University of Groningen

Department of Mathematics

RuG

WORST CASE SYSTEM IDENTIFICATION

by

Johannes Ruurd Hoekstra

November 25, 1993

WORST CASE SYSTEM IDENTIFICATION

by

Johannes Ruurd Hoekstra

November 25, 1993

Preface

This paper gives an impression of the work I did on this subject. Beside the work shown in this paper there was resurch on different types of noise inputs. Furthermore, the *Mathlab* program in chapter 5 was not able to make a success of this algorithm, the computer speed and memory was to limited for all the calculations.
I would like to thank Dr. Trentelman for all the support he gave me and the patience he had.

Yours sincerely
Hans-Ruurd

Contents

Chapter 1

Problem formulation

1.1 Introduction

In this paper we are looking at an identification process.
That means there exists a certain plant of which we don't know much. We would like to know more about this plant, so it will be possible to control it. For building a good controller we need the transfer function of the plant. The algorithm provided in this paper gives an approximation of this transfer function. The approximation is build up as follows.
We begin with n starting points (inputs for our unknown system), which are uniformly spread over the complex unit circle: $z_k = e^{i\theta_k}$, $\theta_k = 2\pi k/n$ for $k = 0, \ldots, n-1$. Measuring the output we collect n point samples of this system: $E_0, \ldots, E_{n-1}$. Using linear interpolation we get a piecewise linear function which contains our n point samples. We design a continuous function which is an approximation of our piecewise linear function, we do so by using Fourier series. This continuous function on the complex unit circle is now extended to the complex unit disk. This extension is used as an approximation of our unknown transfer function.

1.2 Problem formulation

The H_∞ identification problem from [1] is formulated as follows.
Starting with an unknown plant, we will call it $\hat{h}$ in the remainder of this paper, the goal is to identify (make a good approximation of) the transfer function of the unknown plant $\hat{h}$.
We are looking in this paper at stable single-input single-output systems described by the form $y = h * u$ where u and y denote, respectively, the system's input and output and h their transfer function. This convolution ($y = h * u$) is defined by

$$y(t) = \sum_{t'=-\infty}^{\infty} h(t-t')u(t')$$

where $h(k) = 0$ for $k < 0$ and $u(t') = e^{iwt'}$.

$$
\begin{aligned}
y(t) &= \sum_{t'=-\infty}^{\infty} h(t-t')e^{iwt'} \\
&\quad \text{substitute } k = t - t' \\
&= \sum_{k=-\infty}^{\infty} h(k)e^{iw(t-k)} \\
&= e^{iwt} \sum_{k=-\infty}^{\infty} h(k)e^{-iwk} \\
&= e^{iwt} \sum_{k=0}^{\infty} h(k)e^{-iwk}
\end{aligned}
$$

$\sum_{k=0}^{\infty} h(k)z^{-k}$ is the Z-transform of the sequence $h(k)$. The corresponding transfer function $\hat{h}$, is described by

$$\hat{h}(z) = \sum_{k=0}^{\infty} h(k)z^{k}$$

$\hat{h}$ defined above denote the Z-Transform of h evaluated at $1/z$. Then $y(t) = e^{iwt}\hat{h}(e^{-iw})$. This allows us to define stability in terms of analyticity on a disk (rather than the complement of a disk) while at the same time leaving the unit circle invariant.
Unfortunately noise or measurement errors occur on every input-output system, these errors will be called noise. The output y of our system depends not only on u but also on the noise w. We need to make some assumptions on our plant $\hat{h}$ and our noise w to continue:

Assume:

- Plant $\hat{h} \in H_+$ where
  $H_+ := \cup_{\rho>1} H_{\infty,\rho}$, $H_{\infty,\rho}$ is the set of all complex functions that are analytic on a disk D_ρ. If $\hat{h} \in H_+$ then $\hat{h}$ is analytic on $|z| < \rho$ with $\rho > 1$.

- Noise $w \in l_\infty$ where
  l_∞ denotes the normed space $\{f : Z_{+,0} \to C \mid \|f\|_\infty := \sup_{k \in Z_{+,0}} |f(k)| < \infty\}$ with $Z_{+,0} := \{k \in Z : k \geq 0\}$. This means that the noise w is bounded during the identification process.

Our transfer function $\hat{h}$ and the noise w must satisfy the assumptions made in *Assume*. So there must be a disk D_ρ with $\hat{h}$ bounded inside and a bound on the absolute value of the noise.

We also need to know the number of point samples(experimental data) (E_n) we start with. Normally the accuracy of the algorithm depends on the number of samples. Every point sample is just a measurement but is depending on the true transfer function of the plant $\hat{h}$ and the noise w. We will write $E_n(\hat{h}, w)$ instead of E_n.

Given:

- Plant information in the form of a pair $(\rho, M) \in (1, \infty] \times [0, \infty)$ for which it is known that $\hat{h} \in \bar{B}H_{\infty,\rho}(M) := \hat{h} \in \{x \in H_{\infty,\rho} : \|x\|_{H_{\infty,\rho}} \leq M\}$. $\bar{B}H_{\infty,\rho}(M)$ is the set of all $\hat{h}$ which are analytic and bounded by M inside the circle D_ρ with radius ρ .

- A bound $\epsilon \in [0, \infty)$ on the absolute value of the noise w , i.e., $w \in \bar{B}l_\infty(\epsilon)$. $\bar{B}l_\infty(\epsilon) := \{w \in C \mid \|w\|_\infty \leq \epsilon\}$.

- For each information level $n \in Z_+$,
  E_n, holder of the experimental data ($\hat{h} \in H_+$ and noise $w \in l_\infty$), is defined by

$$E_n : H_+ \times l_\infty \to T_n l_\infty \tag{1.1}$$

  where

$$E_n(\hat{h}, w) := U_n \hat{h} + T_n w \tag{1.2}$$

  with $T_n : l_\infty \to l_\infty$ defined by $(T_n f)_k := f_k$ for $k = 0, 1, \ldots, n-1$ and $(T_n f)_k := 0$ for $k \geq n$.
  $U_n : H_+ \to T_n l_\infty$ defined by $(U_n f)_k := f(e^{2\pi i k/n})$ for $k = 0, 1, \ldots, n-1$ and $(U_n f)_k := 0$ for $k \geq n$.

The goal is to identify the transfer function of the unknown plant $\hat{h}$. During the identification proces errors will be made. These errors are formed by noise w and truncations in the algorithm. The algorithm we use must satisfy certain demands, if not, the approximation will not be useful.
The algorithm is useful if the error between the true transfer function and our approximation is small enough. We are using the H_∞ norm to measure this error. Minimizing the H_∞ norm $\|f\|_\infty = \sup_{z<1} |f(z)|$ means minimizing the maximum value(worst case). Minimizing the worst case identification error is certainly a good bound on the error. The error e_n depends on the information level n, our algorithm A_n and the values of ρ, M, ϵ. The notation we use is $e_n(A_n; \rho, M, \epsilon)$. The error is defined as the $sup\|\cdot\|_\infty$ of the difference between the true transfer function $\hat{h}$ and our approximation $A_n(E_n)$. The supremum is taken over $\hat{h} \in \bar{B}H_{\infty,\rho}$ and $w \in \bar{B}l_\infty(\epsilon)$, all possible $\hat{h}$ and w. An algorithm should converge to the true function if there is no noise($\epsilon \to 0$) and the number of steps converges to infinity($n \to \infty$). If $M \to 0$ and $\epsilon \to 0$ and ρ converges to infinity we have the zero function to which the algorithm should also converge.

Find:

- A plan of algorithms $A = \{A_n\}_{n \in Z_+}$ such that for each information level n, the algorithm $A_n : T_n l_\infty \to H_+$ maps the given experimental data into a transfer function estimate $A_n(E_n(\hat{h}, w)) \in H_+$ in such a way that the worst case identification error

$$e_n(A_n; \rho, M, \epsilon) := \sup_{\substack{\hat{h} \in \hat{B}H_{\infty,\rho} \\ w \in \hat{B}l_\infty(\epsilon)}} \|\hat{h} - A_n(E_n(\hat{h}, w))\|_\infty \tag{1.3}$$

  converges as follows

$$\lim_{\substack{\epsilon \to 0 \\ n \to \infty}} e_n(A_n; \rho, M, \epsilon) = 0 \tag{1.4}$$

  and

$$\lim_{\substack{\epsilon \to 0 \\ \rho \to \infty \\ M \to 0}} e_n(A_n; \rho, M, \epsilon) = 0 \tag{1.5}$$

- Derive explicit bounds on $e_n(A_n; \rho, M, \epsilon)$ as a function of ρ, M, ϵ and n.

Chapter 2

The Algorithm

The algorithm provided in this paper for the problem of identification in H_∞ has a two-step structure [1]. A pictorial representation of this structure is given in Fig. 2.1. The algorithm's goal is to identify the plant transfer function $\hat{h}$ using the given information $E_n(\hat{h}, w)$.

In the first step of the algorithm, n noisy point samples of the unknown stable plant are taken to compute a L_∞ approximation; n noisy point samples $\rightarrow$ lineair interpolation $\rightarrow$ Fourier series $\rightarrow$ truncated Fourier series $\rightarrow$ L_∞ approximation. The Fourier series are truncated to enable us to use standard finite- dimensional methods in step 2.

In the second step of the algorithm, this L_∞ approximation is mapped into a stable real-rational H_∞ approximation to the unknown stable plant. It is this H_∞ approximation which serves as the identified plant model. The H_∞ approximation is obtained by computing the best H_∞ approximation to the L_∞ approximation decribed above. This amounts to solving the so-called Nehari problem, and this task is carried out using the AAK approximation theory [2][6].

2.1 Step 1

In step 1 we only know the noisy point samples (experimental data) $E_n(\hat{h}, w)$. They are collected by taking n point samples uniformly spread over the complex unit circle. When we use lineair interpolation on these noisy point samples we get a piecewise lineair function. Calculating Fourier series of a function is easier when this function is (piecewise) lineair, that is why we use lineair interpolation. From [3, theorem 13.2d] we first calculate the Fourier coefficients of $E_n(\hat{h}, w)$ using Discrete Fourier Transform(DFT) (step 1a) and then multiply them by a factor depending on the kind of interpolation used (step 1b). This factor is in our case the Fourier Transform of the linear spline. So step 1 gives us the truncated linear spline Fourier series.

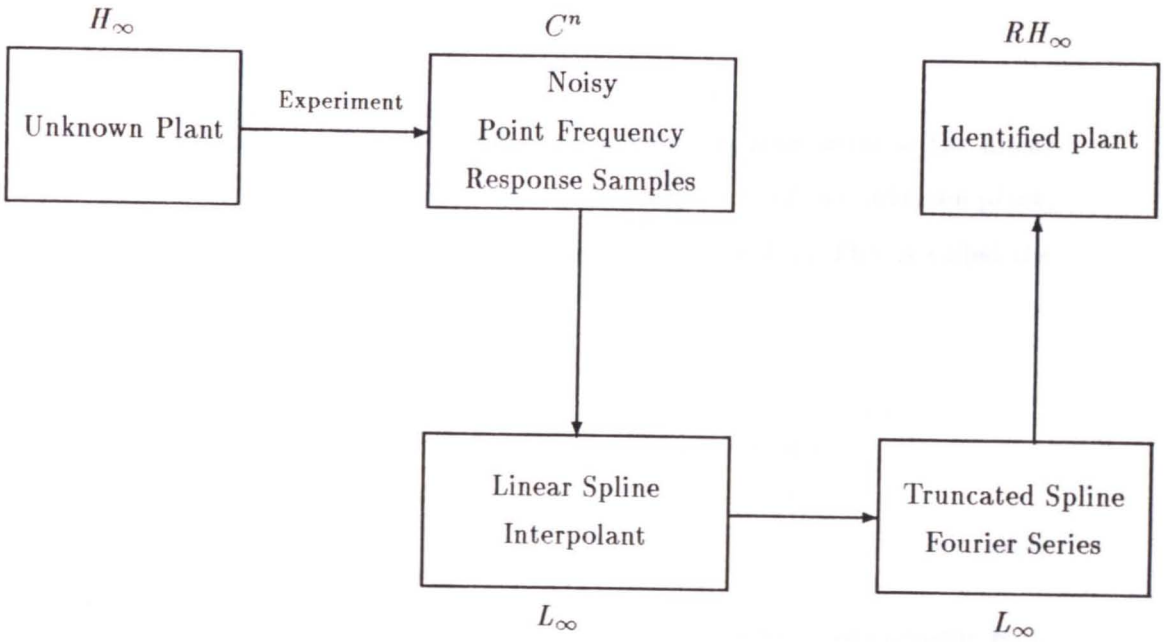


Figure 2.1: Two-step structure of the algorithm for identification in H_∞.

2.1.1 Step 1a:

Select a value for $N \in Z_+$, N is the truncation of the Fourier series. Calculate the DFT-coefficients $\hat{c}_k$ of the noisy point samples $E_n(\hat{h}, w)$ by:

$$\hat{c}_k = \frac{1}{n} \sum_{m=0}^{n-1} (E_n(\hat{h}, w))_m e^{-2\pi i k m/n}, \quad -N \leq k \leq N \tag{2.1}$$

2.1.2 Step 1b:

Attenuate the DFT coefficients $\hat{c}_k$ using the attenuation factors τ_k as follows:

$$c_k = \tau_k \hat{c}_k, \quad -N \leq k \leq N \tag{2.2}$$

where $\tau_0 = 1$ and

$$\tau_k = \left(\frac{n}{\pi k}\right)^2 \left(\sin\frac{\pi k}{n}\right)^2, \quad k \neq 0$$

This attenuation factor τ_k depends on the kind of interpolation which is used on the noisy point samples [3, theorem13.2d]. τ_k from (2.2) is the attenuation factor which corresponds to the linear interpolation. The Fourier coefficients of the linear spline are given by c_k.

2.2 Step 2

In step 2 we start with the coefficients c_k of the truncated Fourier series of the linear spline (2.2) which forms the L_∞ approximation $\varphi(z) = \sum_{k=-N}^{N} c_k z^k$ of our unknown plant. Now we are looking for the best approximation $g \in H_\infty$ to $\varphi \in L_\infty$. This is called the Nehari problem. From [6, §15.3] we see that:

$$g = \varphi - \frac{H_\varphi f}{f}$$

with $H_\varphi f = P_-(\varphi f)$ where P_- is the orthogonal projection operator:

$$P_-\left(\sum_{k=-\infty}^{\infty} c_k z^k\right) = \sum_{k=-\infty}^{-1} c_k z^k \tag{2.3}$$

and f a singular vector of H_φ corresponding to $s_k(H_\varphi)$ (singular value of operator H_φ). The Nehari problem is solved, in two steps, with the AAK approximation theory [2][6, §16.3]

2.2.1 Step 2a:

Use the negative coefficients $(c_k)_{k=-1}^{-N}$ of (2.2) to form the Hankel matrix of H_φ

$$H = \begin{pmatrix} c_{-1} & c_{-2} & \dots & c_{-N} \\ c_{-2} & c_{-3} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ c_{-N} & 0 & \dots & 0 \end{pmatrix} \tag{2.4}$$

and for this matrix obtain the maximum singular value $\bar{\sigma}$ and the corresponding right and left singular vectors $r = [r_1 r_2 \dots r_N]^t, s = [s_1 s_2 \dots s_N]^t$.

