KAO, Charles Tzu-Tai, 1944-
THE SPLINE APPROXIMATION IN SYSTEM SIMULATION BY DIGITAL COMPUTER.

Rice University, Ph.D., 1972
Engineering, electrical

University Microfilms, A XEROX Company, Ann Arbor, Michigan

© 1972

CHARLES TZU-TAI KAO

ALL RIGHTS RESERVED
RICE UNIVERSITY

THE SPLINE APPROXIMATION IN
SYSTEM SIMULATION BY DIGITAL COMPUTER

by

Charles Tzu-Tai Kao

A Thesis Submitted
In Partial Fulfillment Of The
Requirements For The Degree Of

DOCTOR OF PHILOSOPHY

Thesis Director's Signature

Rui J. P. de Figueiredo

Houston, Texas
May, 1972
PLEASE NOTE:

Some pages may have

indistinct print.

Filmed as received.

University Microfilms, A Xerox Education Company
ACKNOWLEDGEMENTS

I wish to express my gratitude to my research advisor Professor R. J. P. de Figueiredo for his constant guidance and many valuable suggestions. I am grateful to Professors A. Miele and C. S. Burrus for having served on the oral examination committee.

Special thanks to my wife, Savina, for her assistance and understanding during these graduate school years and typing of this manuscript.

Finally, I want to thank my parents for their encouragement.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I.</td>
<td>INTRODUCTION TO RESEARCH AREA</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.1 Introduction</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1.2 A Brief Survey of the Pertinent Literature</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>1.3 Proposed Approach</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>1.4 Outlines of this Thesis</td>
<td>8</td>
</tr>
<tr>
<td>II.</td>
<td>FUNDAMENTAL DEFINITIONS AND CONCEPTS</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2.1 Preliminary Definition</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>2.2 On System Simulation and Modeling Concepts</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>2.2.1 Formulation of a Model</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>2.2.2 Formulation of a Digital Simulator</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>2.2.3 Validation</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>2.3 On Approximation Theory and Spline Functions</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>2.4 Discrete System and Computer Simulation</td>
<td>25</td>
</tr>
<tr>
<td>III.</td>
<td>SYSTEM SIMULATION BASED ON FREQUENCY DOMAIN</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>APPROXIMATION BY THE SPLINE FUNCTION</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3.1 Introduction</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>3.1.1 Problem Formulation</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>3.1.2 Outlines of this Chapter</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>3.1.3 Nomenclature</td>
<td>34</td>
</tr>
<tr>
<td></td>
<td>3.2 Description of the General Approach</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>3.3 Principle of Smoothness and Review of Previous Application</td>
<td>43</td>
</tr>
</tbody>
</table>
3.4 Two New Simulation Schemes Based on the Spline Approximation in the Frequency Domain

3.4.1 Scheme I — For Simulating a System with Transfer Function Specified on Frequency Intervals

3.4.2 Scheme II — For Simulating a System with Transfer Function Specified at Discrete Points

3.5 Optimal Property and Error Analysis

3.5.1 Optimal Property and Error Analysis for Scheme I

3.5.2 Optimal Property and Error Analysis for Scheme II

3.6 Generalization and Conclusion

IV. NUMERICAL EXPERIMENTS ON SYSTEM SIMULATION PROBLEMS FROM FREQUENCY DOMAIN APPROACH

4.1 Numerical Examples

4.2 Comparison and Conclusion

V. TIME DOMAIN SIMULATION BASED ON THE SPLINE APPROXIMATION

5.1 Problem Formulation

5.2 Review of Previous Results

5.3 Description of the Method

5.4 Convergence and Error Analysis

5.5 An Illustrative Example

5.6 Generalization and Conclusion
VI. CONCLUSION AND PROPOSALS FOR FURTHER RESEARCH 114
  6.1 Summary and Conclusion 114
  6.2 Suggestions for Further Studies 115

APPENDIX A: The Frequency Domain Simulation Method Based on the Finite Duration Impulse Response Digital Filter Design Technique 117

APPENDIX B: A Sample Program for Scheme I 120

APPENDIX C: A Sample Program for Scheme II 121

REFERENCES 123
CHAPTER I
INTRODUCTION TO RESEARCH AREA

1.1. Introduction

The use of system simulation by means of digital computers is expanding rapidly, and the development of the new techniques is influencing many disciplines. Computer simulation concepts and techniques have been widely applied in various areas such as to communication systems, transportation and queuing systems, automatic control systems and in process control, aerospace, bio medical engineering, social dynamics, ecology, military and civilian engineering problems, etc. Computer simulation is generally used as a problem solving technique when the physical system has certain inherent constraints which prohibit direct experimentation because the latter becomes impossible, impractical, uneconomical or slow. In some cases, it is quite possible that computer simulation is the only method available for the systematic quantitative study of a system.

