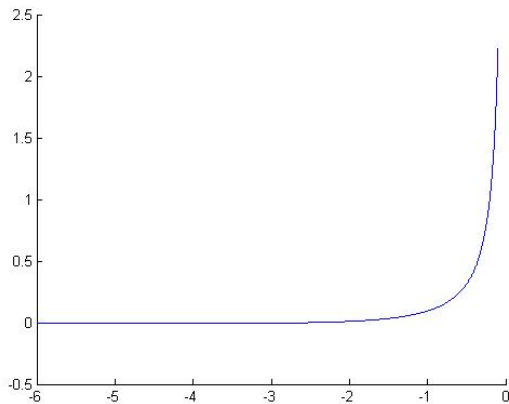


Figure 2: $\|\mathbf{F}\|$ on the interval $(-6, -1)$

We believe the fast rate of convergence seen in IRKA relates to the shape of its function, shown below:

We can see that nearly any choice of the initial shift quickly puts us in the neighborhood of the fixed point and often skips right over troublesome areas.

7 Conclusions

We have examined two frameworks for the optimal \mathcal{H}_2 model reduction problem. The first is a fixed point iteration from the framework of shifting interpolation points and the second is a Newton iteration from the pole-residue framework. We see that the optimal \mathcal{H}_2 model reduction problem is not well suited to be solved using a Newton step iteration in the pole-residue framework. These algorithms all converge much slower than the iterative rational Krylov algorithm. However, the optimal \mathcal{H}_2 model reduction problem seems well suited to be solved by the fixed point method IRKA. This leads us to conclude that IRKA and the projection framework is likely a better choice when solving the model reduction problem.

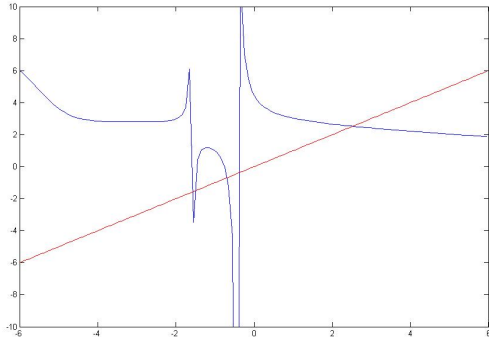


Figure 3: IRKA fixed point function on the shift interval $(-6, 6)$

7.1 Future Efforts

Attempts were made to reformulate the problem by defining $\mathbf{F}(\mathbf{Q}) = \frac{1}{G'(-\hat{\lambda}_i)} - \frac{1}{G'_i(-\hat{\lambda}_i)}$ as well as using Halley's method to speed convergence. Neither of these approaches proved immediately effective. It is, however, unfair to compare accelerated techniques to IRKA, which can be accelerated as well.

References

- [1] A.C. Antoulas, *Approximation of Large-Scale Dynamical Systems (Advances in Design and Control)(Advances in Design and Control)*, Society for Industrial and Applied Mathematics Philadelphia, PA, USA, 2005,
- [2] D. Gaier, *Lectures on Complex Approximation*, Birkhäuser, 1987.
- [3] S. Gugercin, A.C. Antoulas and C.A. Beattie, \mathcal{H}_2 model reduction for large-scale linear dynamical systems. *SIAM Journal on Matrix Analysis and Applications* Vol. 30, Issue: 2, pp. 609-638, 2008.