2.2.2 Step 2b:

Using these quantities, form the identified model

$$A_n^N(E_n(\hat{h}, w))(z) = \sum_{k=-N}^{N} c_k z^k - \frac{\bar{\sigma} \sum_{k=0}^{N-1} s_{N-k} z^k}{z^N \sum_{k=0}^{N-1} r_{k+1} z^k} \tag{2.5}$$

As a result of the Singular Value Decomposition (SVD) required in (2.4) and the manner in which the model is formed in (2.5), each algorithm A_n^N is a nonlinear function of the information $E_n(\hat{h}, w)$.

In forming a plan of algorithms based on A_n^N we wish to admit the possibility that for each information level n a different value of the parameter N may be selected. Correspondingly, let $N(\cdot) : Z_+ \to Z_+$ denote a particular parameter sequence, and then define an associated plan of algorithms as follows:

$$A^{N(\cdot)} := \{A_n^{N(n)}\}_{n \in Z_+}$$

Chapter 3

Error bound & proof

3.1 Error bound

The global error properties of A_n^N (2.5) as defined in (1.3) are given by:

$$e_n(A_n^N;\rho,M,\epsilon) \leq 2\min\left\{\frac{4M\pi}{(\rho-1)}\cdot\frac{1}{n}\,,\ \frac{M\pi^2(\rho+1)}{(\rho-1)^2}\cdot\frac{1}{n^2}\right\} + \frac{4(M+\epsilon)}{\pi^2}\cdot\frac{n^2}{N} + 2\epsilon \quad (3.1)$$

This is, what we where looking for, an explicit bound on $e_n(A_n^N;\rho,M,\epsilon)$ as a function of ρ, M, ϵ and n. And as required in (1.4) and (1.5) they converge to 0 by suitably choosing $N(n)$.

$$\begin{aligned}
&\lim_{\substack{\epsilon\to 0\\ n\to\infty}} e_n(A_n^N;\rho,M,\epsilon)\\
&= \lim_{\substack{\epsilon\to 0\\ n\to\infty}} \left(2\min\left\{\frac{4M\pi}{(\rho-1)}\cdot\frac{1}{n}\,,\ \frac{M\pi^2(\rho+1)}{(\rho-1)^2}\cdot\frac{1}{n^2}\right\} + \frac{4(M+\epsilon)}{\pi^2}\cdot\frac{n^2}{N} + 2\epsilon\right)\\
&= \lim_{\substack{\epsilon\to 0\\ n\to\infty}} \left(2\min\left\{\frac{4M\pi}{(\rho-1)}\cdot\frac{1}{n}\,,\ \frac{M\pi^2(\rho+1)}{(\rho-1)^2}\cdot\frac{1}{n^2}\right\} + \frac{4(M+\epsilon)}{\pi^2}\cdot\frac{n^2}{N}\right) + \lim_{\epsilon\to 0} 2\epsilon
\end{aligned}$$

and

$$\begin{aligned}
&\lim_{\substack{\epsilon\to 0\\ \rho\to\infty\\ M\to 0}} e_n(A_n^N;\rho,M,\epsilon)\\
&= \lim_{\substack{\epsilon\to 0\\ \rho\to\infty\\ M\to 0}} \left(2\min\left\{\frac{4M\pi}{(\rho-1)}\cdot\frac{1}{n}\,,\ \frac{M\pi^2(\rho+1)}{(\rho-1)^2}\cdot\frac{1}{n^2}\right\} + \frac{4(M+\epsilon)}{\pi^2}\cdot\frac{n^2}{N} + 2\epsilon\right)\\
&= \lim_{\substack{\rho\to\infty\\ M\to 0}} 2\min\left\{\frac{4M\pi}{(\rho-1)}\cdot\frac{1}{n}\,,\ \frac{M\pi^2(\rho+1)}{(\rho-1)^2}\cdot\frac{1}{n^2}\right\} + \lim_{\substack{\epsilon\to 0\\ M\to 0}} \frac{4(M+\epsilon)}{\pi^2}\cdot\frac{n^2}{N} + \lim_{\epsilon\to 0} 2\epsilon
\end{aligned}$$

As a consequence of the form of this error bound (3.1), it is easy to see that if $N(\cdot)$ is chosen so that

$$\lim_{n\to\infty} \frac{n^2}{N(n)} = 0 \tag{3.2}$$

then $A_n^{N(n)}$ is a convergent plan of algorithms.
For example, if we chose $N(n) = n^3$ then the global identification error becomes

$$e_n(A_n^N; \rho, M, \epsilon) \;=\; 2\epsilon \;+\; O\left\{\frac{1}{n}\right\} \text{ as } n \to \infty$$

3.2 Proof

3.2.1 L_∞ Approximation from noisy point samples

In this subsection we will calculate the contribution of the error made in *Step* 1. The proof follows the following steps:

1. In definition (3.1) and fact (3.2) we define a piecewise linear function.
2. In fact (3.3) we find the maximal error between the "original function" and the piecewise linear function in terms of the derivatives.
3. In fact (3.4) we calculate the derivatives, making use of Cauchy's formula for derivatives.
4. In lemma (3.5) we substitude the above 3 in each other.
5. In fact (3.6) we describe the truncated linear spline Fourier series and the attanuation factors.
6. In theorem (3.7) we calculate the error made by the truncation of the Fourier series and give a complete maximal error made in *Step* 1.

Definition 3.1 *Let $f \in l_\infty \equiv (\{f : Z_{+,0} \to C \mid \|f\|_\infty := \sup_{k\in Z_{+,0}} |f(k)| < \infty\})$ and $n \in Z_+$. The linear spline which interpolates the first n components of f_k at the points: $z_k = e^{i\theta_k}$, $\theta_k = 2\pi k/n$, $k = 0, \ldots, n-1$ is the function $g = S_n f \in L_\infty \equiv \{g : \delta D \to C \mid$ g is measurable and $\|g\|_\infty := esssup_{z\in\delta D}|g(z)| < \infty\}$ given by:*

$$g(e^{i\theta}) = \begin{cases} f_k + (\theta - \theta_k)\left(\dfrac{f_k - f_{k+1}}{\theta_k - \theta_{k+1}}\right) & \theta_k \le \theta \le \theta_{k+1} \quad k = 0, \ldots, n-2 \\ f_{n-1} + (\theta - \theta_{n-1})\left(\dfrac{f_{n-1} - f_0}{\theta_{n-1} - 2\pi}\right) & \theta_{n-1} \le \theta \le 2\pi \end{cases} \tag{3.3}$$

Fact 3.2 *For each $n \in Z_+$, the operator $S_n : l_\infty \to L_\infty$ defined above is linear and has induced norm $\|S_n\| = \sup_{\|f\|_\infty=1} \|S_n f\|_\infty = 1$*

Proof:
S_n linear : The values f_k appear linearly in the expression given for $g(e^{i\theta})$ in def. (3.1).
$\|S_n\| = 1$: Let $f \in l_\infty$ with $\|f\|_\infty = 1$ i.e. $\|f\|_\infty := \sup_{k\in Z_+} |f_k| = 1$. $g(e^{i\theta})$ interpolates f_k thus: $|g(e^{i\theta})| \leq \max_{k=0,\ldots,n-1}\{f_k\} \leq 1$ for $\theta \in [0, 2\pi]$. $g = S_n f \;\Rightarrow\; \|S_n f\|_\infty \leq 1$ (in L_∞).
Equality is obtained by considering the sequence $f \in T_n l_\infty$ when
$f_k = 1,\ k = 0, \ldots, n-1,\quad f_k = 0,\ k \geq n.$

Fact 3.3 *Let $n \in Z_+$, S_n defined as in definition(3.3) and $(U_n f)_k := f(e^{2\pi ik/n})$ for $k = 0, 1, \ldots, n-1$ and $(U_n f)_k := 0$ for $k \geq n$. If $f \in H_+$, then:*

$$\|f - S_n U_n f\|_\infty \leq \min\left\{ \frac{4\pi}{n}\|f'\|_\infty \,,\ \left(\frac{\pi}{n}\right)^2 \left(\|f'\|_\infty + \|f''\|_\infty\right) \right\} \tag{3.4}$$

Proof:
$f \in H_+$ and $f = R + iI$
From [4, Theorem 6.15] : $d[f, g(\Delta)]_\infty \leq \bar{\Delta}\|f'\|_\infty$ with $\Delta = e^{i\theta}$ and $d[f, g(\Delta)]_\infty$ is the maximum distance between f and $g(e^{i\theta})$ (3.1) and
$\bar{\Delta} := \max_{0\leq k\leq n-1}(z_{k+1} - z_k) \;\leq\; \frac{2\pi}{n}\ \ (z_{k+1} - z_k \;=\; e^{i\theta_{k+1}} - e^{i\theta_k} \;\leq\; \frac{2\pi}{n})$
The maximal error, when using Linear Spline Interpolation, is the maximal derivative of f times the maximal distance between two coefficients of f_k.
$\Longrightarrow$

$$\|R - S_n U_n R\|_\infty \;\leq\; \frac{2\pi}{n}\|R'\|_\infty \;\leq\; \frac{2\pi}{n}\|f'\|_\infty$$

and

$$\|I - S_n U_n I\|_\infty \;\leq\; \frac{2\pi}{n}\|I'\|_\infty \;\leq\; \frac{2\pi}{n}\|f'\|_\infty$$

$$\begin{aligned}
\|f - S_n U_n f\|_\infty \;&=\; \|R + iI - S_n U_n(R + iI)\|_\infty \\
&\quad\ \text{Since } U_n f = U_n R + iU_n I \text{ and by the fact that} \\
&\quad\ S_n \text{ is linear, this is equal to} \\
&=\; \|R - S_n U_n R \;+\; iI - iS_n U_n I\|_\infty \\
&\leq\; \|R - S_n U_n R\|_\infty \;+\; \|I - S_n U_n I\|_\infty \\
&\leq\; \frac{4\pi}{n}\|f'\|_\infty
\end{aligned}$$

From [4, Theorem 6.15] : $d[f, g(\Delta)]_\infty \leq \frac{\bar{\Delta}^2}{8}\|f''\|_\infty$ with $\Delta = e^{i\theta}$ and $\bar{\Delta} := \max_{0\leq k\leq n-1}(z_{k+1} - z_k) \;\leq\; \frac{2\pi}{n}$
Thus,

$$\|R - S_n U_n R\|_\infty \;\leq\; \frac{(\frac{2\pi}{n})^2}{8}\|R''\|_\infty \;\leq\; \frac{\pi^2}{2n^2}\|f''\|_\infty$$

and

$$\|I - S_n U_n I\|_\infty \leq \frac{(\frac{2\pi}{n})^2}{8}\|I''\|_\infty \leq \frac{\pi^2}{2n^2}\|f''\|_\infty$$

So we get

$$\|f - S_n U_n f\|_\infty \leq \frac{\pi^2}{n^2}\|f''\|_\infty \,. \tag{3.5}$$

Of course, this implies

$$\|f - S_n U_n f\|_\infty \leq \frac{\pi^2}{n^2}\left(\|f'\|_\infty + \|f''\|_\infty\right) \tag{3.6}$$

Why the writers of [1] have chosen for (3.6) instead of (3.5) is unclear.

Fact 3.4 *Let $(\rho, M) \in (1, \infty] \times [0, \infty)$ and $k \in Z_+$. If $f \in \bar{B}H_{\infty,\rho}(M)$, then*

$$\|f^{(k)}\|_\infty \leq \frac{k!M}{(\rho - 1)^k} \tag{3.7}$$

Proof:
$f \in \bar{B}H_{\infty,\rho}(M) \Rightarrow f$ is analytic and bounded by M inside the circle D_ρ.
$\Longrightarrow$ f is analytic in a disk containing $D_{\rho-\xi}$ for $0 \leq \xi \leq \rho$. Cauchy's formula for derivatives [5] gives that for any $z \in C$ with $|z| = 1$

$$f^{(k)}(z) = \frac{k!}{2\pi i}\int_\Gamma \frac{f(w)}{(w-z)^{k+1}}dw \qquad \xi < \rho - 1 \tag{3.8}$$

where $\Gamma = \{w \in C : |w - z| = \rho - \xi - 1\}$ with ξ small enough
(Γ inside $D_{\rho-\xi} \Rightarrow f$ analytic inside and on Γ and z lies inside Γ).
Since $f \in \bar{B}H_{\infty,\rho}(M)$ we have $\|f(z)\|_{H_{\infty,\rho}} \leq M$.
Hence

$$\begin{aligned} |f^{(k)}(z)| &\leq \left|\frac{k!}{2\pi i}\right| \frac{M \times (length\ \Gamma)}{|w-z|^{k+1}} \\ &\leq \frac{k!}{2\pi}\frac{M 2\pi(\rho - \xi - 1)}{(\rho - \xi - 1)^{k+1}} \\ &\leq \frac{k!M}{(\rho - \xi - 1)^k} \end{aligned}$$

The result follows by taking the supremum over all $|z| = 1$, and then the limit as $\xi \to 0$.