The availability of high speed computers provides us a very powerful, flexible and economical tool to perform almost all kinds of simulations. Computer simulation can be applied to a wide variety of systems, both real and hypothetical, and it can be employed for many different purposes. The many sophisticated peripheral types of equipment associated with modern day computers, such as D/A or A/D
converters, display systems, etc., greatly enhance their simulation capability.

Normally, the static response of a physical system can be simulated with less effort on a digital computer than its dynamic response. The purpose of this thesis is to develop techniques for simulating the dynamic response of a linear, time-invariant system under the assumption that the transfer function or the input belongs to certain classes. Another basic assumption made is that when performing the simulation we only have partial knowledge of the values of the transfer function or the input. Specifically we assume in Chapter III that we know a-priori either the frequency response over certain frequency intervals or at certain values of the frequency. In chapter V we assume that the values of the input only at the sampling instants of time are known. Under these conditions then, our objective has been to develop techniques for the design of a discrete system which can be easily implemented on a digital computer to simulate the actual system in an efficient and satisfactory way.

Works of other authors pertinent to this problem will be discussed in the following section. The need of a new simulation scheme to fit our requirements and the proposed approach are explained in Section 1.3. The outline of the results developed in this dissertation will be stated in
1.2. **A Brief Survey of The Pertinent Literature**

The methods for simulating the dynamic response of a continuous system on a digital computer can be classified into two categories involving frequency domain and time domain approximations. Most of the previous literature pertinent to our problem is related to digital filter design techniques.

In the frequency domain methods, various transformation techniques which map the transfer function of the continuous system into the transfer function of the discrete system were developed by Steiglitz [12], Greaves and Cadzow [19], Gold and Rader [1,2], etc., several windowing techniques were also developed by Kaiser [59], Ku, et al [12]. These methods have a common disadvantage that they are applicable only if the transfer function of the continuous system is given.

When the desired frequency response is piecewise specified (i.e. over given disjoint frequency intervals), Martin [4] and Ormsby [5] reconstructed the frequency response with the sine and palabolic functions respectively to improve the convergence rate of the corresponding Fourier series truncation method. They showed that the resulting non-recursive filter is more efficient, but prvided no error analysis nor indicated any optimal property associated with their schemes.
Several numerical algorithms were developed to design the optimal digital filter in the Chebyshev-sense by Parks and McClellan[70], Herrman[3], Hofstetter[76], Tufts and Francis[67]. These methods are usually applied to design simple cases such as the ideal low pass filter. It is to be expected that these methods will turn into a complicated and time-consuming procedure when applied to the general case. The entire algorithm might not converge to the solution if improper initial conditions are assumed. For the purpose of comparison, we may say that to design an ideal low-pass filter with length 95 by the Parks-McClellan approach (which uses the Remes algorithm), 200 seconds are required on the Rice Burrough 5500 computer†. Numerical experiments given in Chapter IV show that for the same ideal low-pass filter, the time required to design a total of 50 digital filters of length 3 to 101 is less than 2 seconds on the same computer by our method, without too much difference in the error. (See Example 4.2 in the Chapter IV this thesis). Also none of these methods considered phase frequency response as we do.

As the frequency response of the actual system is pointwise defined, (i.e. only at a discrete set of points in frequency), Fleischer[48] developed least-squares-fitting methods and Rabiner[50] used the "Frequency Sampling" method. Experience[12,50] tells us that the error between two frequency samples might be significant.

Note:† In a private communication, Professor T.W. Parks has informed the author of this thesis that he has been able to reduce the 200 seconds time interval quoted in reference 70 to 50 seconds.
In the time domain approach, the main idea is to approximate the value of the convolution integral which is associated with the actual system. Normally, the transfer function or the impulse response of the system is completely defined and the input signal is given in a tabular form. Various methods are developed to reconstruct the convolution integrand. For example, R. Vich [40] used the Gregory formula, Harrison and Leon [21] used piecewise polynomials, de Figueiredo and Kao [37] used the cubic spline, de Figueiredo and Natravali 36 used the generalized spline. Certain optimal properties were derived for the last two schemes.

In addition to the above approaches, a third choice is to use a particular problem-oriented simulation language such as CSMP. [8, 75] Usually this is a more expensive and inefficient way to solve the simulation problem. [13] The merit and disadvantages of this approach will be discussed in Section 2.2.