Lemma: 3.5 *Let $(\rho, M) \in (1, \infty] \times [0, \infty)$, $\epsilon \in [0, \infty)$ and $n \in Z_+$. If $(\hat{h}, w) \in \bar{B}H_{\infty,\rho}(M) \times \bar{B}l_\infty(\epsilon)$ then:*

$$\|\hat{h} - S_n(E_n(\hat{h}, w))\|_\infty \leq \min\left\{ \frac{4\pi M}{n(\rho - 1)}\,,\ \left(\frac{\pi}{n}\right)^2\left(\frac{M(\rho+1)}{(\rho-1)^2}\right)\right\} + \epsilon \tag{3.9}$$

Proof:
From (1.2), fact (3.2) and the triangle inequality

$$\begin{aligned}
\|\hat{h} - S_n(E_n(\hat{h}, w))\|_\infty &= \|\hat{h} - S_n(U_n\hat{h} + T_n w)\|_\infty \\
&= \|\hat{h} - S_n(U_n\hat{h}) - S_n(T_n w)\|_\infty \\
&\leq \|\hat{h} - S_n(U_n\hat{h})\|_\infty + \|S_n(T_n w)\|_\infty \\
&\leq \|\hat{h} - S_n(U_n\hat{h})\|_\infty + \epsilon
\end{aligned}$$

From fact (3.3)

$$\|\hat{h} - S_n(U_n\hat{h})\|_\infty \leq \min\left\{ \frac{4\pi M}{n}\|\hat{h}'\|_\infty \,,\, \left(\frac{\pi}{n}\right)^2 \left(\|\hat{h}'\|_\infty + \|\hat{h}''\|_\infty\right) \right\}$$

From fact (3.4)

$$\begin{aligned}
\|\hat{h} - S_n(U_n\hat{h})\|_\infty &\leq \min\left\{ \frac{4\pi}{n}\frac{M}{\rho - 1} \,,\, \left(\frac{\pi}{n}\right)^2 \left(\frac{M}{\rho - 1} + \frac{2M}{(\rho - 1)^2}\right) \right\} \\
&\leq \min\left\{ \frac{4\pi M}{n(\rho - 1)} \,,\, \left(\frac{\pi}{n}\right)^2 \left(\frac{M(\rho + 1)}{(\rho - 1)^2}\right) \right\}
\end{aligned}$$

$\Longrightarrow$

$$\|\hat{h} - S_n(E_n(\hat{h}, w))\|_\infty \leq \min\left\{ \frac{4\pi M}{n(\rho - 1)} \,,\, \left(\frac{\pi}{n}\right)^2 \left(\frac{M(\rho + 1)}{(\rho - 1)^2}\right) \right\} + \epsilon \tag{3.10}$$

We see that in the above error bound (3.10) from lemma (3.5) the terms due to n and ϵ are decoupled. This means that as $n \to \infty$ the linear spline approximation converges to 0 within the constant tolerance ϵ. It is not possible for this approximation to have a worst case uncertainty error less than the worst case measurement error at any point. ϵ is a lower bound on the optimal identification error.

We will show that truncation of the linear spline Fourier series also provides an L_∞ approximation to $\hat{h}$. This extension, which makes use of attenuation factors, provides a direct connection between the uniform samples $(E_n(\hat{h}, w))$ and the Fourier series coefficients of the corresponding linear spline interpolant (S_n) [3, theorem 13.2d].
The Fourier series coefficients of a function $f \in H_+$ are given by

$$c_k(f) := \frac{1}{2\pi}\int_0^{2\pi} f(e^{i\theta})e^{ik\theta}d\theta \qquad k \in Z$$

The DFT coefficients of the sample sequence $(U_n f)_m$ are given for each $n \in Z_+$ by

$$\hat{c}_k(U_n f) := \frac{1}{n}\sum_{m=0}^{n-1}(U_n f)_m e^{\frac{-2\pi ikm}{n}} \qquad k \in Z$$

But we have the linear spline interpolant (S_n) of $U_n f$ so we need $c_k(S_n U_n f)$.

Fact 3.6 *Let $n \in Z_+$. If $f \in H_+$, then for each $k \in Z$ $c_k(S_n U_n f) = \tau_k \hat{c}_k(U_n f)$ where*

$$\tau_k := \left(\frac{n}{\pi k}\right)^2 \left(\sin\frac{\pi k}{n}\right)^2, \quad k \neq 0 \quad \tau_0 := 1 \tag{3.11}$$

Proof:
In [3, Attennuation Factors] we find the proof of this fact and that the Fourier coefficients of the linear spline interpolation are (3.11).

Theorem: 3.7 *Let $(\rho, M) \in (1, \infty] \times [0, \infty)$, $\epsilon \in [0, \infty)$, $n \in Z_+$, $N \in Z_+$ and let $c_k, \hat{c}_k$, and τ_k be as defined above. If $(\hat{h}, w) \in \bar{B}H_{\infty,\rho}(M) \times \bar{B}l_\infty(\epsilon)$ then:*

$$\begin{aligned} F_{n,N}(E_n(\hat{h}, w))(e^{i\theta}) &:= \sum_{k=-N}^{N} c_k(S_n(E_n(\hat{h}, w)))e^{ik\theta} \\ &= \sum_{k=-N}^{N} \tau_k \hat{c}_k(E_n(\hat{h}, w))e^{ik\theta} \end{aligned} \tag{3.12}$$

and

$$\|\hat{h} - F_{n,N}(E_n(\hat{h}, w))\|_\infty \leq \min\left\{ \frac{4\pi M}{n(\rho - 1)}, \left(\frac{\pi}{n}\right)^2 \left(\frac{M(\rho + 1)}{(\rho - 1)^2}\right) \right\} + \frac{2(M + \epsilon)n^2}{N\pi^2} + \epsilon \tag{3.13}$$

Proof:
The first statement (3.12) follows directly from fact(3.6), the second statement is a little more complex. We start with the Fourier expansion of $S_n(E_n(\hat{h}, w))$ and the definition of $F_{n,N}$

$$\begin{aligned} S_n(E_n(\hat{h}, w))(e^{i\theta}) &= \sum_{k=-\infty}^{\infty} c_k(S_n(E_n(\hat{h}, w)))e^{ik\theta} \\ F_{n,N}(E_n(\hat{h}, w))(e^{i\theta}) &:= \sum_{k=-N}^{N} c_k(S_n(E_n(\hat{h}, w)))e^{ik\theta} \end{aligned}$$

$\Longrightarrow$

$$|S_n(E_n(\hat{h}, w))(e^{i\theta}) - F_{n,N}(E_n(\hat{h}, w))(e^{i\theta})| \leq \sum_{k=-\infty}^{-N-1} |c_k(S_n(E_n(\hat{h}, w)))| + \sum_{k=N+1}^{\infty} |c_k(S_n(E_n(\hat{h}, w)))|$$

$$\begin{aligned} |c_k(S_n(E_n(\hat{h}, w)))| &= |\tau_k \hat{c}_k(E_n(\hat{h}, w))| \\ &= |\tau_k \hat{c}_k(U_n \hat{h} + T_n w)| \\ &= |\tau_k(\hat{c}_k(U_n \hat{h}) + \hat{c}_k(T_n w))| \\ &\leq |\tau_k(M + \epsilon)| \\ &\leq \left(\frac{n}{\pi k}\right)^2 (M + \epsilon) \end{aligned}$$

$\Longrightarrow$

$$
\begin{aligned}
|S_n(E_n(\hat{h},w))(e^{i\theta}) \;-\; F_{n,N}(E_n(\hat{h},w))(e^{i\theta})| \;&\leq\; \left(\sum_{k=-\infty}^{-N-1}\frac{1}{k^2} \;+\; \sum_{k=N+1}^{\infty}\frac{1}{k^2}\right)\times\left(\frac{n}{\pi}\right)^2(M+\epsilon)\\
&\leq\; \left(\int_{-\infty}^{-N}\frac{1}{k^2}dk \;+\; \int_{N}^{\infty}\frac{1}{k^2}dk\right)\times\left(\frac{n}{\pi}\right)^2(M+\epsilon)\\
&\leq\; \left(\frac{-1}{k}\,|_{-\infty}^{-N} \;+\; \frac{-1}{k}\,|_{N}^{\infty}\right)\times\left(\frac{n}{\pi}\right)^2(M+\epsilon)\\
&\leq\; \left(\frac{1}{N} \;+\; \frac{1}{N}\right)\times\left(\frac{n}{\pi}\right)^2(M+\epsilon)\\
&\leq\; \frac{2(M+\epsilon)n^2}{N\pi^2}
\end{aligned}
$$

$$
\|\hat{h}-F_{n,N}(E_n(\hat{h},w))\|_\infty \;\leq\; \|\hat{h}-S_n(E_n(\hat{h},w))\|_\infty + \|S_n(E_n(\hat{h},w))(e^{i\theta})-F_{n,N}(E_n(\hat{h},w))(e^{i\theta})\|_\infty
$$

$\Longrightarrow$

$$
\|\hat{h}-F_{n,N}(E_n(\hat{h},w))\|_\infty \;\leq\; \min\left\{\frac{4\pi M}{n(\rho-1)}\,,\;\left(\frac{\pi}{n}\right)^2\left(\frac{M(\rho+1)}{(\rho-1)^2}\right)\right\} + \frac{2(M+\epsilon)n^2}{N\pi^2} + \epsilon \tag{3.14}
$$

This is the maximal error made in Step 1.

3.2.2 H_∞ Identification from L_∞ Approximation

In the second step of the algorithm A_n^N (2.4 , 2.5), the L_∞ approximation is mapped into a stable real rational H_∞ approximation to the plant. By L_∞ approximation we mean the truncated Fourier series of the piecewise linear approximation.

Corollary: 3.8 *Let $(\rho, M) \in (1,\infty] \times [0,\infty)$, $\epsilon \in [0,\infty)$ $n \in Z_+$ and $N \in Z_+$. If $(\hat{h}, w) \in \bar{B}H_{\infty,\rho}(M) \times \bar{B}l_\infty(\epsilon)$ then*

$$dist(F_{n,N}(E_n(\hat{h}, w))\,,\, H_\infty) \leq \min\left\{ \frac{4\pi M}{n(\rho-1)}\,,\, \left(\frac{\pi}{n}\right)^2 \left(\frac{M(\rho+1)}{(\rho-1)^2}\right) \right\} + \frac{2(M+\epsilon)n^2}{N\pi^2} + \epsilon \tag{3.15}$$

where $dist(a, B) := \inf_{b\in B}\|a-b\|_\infty$ is the distance between a and B and $F_{n,N}(E_n(\hat{h}, w))$ is defined as in (3.12).

Proof: Since $\hat{h} \in H_\infty$, (3.15) is a direct result of (3.14).

This result (3.15) is the reason why we take the best H_∞ approximation to the L_∞ function $F_{n,N}(E_n(\hat{h}, w))$ as the identified model, because $dist(F_{n.N}, H_{infty}) \leq dist(F_{n.N}, \hat{h})$ since $\hat{h} \in H_\infty$. The problem of finding the best H_∞ approximation to the L_∞ function is called the Nehari problem. This problem can be solved using the AAK approximation theory [2], a partial statement of this theory can be found in [6, §16.3]

Fact 3.9 *Let $\varphi \in L_\infty$. Define the Hankel operator $H_\varphi : H_2 \to L_2 \ominus H_2$ associated with φ as follows*

$$H_\varphi f = P_-(\varphi f)$$

where here P_- denotes the orthogonal projection (2.3) from L_2 to $L_2 \ominus H_2$.

Define a maximizing vector for H_φ as an element $f \in H_2$ satisfying; $\|H_\varphi f\|_2 = \|H_\varphi\|\|f\|_2$ Finally, suppose a maximizing vector f for H_φ exists, and define

$$\psi = \varphi - \frac{H_\varphi f}{f}$$

Under these conditions, $\psi \in H_\infty$ and $\|\varphi - \psi\|_\infty = dist(\varphi, H_\infty)$, ψ is the best approximation of φ in H_∞.

Applying fact (3.9) with $\varphi = F_{n,N}(E_n(\hat{h}, w))$ as defined in theorem (3.7), the identified model $A_n^N(E_n(\hat{h}, w))$ given in step 2b (2.5) is obtained, and the error properties are the same as in *Step* 1.

3.2.3 Error bound

The proof of the Error bound, described in the above two subsections is now complete. We see that, due to the properties of the algorithm, the error bound depends on the first step. In particular the choice of the piecewise linear function S_n, in the form of the attenuation factors (see Step 1), is responsible for the form of the Error bound. By changing the attenuation factors we can control the effects of the noise, in our experimental data $E_n(\hat{h}, w)$, on the identification.
The identification error is: $\leq \ \|\hat{h} - F_{n,N}(E_n(\hat{h}, w))\|_\infty \ + \ dist(F_{n,N}(E_n(\hat{h}, w)) \ , \ H_\infty)$
$\Longrightarrow$

$$e_n(A_n^N; \rho, M, \epsilon) \ \leq \ 2 \min \left\{ \frac{4M\pi}{(\rho - 1)} \cdot \frac{1}{n} \ , \ \frac{M\pi^2(\rho + 1)}{(\rho - 1)^2} \cdot \frac{1}{n^2} \right\} + \frac{4(M + \epsilon)}{\pi^2} \cdot \frac{n^2}{N} + 2\epsilon$$

3.3 Comment on the algorithm

- It is natural to require that the identified model $A_n^N(E_n(\hat{h},w))$ from (2.5) is real rational. This is only possible when the DFT-coefficients $\hat{c}_k$ are real. We can ensure this by requiring that (for n even)

$$(E_n(\hat{h},w)_{\frac{n}{2}+1+k})e^{-2\pi ikm/n} = [(E_n(\hat{h},w)_{\frac{n}{2}+1-k})e^{2\pi ikm/n}]^* \text{ for } k = 1,2,\ldots,\frac{n}{2}-1 \tag{3.16}$$

$\Longrightarrow$

$$E_n(\hat{h},w)_{\frac{n}{2}+1+k} = [E_n(\hat{h},w)_{\frac{n}{2}+1-k}]^* \text{ for } k = 1,2,\ldots,\frac{n}{2}-1$$

and $E_n(\hat{h},w)_1$, $E_n(\hat{h},w)_{\frac{n}{2}+1}$ being real. E_n must occur in conjugated pairs. Without loss of generality we will take the random noise from δD_ϵ instead of D_ϵ with $\epsilon \in [0,\infty)$. Now define

$$\epsilon \cdot \left(e^{\frac{2\pi i(R)_{\frac{n}{2}+1+k}}{n}} \right) \text{ for } 0 \leq (R)_{\frac{n}{2}+1+k} \leq n-1$$

as the random noise $w \in \bar{B}l_\infty(\epsilon)$ which occours at sample $\frac{n}{2}+1+k$
(1.2) $\Longrightarrow$

$$E_n(\hat{h},w)_{\frac{n}{2}+1+k} = \hat{h}\left(e^{\frac{2\pi i(\frac{n}{2}+1+k)}{n}}\right) + \epsilon \cdot \left(e^{\frac{2\pi i(R)_{\frac{n}{2}+1+k}}{n}}\right)$$

$$\begin{aligned} E_n(\hat{h},w)_{\frac{n}{2}+1-k} &= \hat{h}\left(e^{\frac{2\pi i(\frac{n}{2}+1-k)}{n}}\right) + \epsilon \cdot \left(e^{\frac{2\pi i(R)_{\frac{n}{2}+1-k}}{n}}\right) \\ &= \overline{\hat{h}\left(e^{\frac{2\pi i(\frac{n}{2}+1+k)}{n}}\right)} + \epsilon \cdot \left(e^{\frac{2\pi i(R)_{\frac{n}{2}+1-k}}{n}}\right) \end{aligned}$$

The DFT-coefficients $\hat{c}_k$ from (2.1) must be real !
$\Longrightarrow E_n(\hat{h},w)_{\frac{n}{2}+1+k} + E_n(\hat{h},w)_{\frac{n}{2}+1-k}$ must be real.
$\Longrightarrow \epsilon \cdot \left(e^{\frac{2\pi i(R)_{\frac{n}{2}+1+k}}{n}}\right)$ and $\epsilon \cdot \left(e^{\frac{2\pi i(R)_{\frac{n}{2}+1-k}}{n}}\right)$ must be complex conjugate.
$\Longrightarrow$ The ball of noise $\bar{B}l_\infty(\epsilon)$ should thus be taken those complex numbers which satisfy the (3.16) complex conjugate symmetry. The noise, which can effect our system, will be restricted on his randomness (complex conjugate). This is a necessary restriction on the random occurence of the noise, but isn't it to strong? Will the general noise on our system satisfy these restrictions? This question will not be answered in this paper.

- The identified model given in (2.5) corresponds to a $(2N-1)$th-order rational transfer function (the Mc. Millan degree of $A_n^N(E_n(\hat{h},w))(z)$ is $2N-1$). N causes the truncation of the Fourier series and is there by a influence upon the identification error. From (3.2) we see that N should be chosen large enough such that $\lim_{n\to\infty} \frac{n^2}{N(n)} = 0$. It might be advisable to use model reduction techniques on the transfer function to obtain a reduced-order H_∞ approximation.

Chapter 4

Recursive Algorithm

For the case where $n(m) = 2^m \quad m \in Z_{+,0}$, the frequencies corresponding to the set of samples $E_{n(m)}(\hat{h}, w)$ are also uniformly spread over the unit circle. If our approximation is not accurate enough, we have to take more experimental data. To reduce the number of calculations we want to make use of the approximation we already know. That is why we have chosen $n(m) = 2^m \Rightarrow n(m+1) = 2^{m+1}$, the new experimental data interleave exactly so that the a posteriori information for level $n(m+1)$ can be given recursively by adding $n(m)$ point frequency response estimates to those of $E_{n(m)}(\hat{h}, w)$. Thus it should be possible to redefine the Algorithm.

4.1 Step 1a

$$\hat{c}_k(m) = \frac{1}{2^m} \sum_{l=0}^{2^m-1} (E_{2^m}(\hat{h}, w))_l e^{\frac{-2\pi i k l}{2^m}} \qquad -N(m) \leq k \leq N(m)$$

$$\hat{c}_k(m+1) = \frac{1}{2^{m+1}} \sum_{l=0}^{2^{m+1}-1} (E_{2^{m+1}}(\hat{h}, w))_l e^{\frac{-2\pi i k l}{2^{m+1}}} \qquad -N(m+1) \leq k \leq N(m+1)$$

We will try to express $\hat{c}_k(m+1)$ in terms of $\hat{c}_k(m)$ on $-N(m) \leq k \leq N(m)$, outside this range $\hat{c}_k(m+1)$ stays the same.

$$\begin{aligned} \hat{c}_k(m+1) &= \frac{1}{2^{m+1}} \sum_{l=0}^{2^m-1} (E_{2^{m+1}}(\hat{h}, w))_{2l} e^{\frac{-2\pi i k 2l}{2^{m+1}}} \\ &+ \frac{1}{2^{m+1}} \sum_{l=0}^{2^m-1} (E_{2^{m+1}}(\hat{h}, w))_{2l+1} e^{\frac{-2\pi i k (2l+1)}{2^{m+1}}} \quad (:= A_k(m+1)) \end{aligned}$$

$$\hat{c}_k(m+1) = \begin{cases} \frac{1}{2}\hat{c}_k(m) + \frac{1}{2^{m+1}} \sum_{l=0}^{2^m-1} (E_{2^{m+1}}(\hat{h}, w))_{2l+1} e^{\frac{-2\pi i k(2l+1)}{2^{m+1}}} & -N(m) \leq k \leq N(m) \\ \frac{1}{2^{m+1}} \sum_{l=0}^{2^{m+1}-1} (E_{2^{m+1}}(\hat{h}, w))_l e^{\frac{-2\pi i k l}{2^{m+1}}} & N(m) \leq |k| \leq N(m+1) \end{cases} \tag{4.1}$$

4.2 step 1b

We were succesfull in redefining Step 1a into a recursive algorithm, now we have to do the same with Step 1b.

$$\tau_k(m) = \left(\frac{2^m}{\pi k}\right)^2 \left(\sin\frac{\pi k}{2^m}\right)^2 \qquad -N(m) \leq k \leq N(m) \; k \neq 0 \qquad \tau_0(m) = 1$$
$$\tau_k(m+1) = \left(\frac{2^{m+1}}{\pi k}\right)^2 \left(\sin\frac{\pi k}{2^{m+1}}\right)^2 \qquad -N(m+1) \leq k \leq N(m+1) \; k \neq 0 \quad \tau_0(m+1) = 1$$

We will try to formulate $\tau_k(m+1)$ in terms of $\tau_k(m)$ This is not possible in the same way as in Step 1a, but we can use the information of $\tau_k(m)$ in an other way.

$$\tau_0(m+1) = \tau_0(m) = 1$$

$$\tau_{2k}(m+1) = \left(\frac{2^{m+1}}{\pi 2k}\right)^2 \left(\sin\frac{\pi 2k}{2^{m+1}}\right)^2 = \left(\frac{2^m}{\pi k}\right)^2 \left(\sin\frac{\pi k}{2^m}\right)^2 = \tau_k(m) \qquad -N(m) \leq 2k \leq N(m$$
$$\tau_{2k+1}(m+1) = \left(\frac{2^{m+1}}{\pi(2k+1)}\right)^2 \left(\sin\frac{\pi(2k+1)}{2^{m+1}}\right)^2 \qquad -N(m) \leq 2k+1 \leq N(m$$
$$\tau_k(m+1) = \left(\frac{2^{m+1}}{\pi k}\right)^2 \left(\sin\frac{\pi k}{2^{m+1}}\right)^2 \qquad N(m) \leq |k| \leq N(m+1$$

We can combine Step 1a and Step 1b a little bit, we will do so on the interval $-N(m) \leq 2k \leq N(m) \;\; k \neq 0$

$$\tau_{2k}(m+1) = \tau_k(m) = \frac{c_k(m)}{\hat{c}_k(m)}$$
$$\hat{c}_{2k}(m+1) = \frac{1}{2}\hat{c}_{2k}(m) + A_{2k}(m+1)$$

$$c_{2k}(m+1) = \tau_{2k}(m+1) \cdot \hat{c}_{2k}(m+1) = \frac{1}{2}c_k(m) \cdot \frac{\hat{c}_{2k}(m)}{\hat{c}_k(m)} + \tau_{2k}(m+1) \cdot A_{2k}(m+1)$$
$$c_{2k+1}(m+1) = \tau_{2k+1}(m+1) \cdot \hat{c}_{2k+1}(m+1) + \tau_{2k+1}(m+1) \cdot A_{2k}(m+1)$$

This was not what we were looking for. If we want to redefine Step 1b we have to do it in the same way as Step 1a, because we have to combine them. The only alternative is to look at an other window function (attenuation factors). This new window function is more convenient to make a recursive algorithm.

$$\begin{aligned}
\tau_k(m) &= 1 - \tfrac{|k|}{N(m)} & -N(m) \le k \le N(m) \qquad \tau_k(m) = 0 \quad |k| \ge N(m) \\
\tau_k(m+1) &= 1 - \tfrac{|k|}{N(m+1)} & -N(m+1) \le k \le N(m+1) \quad \tau_k(m+1) = 0 \quad |k| \ge N(m+1)
\end{aligned}$$

Assume $N(m+1) = p \cdot N(m)$ with $p \ge 1$, we can do this without loss of generallity.

$$\begin{aligned}
\tau_k(m+1) &= 1 - \frac{|k|}{p \cdot N(m)} & -p \cdot N(m) \le k \le p \cdot N(m) \\
&= 1 - \frac{|k|}{N(m)} + \frac{(p-1)|k|}{p \cdot N(m)} & -p \cdot N(m) \le k \le p \cdot N(m) \\
&= \tau_k(m) + \frac{(p-1)|k|}{p \cdot N(m)} & -N(m) \le k \le N(m)
\end{aligned}$$

$$\tau_k(m+1) = \begin{cases} \tau_k(m) + \dfrac{(p-1)|k|}{p \cdot N(m)} & -N(m) \le k \le N(m) \\ 1 - \dfrac{|k|}{p \cdot N(m)} & -p \cdot N(m) \le k \le -N(m),\ N(m) \le k \le p \cdot N(m) \end{cases} \tag{4.2}$$

Combine Step 1a with this window function gives us:

$$c_k(m) = \tau_k(m) \cdot \hat{c}_k(m) \quad \Longrightarrow \quad \hat{c}_k(m) = \frac{c_k(m)}{\tau_k(m)}$$

$$\hat{c}_k(m+1) = \frac{1}{2}\hat{c}_k(m) + A_k(m+1)$$

We will look at the interval $-N(m) \le k \le N(m)$ because we can manipulate on it.

$$\begin{aligned}
c_k(m+1) &= \tau_k(m+1) \cdot \hat{c}_k(m+1) \\
&= \left(\tau_k(m) + \frac{(p-1)|k|}{p \cdot N(m)}\right)\left(\frac{1}{2}\hat{c}_k(m)\right) + \tau_k(m+1)A_k(m+1) \\
&= \frac{1}{2}\tau_k(m)\hat{c}_k(m) + \frac{1}{2}\frac{(p-1)|k|}{p \cdot N(m)}\hat{c}_k(m) + \tau_k(m+1)A_k(m+1) \\
&= \frac{1}{2}c_k(m) + \frac{1}{2}\frac{(p-1)|k|}{p \cdot N(m)}\frac{c_k(m)}{\tau_k(m)} + \tau_k(m+1)A_k(m+1) \\
&= \frac{1}{2}c_k(m)\left(1 + \frac{(p-1)|k|}{p \cdot N(m) \cdot \tau_k(m)}\right) + \tau_k(m+1)A_k(m+1)
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2}c_k(m)\left(1 + \frac{(p-1)|k|}{p\cdot N(m)\cdot\left(1-\frac{|k|}{N(m)}\right)}\right) + \tau_k(m+1)A_k(m+1) \\
&= \frac{1}{2}c_k(m)\cdot\left(\frac{p\cdot N(m)+|k|}{p\cdot N(m)+p|k|}\right) + \tau_k(m+1)A_k(m+1) \qquad (4.3)
\end{aligned}
$$

On $-p\times N(m)\le k\le -N(m)$, $N(m)\le k\le p\times N(m)$ everything stays the same.

$$
c_k(m+1) = \left(1-\frac{|k|}{p\times N(m)}\right)\frac{1}{2^{m+1}}\sum_{l=0}^{2^{m+1}-1}(E_{2^{m+1}}(\hat{h},w))_l e^{\frac{-2\pi ikl}{2^{m+1}}}
$$

For the beautiful case where $p=1$ the recursive step of Step 1 looks like this:

$$
\boldsymbol{c_k(m+1) = \frac{1}{2}c_k(m) + \tau_k(m+1)A_k(m+1)} \qquad (4.4)
$$

4.3 Step 2a

$$
H(m) = \begin{pmatrix} c_{-1}(m) & c_{-2}(m) & \dots & c_{-N(m)}(m) \\ c_{-2}(m) & c_{-3}(m) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ c_{-N(m)}(m) & 0 & \dots & 0 \end{pmatrix}
$$

$$
H(m+1) = \begin{pmatrix} c_{-1}(m+1) & c_{-2}(m+1) & \dots & c_{-N(m+1)}(m+1) \\ c_{-2}(m+1) & c_{-3}(m+1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ c_{-N(m+1)}(m+1) & 0 & \dots & 0 \end{pmatrix}
$$

We have to obtain the maximum singular value $\bar{\sigma}(m+1)$ and the corresponding right and left singular vectors $r(m+1)^t$, $s(m+1)^t$ from $H(m+1)$. It appears to be impossible to make a recursive step from $H(m)$, $\bar{\sigma}(m)$, $r(m)^t$, $s(m)^t$ to $H(m+1)$.

4.4 Step 2b

$$
A_{n(m)}^{N(m)}(E_n(\hat{h},w))(z) = \sum_{k=-N(m)}^{N(m)} c_k z^k - \frac{\bar{\sigma}\sum_{k=0}^{N(m)-1} s_{N(m)-k}z^k}{z^{N(m)}\sum_{k=0}^{N-1} r_{k+1}z^k} \qquad (4.5)
$$

As a consequence of the failure to write Step 2b as a recursive step, the only thing we can do is to rewrite the linear part of (4.5).

$$B_{n(m)}^{N(m)} = \sum_{k=-N(m)}^{N(m)} c_k(m) z^k$$

We will do so by using the second window function

$$\begin{aligned}
B_{n(m+1)}^{N(m+1)} &= \sum_{k=-N(m)}^{N(m)} c_k(m+1) z^k + \sum_{k=-N(m+1)}^{-N(m)} c_k(m+1) z^k + \sum_{k=N(m)}^{N(m+1)} c_k(m+1) z^k \\
&= \sum_{k=-N(m)}^{N(m)} \left(\frac{1}{2} c_k(m) \cdot \left(\frac{pN(m)+|k|}{pN(m)+p|k|} \right) + \tau_k(m+1) A_k(m+1) \right) z^k \\
&+ \sum_{k=-pN(m)}^{-N(m)} c_k(m+1) z^k + \sum_{k=N(m)}^{pN(m)} c_k(m+1) z^k \\
&= \sum_{k=-N(m)}^{N(m)} \frac{1}{2} c_k(m) \cdot \left(\frac{pN(m)+|k|}{pN(m)+p|k|} \right) z^k + \sum_{k=-N(m)}^{N(m)} \tau_k(m+1) A_k(m+1) z^k \\
&+ \sum_{k=-pN(m)}^{-N(m)} c_k(m+1) z^k + \sum_{k=N(m)}^{pN(m)} c_k(m+1) z^k
\end{aligned}$$

It is not possible to write this in terms of $B_{n(m)}^{N(m)}$, only for $p = 1$ we get

$$B_{n(m+1)}^{N(m+1)} = \frac{1}{2} B_{n(m)}^{N(m)} + \sum_{k=-N(m)}^{N(m)} \tau_k(m+1) A_k(m+1) z^k \tag{4.6}$$

Instead of using (4.3) we will go back to (4.2)

$$\begin{aligned}
B_{n(m+1)}^{N(m+1)} &= \sum_{k=-N(m)}^{N(m)} \left(\tau_k(m) + \frac{(p-1)|k|}{p \times N(m)} \right) \left(\frac{1}{2} \hat{c}_k(m) + A_k(m+1) \right) z^k \\
&+ \sum_{k=-N(m+1)}^{-N(m)} c_k(m+1) z^k + \sum_{k=N(m)}^{-N(m+1)} c_k(m+1) z^k \\
&= \sum_{k=-N(m)}^{N(m)} \left(\frac{1}{2} \tau_k(m) \hat{c}_k(m) + \frac{1}{2} \hat{c}_k(m) \frac{(p-1)|k|}{p \times N(m)} + \tau_k(m+1) \cdot A_k(m+1) \right) z^k \\
&+ \sum_{k=-N(m+1)}^{-N(m)} c_k(m+1) z^k + \sum_{k=N(m)}^{-N(m+1)} c_k(m+1) z^k
\end{aligned}$$

$$
\begin{aligned}
B_{n(m+1)}^{N(m+1)} &= \frac{1}{2} B_{n(m)}^{N(m)} + \sum_{k=-N(m)}^{N(m)} \left(\frac{1}{2} \hat{c}_k(m) \frac{(p-1)|k|}{p \times N(m)} + \tau_k(m+1) \cdot A_k(m+1) \right) z^k \\
&+ \sum_{k=-N(m+1)}^{-N(m)} c_k(m+1) z^k + \sum_{k=N(m)}^{-N(m+1)} c_k(m+1) z^k
\end{aligned}
\tag{4.7}
$$

4.5 Conclusion Recursive Algorithm

It is not possible to make a nice Recursive Algorithm in the terms of: we have an approximation $A_{n(m)}^{N(m)}$ and we are going to adjust it with the next recursive step. For the case where $N(m) = N(m+1)$ a lot of problems disappear.
It is possible to redefine Step 1 into a recursive step if we have chosen a convenient window function, see (4.3) and if $N(m) = N(m+1)$ (4.4).
The nonlinear behavior of Step 2 makes it impossible to make a nice recursive step. Only the linear part of $A_{n(m)}^{N(m)}$ namely $B_{n(m)}^{N(m)}$ is convenient to redefine, see (4.7) and if $N(m) = N(m+1)$ (4.6).