1.3. Proposed Approach

From the above discussion, it is found that most of the existing simulation schemes require the continuous (actual) system to be completely defined. The other schemes have also shown the disadvantages regarding accuracy or proved to be complicated. Moreover, most frequency domain techniques have ignored the phase criterion. This has partly motivated us to seek an efficient solution to the
above problem based on the spline approximation. We state the proposed approaches and the main contribution as follows:

For studying computer simulation problems, one usually attempts to derive an effective approximation procedure or develop an applicable computational algorithm to evaluate the actual system output. [8, 9, 10, 44, 47] There are three main steps for solving system simulation problems: the first step is to find an adequate mathematical model which represents the actual system with sufficient detail and accuracy, but is not too complicated to be analyzed by computer. The second step is to derive a discrete system and then implement it on a digital computer to properly duplicate the performance of the continuous system. The third step is to check out the whole simulation scheme and evaluate the simulation errors by some test runs. If the results are satisfactory, actual data may be applied to study the system. Otherwise we go back to step 1 and step 2 to make certain changes to improve the simulation scheme. This procedure is continued until all the specified criteria are satisfied.

In the above simulation procedure, various approximation methods are employed in step one and step two to solve the system modeling problem and the discrete system derivation problem. Due to the many desirable properties such as high convergence rate and optimal approximation
properties of the spline function approximation explored recently, \[39, 42, 58\] we are interested in applying spline functions to solve these problems.

It is clear that step one is omitted when the transfer function of the system is completely defined. The spline approximation is applied in the frequency domain to approximate the frequency response of the optimal discrete system when the frequency response of the actual system is only partially defined. A non-recursive digital filter type simulator is derived which has both excellent magnitude and phase responses. In this application, spline approximation is used to solve the system modeling problem so that an error bound in frequency domain is minimized.

We also consider the case in which the frequency response of the actual system is completely specified but the input signal is known to have energy only on certain prescribed frequency intervals. In this case, we replace the actual frequency response on those frequency intervals where the signal has zero energy by spline approximation. This leads to an improved simulation scheme and is explained in detail in Chapter III.

Finally, when the system function is completely specified the spline approximation is used in the time domain to interpolate the samples of the input and then yield the structure of an optimal discrete simulator as
presented in Chapter V.

1.4. Outline of This Dissertation

We start with the fundamental definitions and concepts stated in Chapter II. Emphasis is on the introduction of spline functions, system simulation and modeling concepts. Approximation theorems and the basics of computer simulation are also presented. In Chapter III, we develop the simulation schemes from the frequency domain approach based on the spline approximation. The optimal property consisting of the minimization of an error bound is discussed, and the effect of the smoothness properties of the spline approximation in the construction of the desired frequency response of the digital simulator is studied. The theoretical developments in Chapter III are further articulated by the numerical experiments, performed on the Rice Burroughs 5500 digital computer, discussed in Chapter IV. In Chapter V, we discuss the simulation scheme based on time domain approach. An efficient computation scheme based on the spline approximation and a numerical example are given to illustrate the method. Conclusions and proposals for further research are presented in Chapter VI. In all cases, error bounds are presented in explicit forms.

An interesting problem of interpolating the frequency samples by the trigonometric functions, namely, sine and cosine functions such that a finite duration non-recursive
digital filter can be derived is given in Appendix A. Two sample computer programs in Fortran IV to evaluate the coefficients of the discrete simulator by using the cubic spline approximation are listed in Appendixes B and C. Figures are at the end of each chapter.
CHAPTER II
FUNDAMENTAL DEFINITIONS AND CONCEPTS

2.1. Preliminary Definition

In this chapter, we summarize the definitions and results which are applicable to all chapters. Additional notations and definitions appear in Chapters III and V pertinent to the topics defined respectively in those individual chapters.

a) System: "A system is an aggregation or assemblage of objects joined by some regular interaction or interdependence".\[8\]
b) Input: The input to the system is either a time function \(x(t)\) or a sequence \(\{x_k\}\) which consists of the external force or stimulus acting upon the system. We call \(x(t)\) a continuous-time signal which is necessarily time-limited, but consideration of theoretically band-limited signals will not be precluded. \(\{x_k\}\) is the discrete-time signal which is a sequence of numbers usually obtained by sampling a continuous time signal at time instant \(t_k\), i.e. \(x_k = x(t_k)\).

c) Output: Similarly, \(y(t)\) denotes the value of continuous-time output and \(y_k\) is the sample of \(y(t)\) at \(t = t_k\).

d) Simulation: "Simulation is essentially a working analogy. Analogy means similarity of properties or relations without identity".\[44\]

e) \(L\{\cdot\}\) and \(L^{-1}\{\cdot\}\) denote Laplace transform pair.

f) \(F\{\cdot\}\) and \(F^{-1}\{\cdot\}\) denote Fourier transform pair.