Chapter 5

Matlab

5.1 Program

% System Identification

```
clear;              % Clear memory
eps;                % Increase accurasy
i = sqrt(-1);       % Definition
rand('normal')      % "Rand"
l = 0;              % Initial
n = 0;              % Initial
```

```
% Number of experimental data E_n
% Number of DFT coefficients we want to calculate.

m = input('Number of experimental data is two to the power: ');
n = 2^m;        %Number of experimental data
N = input(' Number of DFT coefficients: ');


% Input of the noise on the system.
% Calculate the experimantal data.

disp(' ')
disp(' Do you want noise on this system ? ')
disp(' Yes : give the absolute value of the noise, ')
disp(' No : insert 0 ')
disp(' ')
```

```
fout = input('Give the absolute value of the noise : ');
if fout < 0
    error(' The absolute value must be larger than zero !!! ')
    else
        disp(' ')
        disp(' Calculate the experimantal data from the Transfer Fuction ')
        for l = 1:n
            % E is uniformly spread over the complex unit circle
            E = exp((2.*pi.*i.*(l-1))./n);
            % The noise comes out of the complex bowl with radius fout
            VERSTORING = fout.*rand.*exp(2.*pi.*i.*rand);
            ENHW(l) = Function(E) + VERSTORING;
        end
end



% Calculate the DFT coefficients
disp(' ')
disp(' Calculate the DFT-coefficients ')


% Initial
for p = 1:N, CKP(p) = 0;, end


% Calculate the DFT coefficients k > 0
% CKP = coefficients CK with K Positive
for p = 1:N;
    for l = 1:n
        DUMMY = CKP(p) + ENHW(l).*exp((-2.*pi.*i.*p.*(l-1))./n);
        CKP(p) = DUMMY;
    end
end


% DFT * window function (step 1b)
for p = 1:N;
    DUMMY = (CKP(p))./n;        % (definition)
    CKP(p) = DUMMY.*((n./(pi.*p)).*sin((pi.*p)./n));
end
```

```
% Calculate the DFT coefficients k = 0
% CKPN = coefficient CK with K Positive and Negative
% Window function at k = 0 equals 1
CKPN = 0;
for l = 1:n
    DUMMY = CKPN + ENHW(l);
    CKPN = DUMMY;
end
CKPN = DUMMY./n;        % (Definition)


% Initial
for p = 1:N, CKN(p) = 0;, end;


% Calculate the DFT coefficients k < 0
% CKN = coefficients CK with K Negative
for p =1:N
    for l = 1:n
        DUMMY = CKN(p) + ENHW(l).*exp((2.*pi.*i.*p.*(l-1))./n);
        CKN(p) = DUMMY;
    end
end


% DFT * window function (step 1b)
for p = 1:N
    DUMMY = (CKN(p))./n;        % (definitie)
    CKN(p) = DUMMY.*((n./(-pi.*p)).*sin((-pi.*p)./n));
end


% Step 2a, singular value decomposition from the Hankel matrix.
% Calculate the corresponding right and left singular vector.
disp(' ')
disp(' Calculate the HANKEL matrix')


% The Hankel matrix is made of the coefficients CK with
% the negative index (CKN).
% Hankel is a command in matlab.
H = hankel(real(CKN));
```

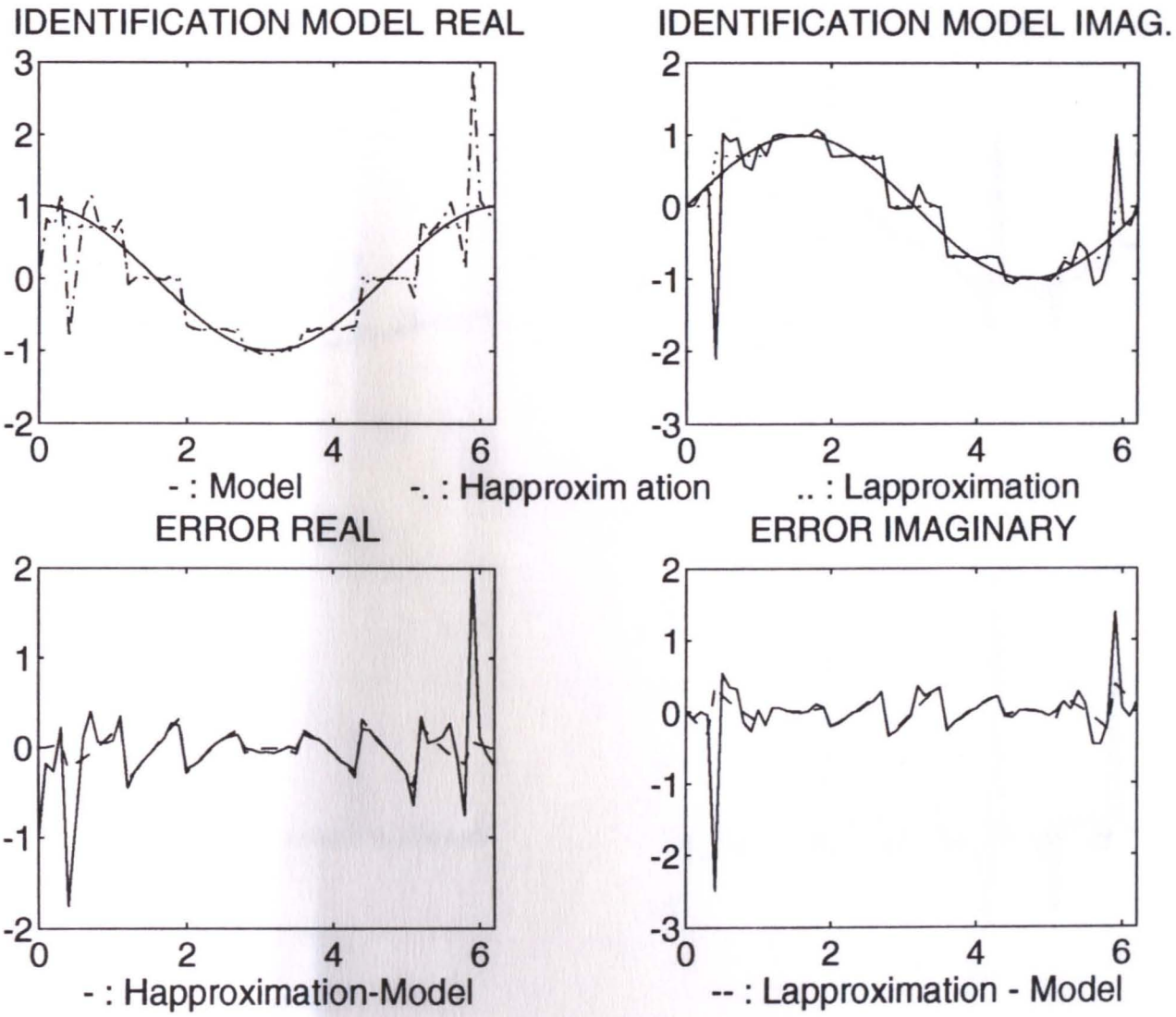
```
% Singular value decomposition
% svd is a command in matlab with S singular matrix
% and U & V unitary matrices : H = U * S * V' .
disp(' ')
disp(' Calculate the Singular values')
[U,S,V] = svd(H);
disp(' Largest Singular value'), S(1,1)


% Step 2b, Calculate the approximation

MODEL = 0;                % Real model without noise
HBENADERING = 0;          % Algorithm approximation (H∞)
LBENADERING = 0;          % Approximation (L∞)


w = 0 : 0.1 : 2. * π;     % Plot accuracy

z = exp(i. * w);          % Spread over the complex unit circle


disp('Calculation of the Happroximation ')


DUMMY1 = 0;
for p = 1:N
    DUMMY2 = DUMMY1 + CKP(p).*(z^p);
    DUMMY1 = DUMMY2;
end
CKZKP = DUMMY1;           % CKZKP : coefficients CK * Z^K with K Positive


DUMMY1 = 0;
for p = 1:N
    DUMMY2 = DUMMY1 + CKN(p).*(z^-p);
    DUMMY1 = DUMMY2;
end
CKZKN = DUMMY1;           % CKZKN : coefficients CK * Z^K with K Negative


% SOM from K= -N till N of CK * Z^K

CKZK = CKZKN + CKPN + CKZKP;
```

```
% Calculate the numerator, U is left singular vector
% of singulare value S(1,1)
DUMMY1 = 0;
for p = 1:N
    DUMMY2 = DUMMY1 + U(N-p+1,1).*(z^(p-1));
    DUMMY1 = DUMMY2;
end

TELLER = S(1,1).*DUMMY1;


% Calculate the denumerator, V is right singular vector
% of singular value S(1,1)
DUMMY1 = 0;
for p = 1:N
    DUMMY2 = DUMMY1 + V(1,p).*(z^(p-1));
    DUMMY1 = DUMMY2;
end

NOEMER = DUMMY1.*(z^N);


% Approximation agorithm step 1a till 2b
HBENADERING = CKZK - (TELLER./NOEMER);


% Approximatio step 1 :
% SOM from K= -N till N of CK * Z^K
LBENADERING = CKZK;


% Real model without noise.
MODEL = Function(z);
```

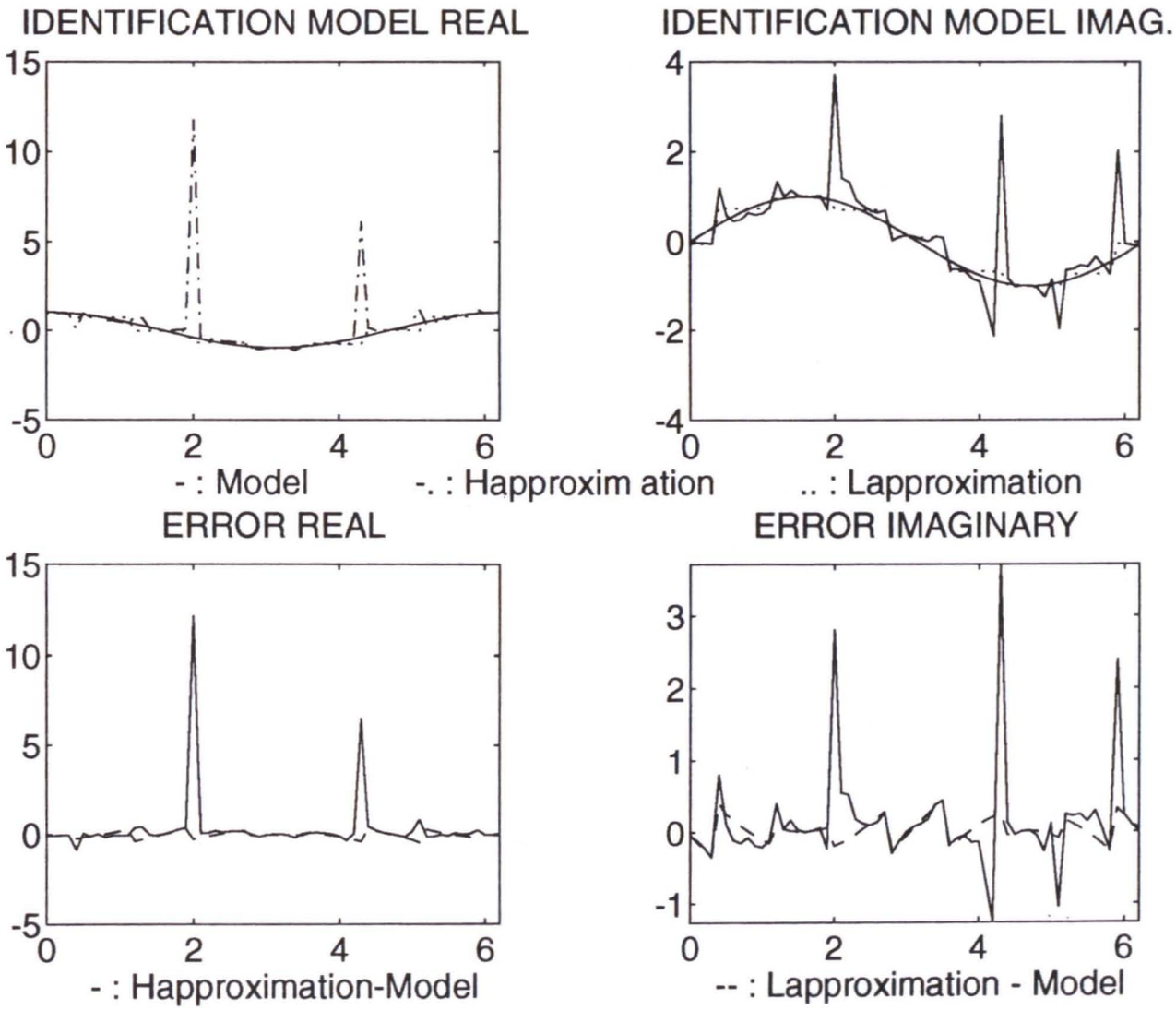
```
% Pictures.
clg


subplot(221),
plot(w,MODEL,'-g',w,HBENADERING,'-.r',w,LBENADERING,':w')
title('IDENTIFICATION MODEL REAL')
xlabel(' - : Model -. : Happroximation ')

subplot(223),
plot(w,real(BENADERING-MODEL),'-r',w,real(LBENADERING-MODEL),'-g')
title('ERROR REAL')
xlabel(' - : Happroximation - Model ')

subplot(222),
plot(w,imag(MODEL),'-g',w,imag(BENADERING),'-r',w,imag(LBENADERING),':w')
title('IDENTIFICATION MODEL IMAG.')
xlabel(' .. : Lapproximation ')
subplot(224),
plot(w,imag(BENADERING-MODEL),'-r',w,imag(LBENADERING-MODEL),'-g')
title('ERROR IMAGINARY')
xlabel(' - : Lapproximation - Model ')
meta plaat1


pause
clg



end
```

5.2 Pictures

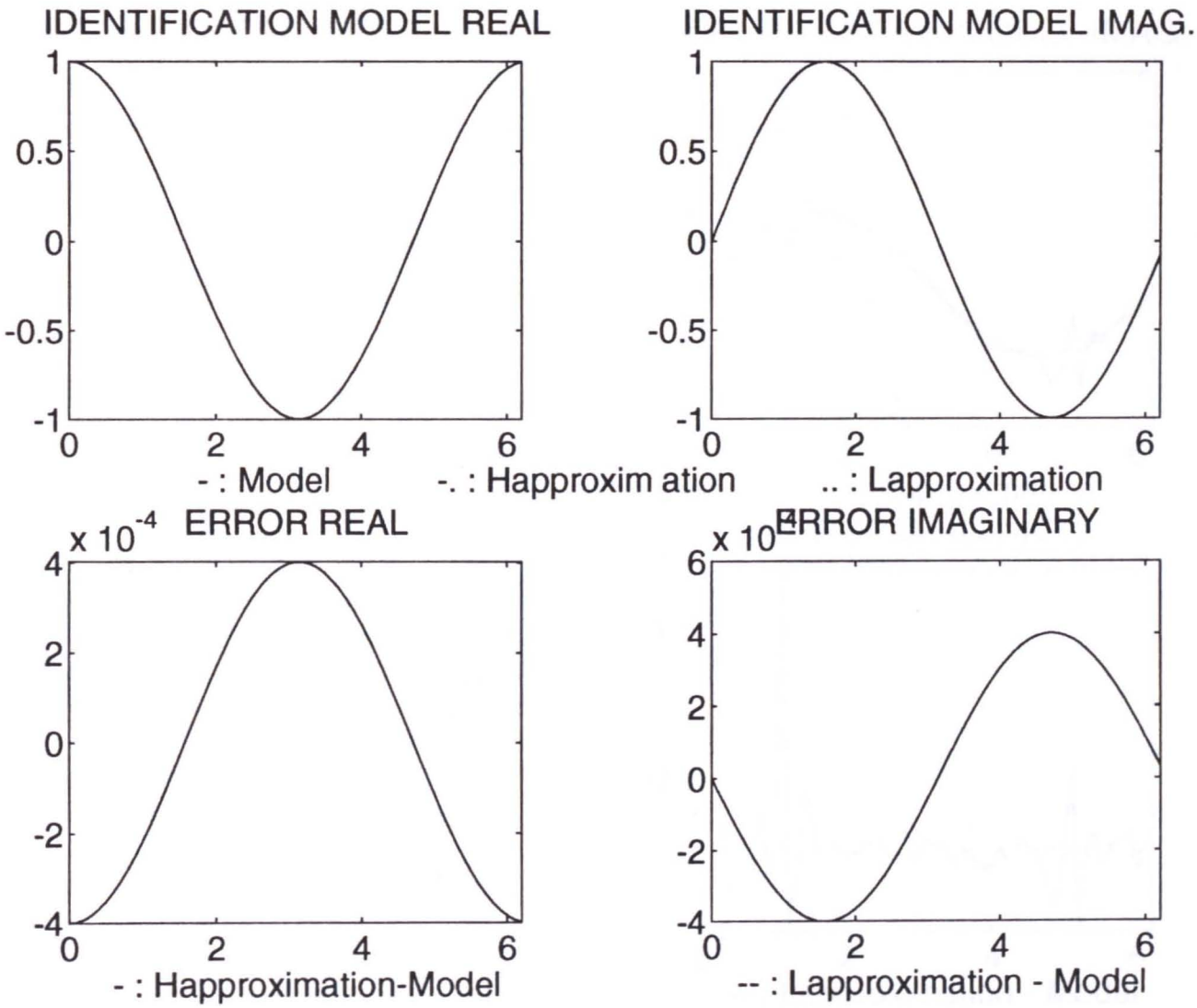
$\hat{h}(z)$	$=$	z
n	$=$	2^3
N	$=$	512
w	$\in$	$Bl_\infty(0)$



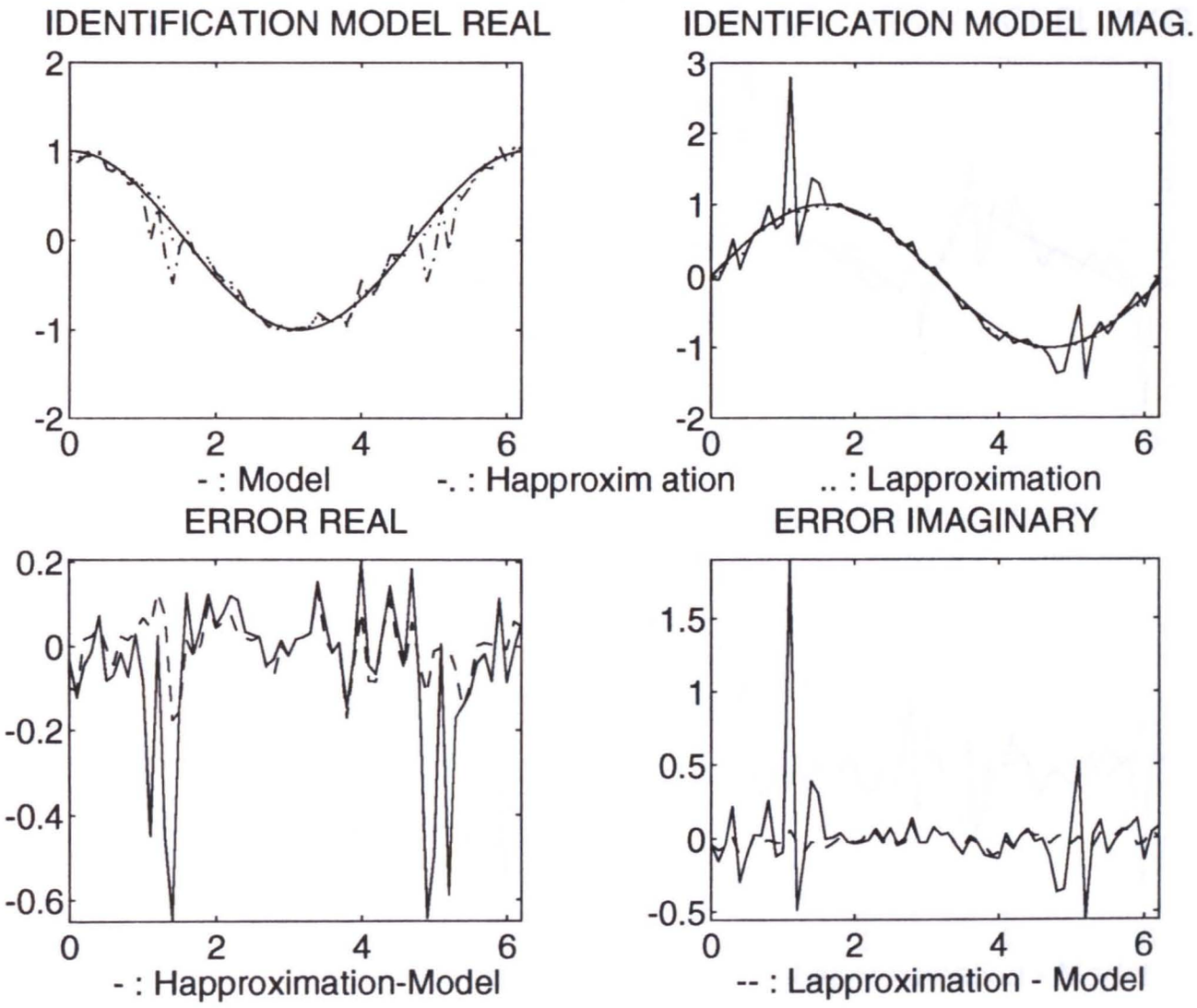
$\hat{h}(z)$	$=$	z
n	$=$	2^3
N	$=$	512
w	$\in$	$Bl_\infty(0.1)$



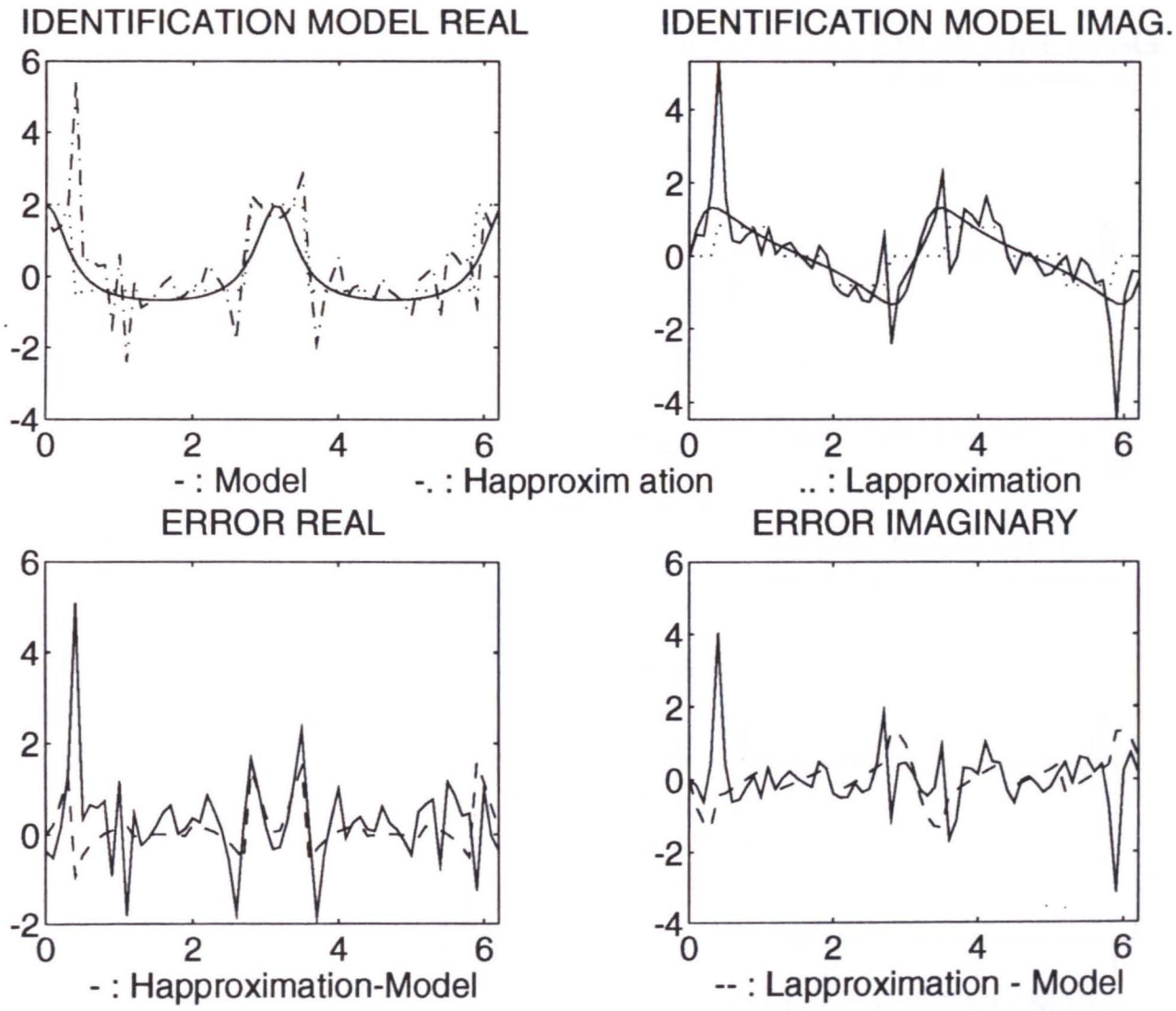
$\hat{h}(z)$	$=$	z
n	$=$	2^6
N	$=$	25
w	$\in$	$\bar{B}l_\infty(0)$



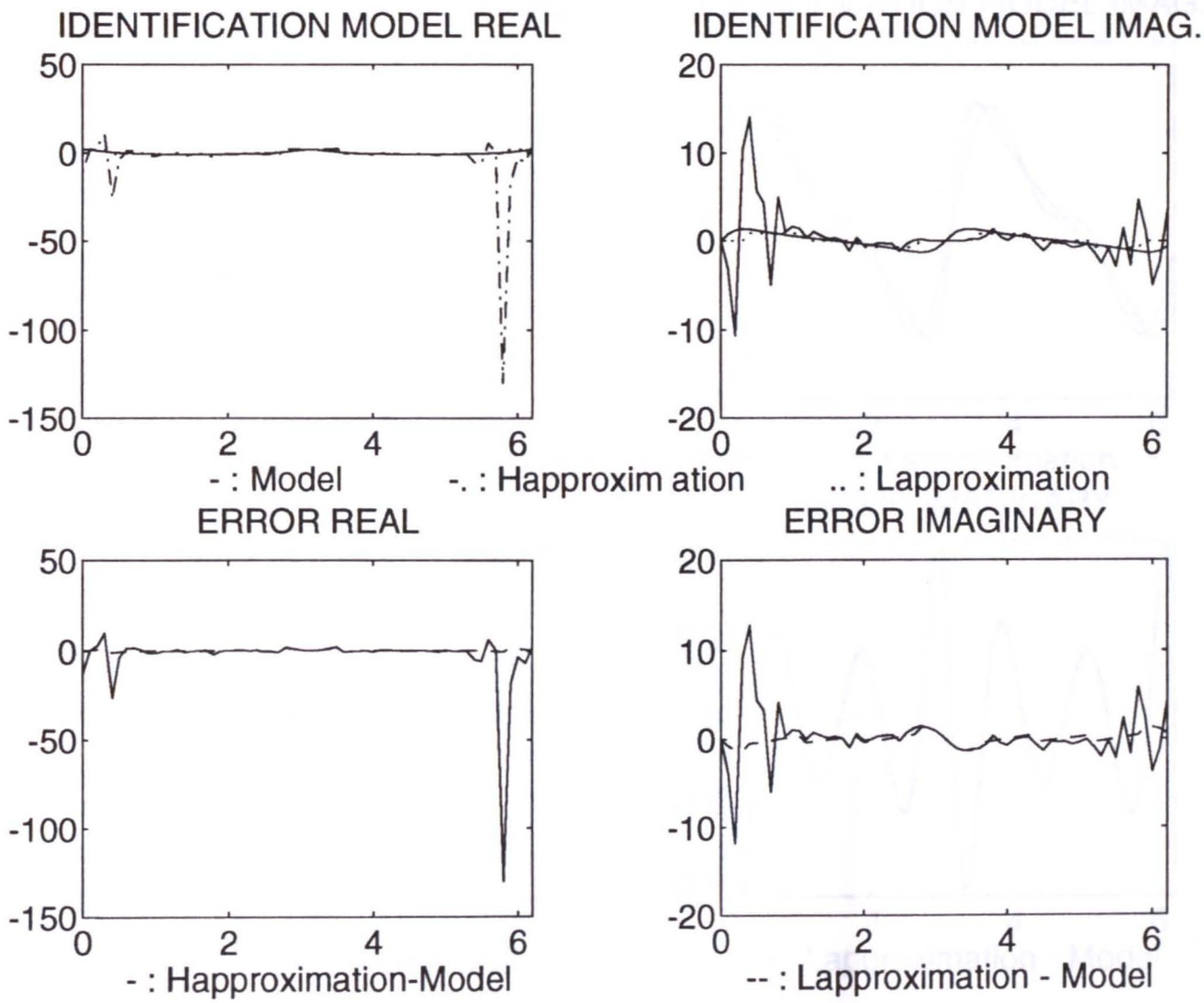
$\hat{h}(z)$	$=$	z
n	$=$	2^6
N	$=$	25
w	$\in$	$Bl_\infty(0.1)$



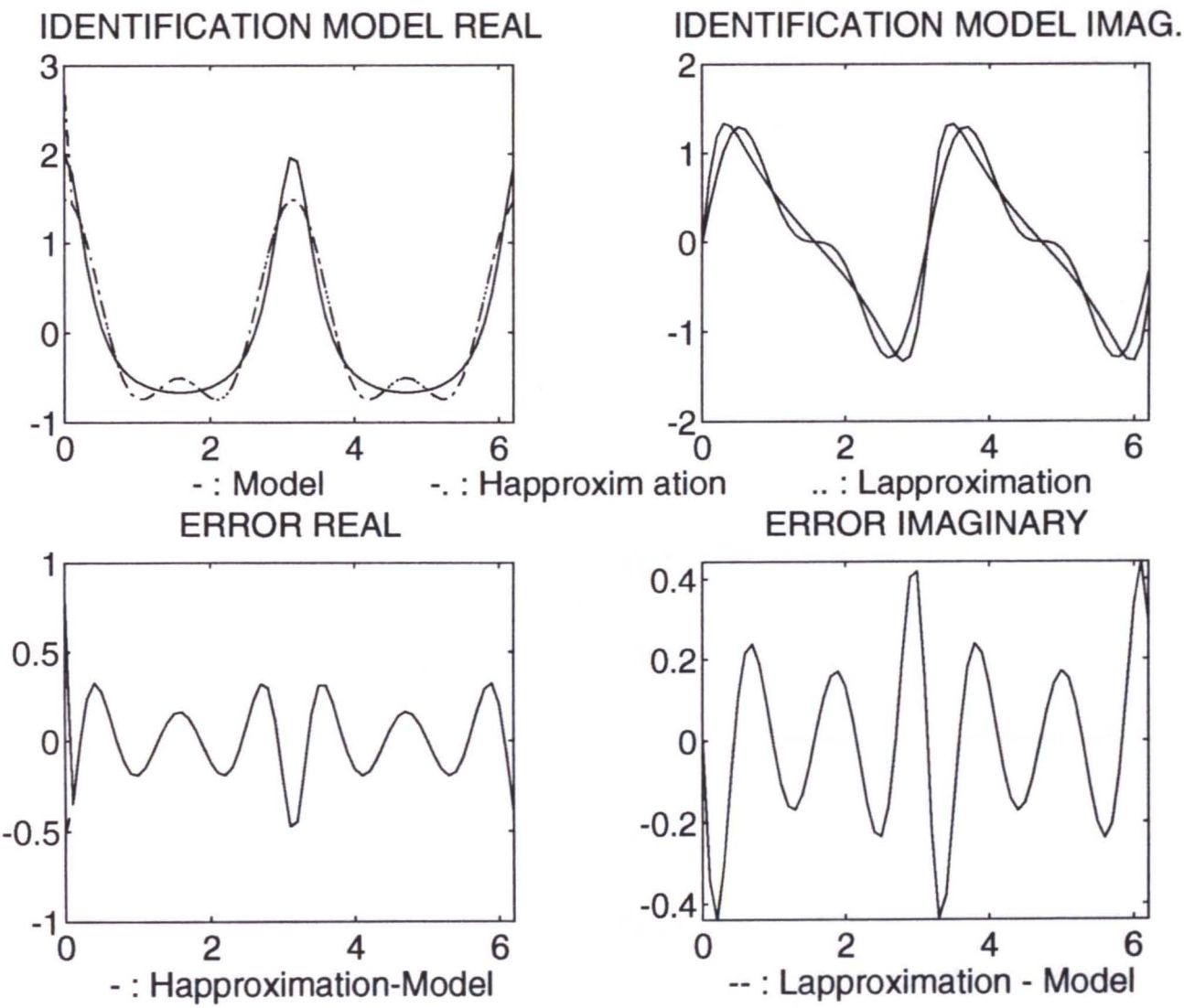
$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^3
N	$=$	512
w	$\in$	$Bl_\infty(0)$



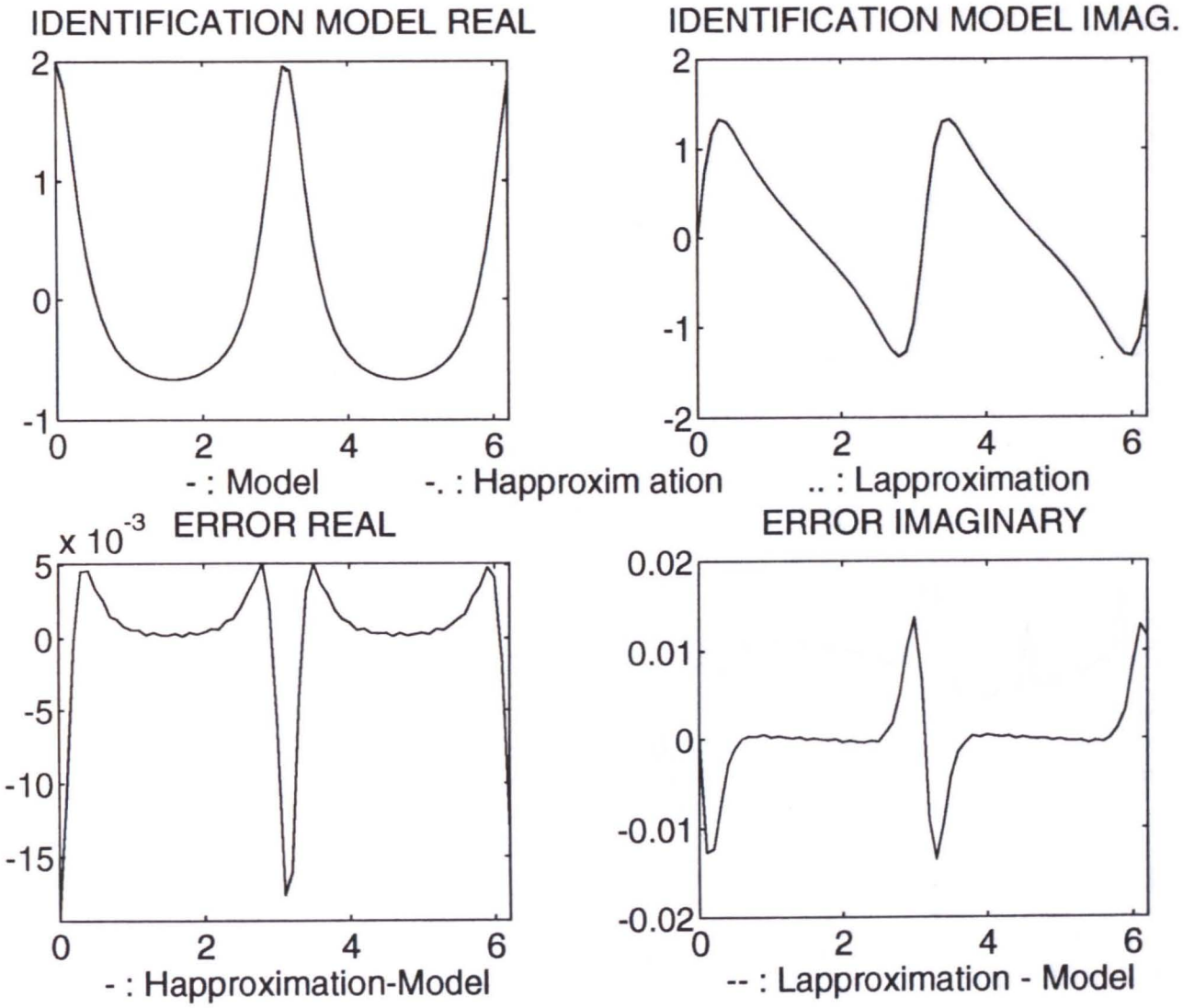
$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^3
N	$=$	512
w	$\in$	$Bl_\infty(0.1)$



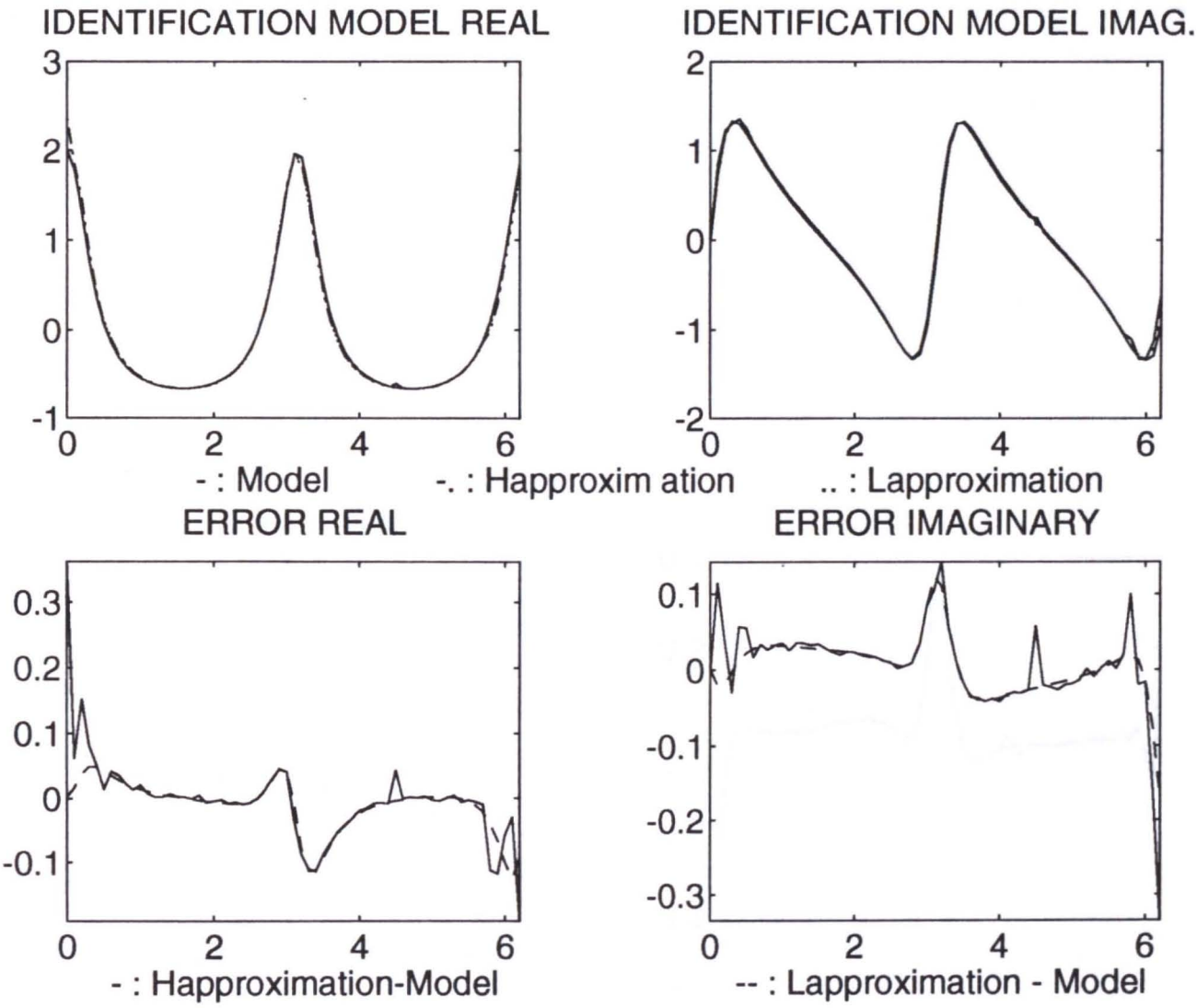
$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^6
N	$=$	5
w	$\in$	$Bl_\infty(0)$



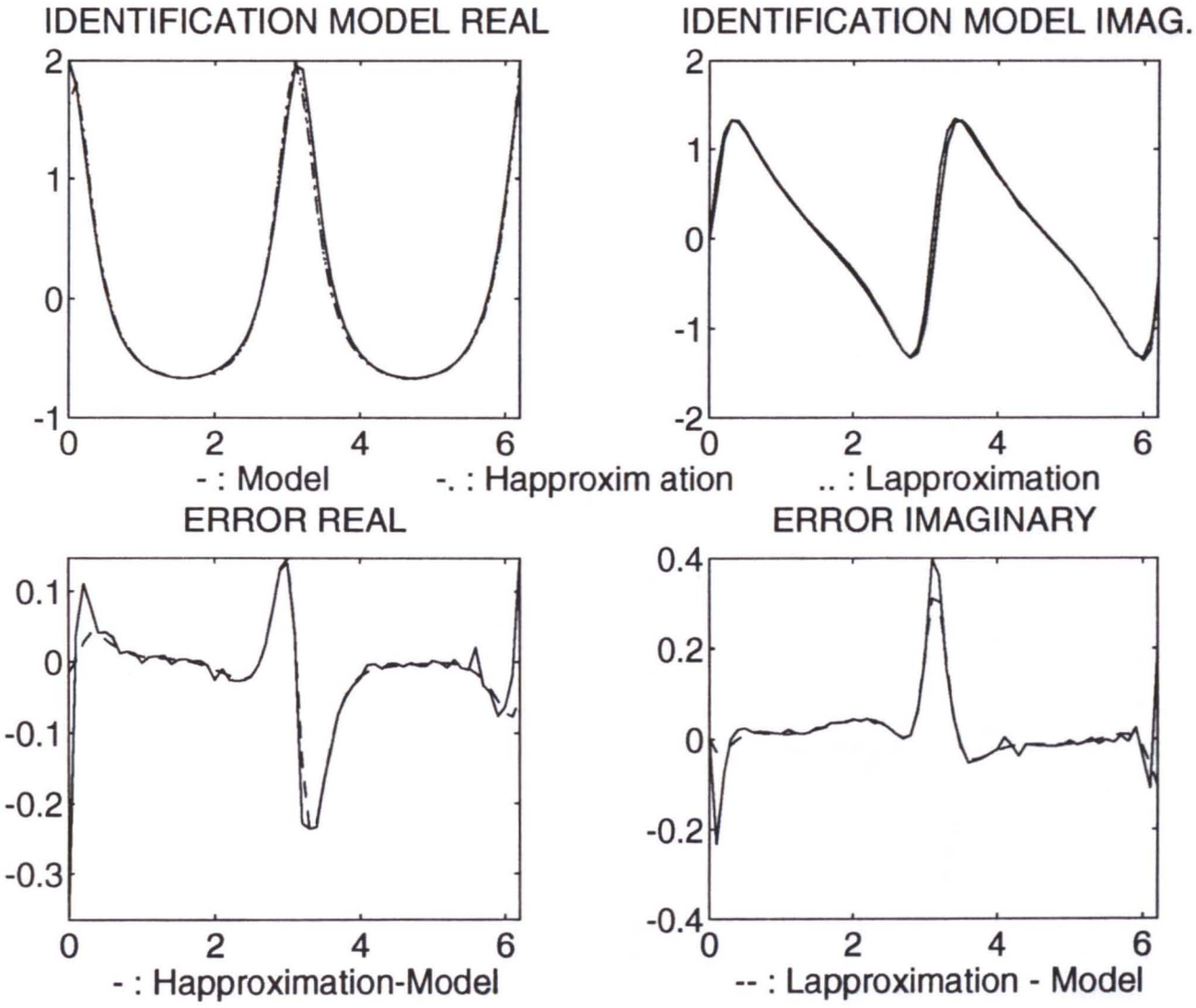
$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^6
N	$=$	25
w	$\in$	$\bar{B}l_\infty(0)$



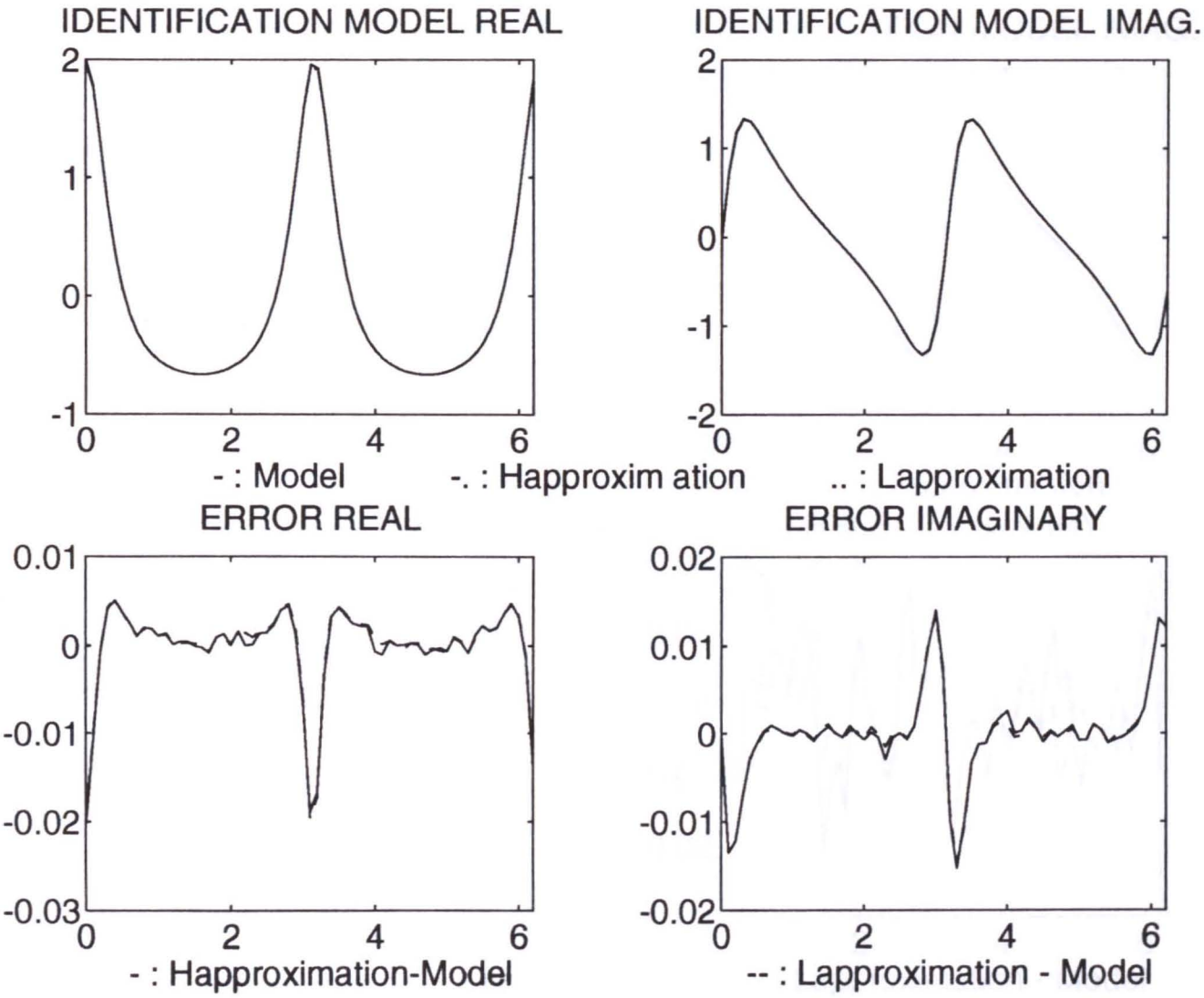
$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^6
N	$=$	100
w	$\in$	$\bar{B}l_\infty(0)$



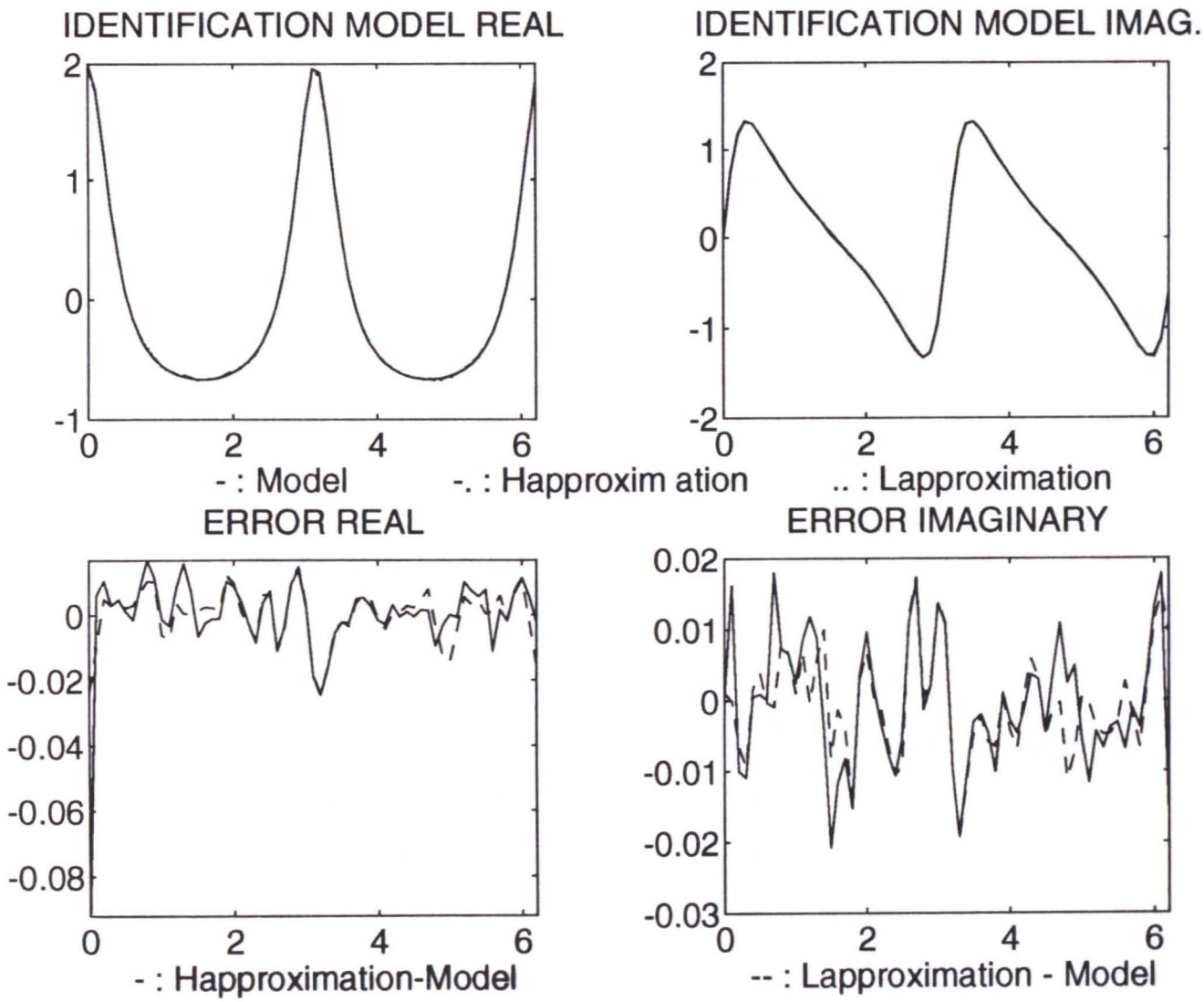
$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^6
N	$=$	250
w	$\in$	$\bar{B}l_\infty(0)$



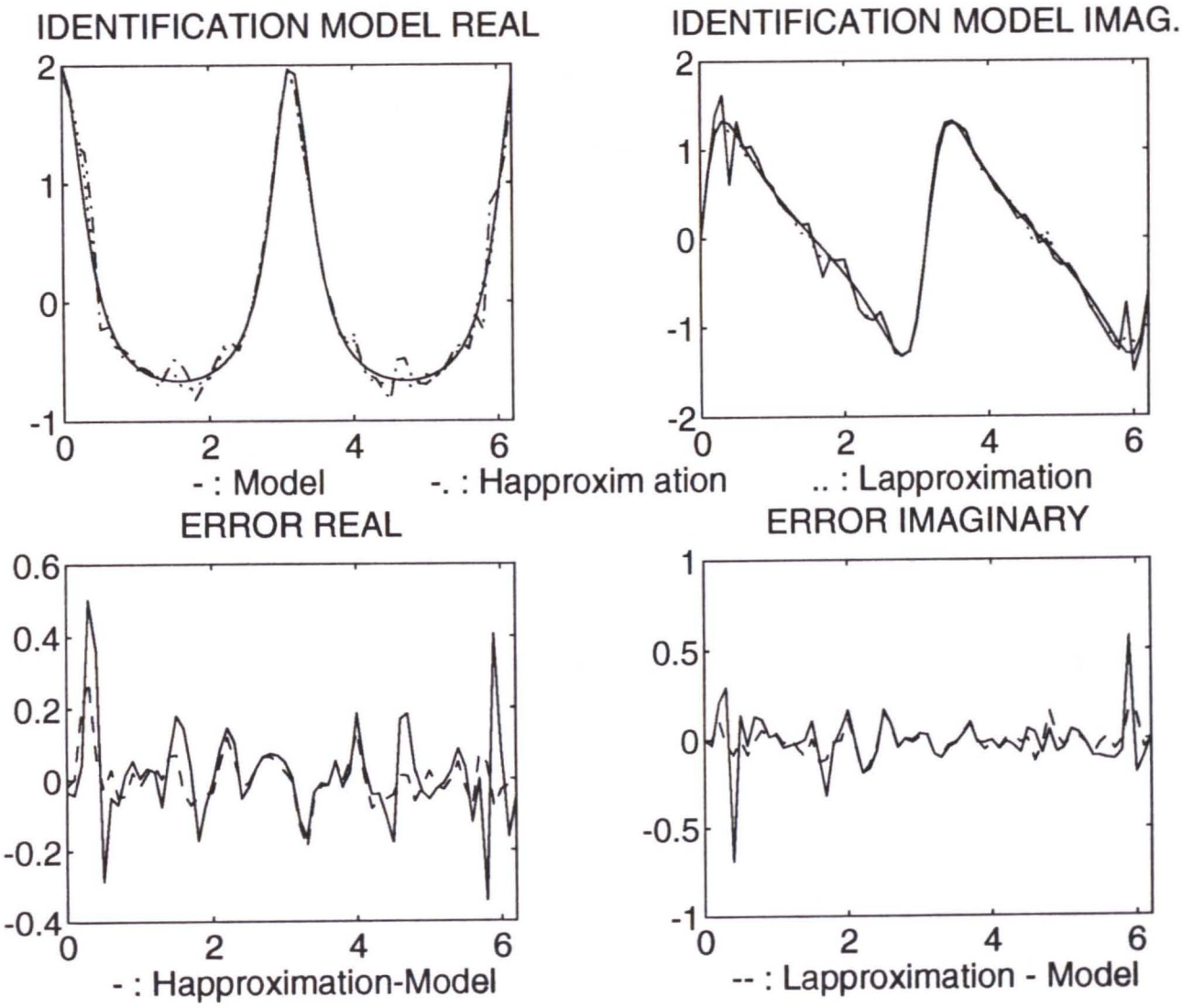
$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^6
N	$=$	25
w	$\in$	$\bar{B}l_\infty(0.001)$



$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^6
N	$=$	25
w	$\in$	$Bl_\infty(0.01)$



$\hat{h}(z)$	$=$	$\dfrac{z^2}{1-\frac{z^2}{2}}$
n	$=$	2^6
N	$=$	25
w	$\in$	$Bl_\infty(0.1)$



Bibliography

[1] A.J. Helmicki, C.A. Jacobson, and C.N. Nett
"Control Oriented System Identification:
A Worst-Case/Deterministic Approach in H_∞"
IEEE Transactions on automatic control, vol.36 no.10,
pp. 1163-1176 October 1991

[2] V.M. Adayan, D.Z. Arov, and M.G. Krein
"Analytic properties of Schmidt pairs for a Hankel operator
and the generelized Schur - Takaki problem"
math. USSR Sbornik, vol.15, no.1 pp. 31-73 1971

[3] P. Henrici
"Applied and Computational Complex Analysis, Vol.III"
New York, pp. 22-24,34 1986

[4] L.L. Schumaker
"Spline Functions: Basic Theory"
New York, pp. 220-222 Theorem 6.15 1981

[5] H.A. Priestley
"Introduction to Complex Analysis"
Oxford, pp. 66-67 Revised edition 1990

[6] N. Young
"An Introduction to Hilbert Space"
London, U.K. pp. 187-199, 210-219 1988