Note: \(^\dagger\) The correct notation for a function would have been \(x(\cdot)\). However, we write as indicated to display the symbol used for the argument. The procedure will be followed where appropriate.
g) $z\{\cdot\}$ and $z^{-1}\{\cdot\}$ denote z-transform pair.

h) Transfer function for Continuous-time system $H(s)$:

For a linear time-invariant system, the transfer function $H(s)$ of the system is the Laplace transform of its impulse response $h(t)$ or equivalently, the ratio of the Laplace transform of the zero-state response $Y(s)$ to that of the input $X(s)$.

i) Transfer function for a discrete system $H_D(z)$:

For a linear, time-invariant, continuous state discrete time system (henceforth denoted simply by discrete system), the value of its transfer function $H_D(z)$ is the z-transform of its weighting sequence (digital impulse response) $\{c_k\}$, or equivalently, the ratio of the z-transform of the output sequence $Y(z)$ to that of the input $X(z)$ and is of the form

$$H_D(z) = \frac{a_0 + a_1 z^{-1} + \ldots + a_m z^{-m}}{1 + \beta_1 z^{-1} + \ldots + \beta_n z^{-n}}.$$ (2.1)

$H_D(z)$ is also called the transfer function of a digital or numerical filter. It is called a non-recursive digital filter if and only if $\beta_i = 0$ for $i = 1, 2, \ldots, n$, otherwise it is called recursive digital filter and in such a case, a feedback mechanism is employed.

j) Frequency response: The frequency response of a linear time-invariant continuous state, continuous time system (henceforth called simply continuous system) is the Fourier
transform of the impulse of the system and it is denoted by

\[ H(j\omega) = H_R(\omega) + jH_X(\omega) \tag{2.2} \]

where \( H_R(\omega) \) and \( H_X(\omega) \) are real valued functions of the frequency \( \omega \) and they represent the real and imaginary parts of \( H(\omega) \).

Throughout this thesis, the frequency response means the values of \( H_R(\omega) \) and \( H_X(\omega) \) if it is not especially defined.

For a discrete system, the frequency response is the value of \( H_D(z) \bigg|_{z=e^{j\omega T}} \), where \( T \) is the sampling interval.

The frequency response of a discrete system, being periodic in nature, can never be band-limited. In software simulation, we generally use a discrete system to simulate the performance of a continuous system. Hence the frequency response of these two systems are compared in the frequency range from 0 to \( \Omega_F \), where \( 2\Omega_F = 2\pi/T \) is the sampling frequency.

k) The frequency response or spectrum, \( H(j\omega) \), is said to be band-limited if \( |H(j\omega)| = 0 \) for \( |\omega| > \Omega \), where \( \Omega \) is a constant.

m) \( \mathcal{K}^n_{[a,b]} \): The class of all real valued functions defined on \([a,b]\) which possess an absolutely continuous \((n-1)\)th derivative on \([a,b]\) and whose \(n\)th derivative is in \( L^2(a,b) \).

n) \( \mathcal{K}^n_{p[a,b]} \): The subclass of function in \( \mathcal{K}^n(a,b) \), which together
with their first \( n-1 \) derivatives, have continuous periodic extensions to \((-\infty, \infty)\) of period \([b-a]\).

2.2. On System Simulation and Modeling Concepts

Although the applications to which simulation techniques are pertinent are diverse, the process or approach to the simulation problem is essentially the same. The design of a digital simulator to simulate a physical system on computer usually involves a procedure consisting of three major steps: formulation of a model, formulation of a digital simulator, and validation. These steps are described as follows:

2.2.1. Formulation of A Model

The most essential step in simulation is the construction of a working mathematical or physical model presenting similarity of properties or relationships with the natural system under study.[8,44]

Since the real world is too complicated to be described faithfully by a set of equations, any resulting model, even if it could be constructed, would make computation or analysis difficult. Therefore, it is necessary to develop models that only approximate the real world to make them mathematically and computationally feasible. There is no strict rule how this model may be obtained so that we may be able to construct analogous systems based on this model to take measurements and observations to predict the performance of the actual system. In most cases, simulation studies, to be
properly done, require the making of new models to fit, with the required precision determined by the applications. According to D. Chorafas\textsuperscript{[44]}, it is like developing a special kind of medicine to suit the needs of an individual patient.

There are many techniques for defining a model ranging from rigorous mathematical treatment on one extreme to a heuristic common sense treatment on the other extreme. In other words, we may treat the actual system as a black-box and use pure mathematical approximation methods to interpolate or find the characteristics of the system such as a Markov chain. This approach is mostly employed in social, economic, political, medical and certain complicated or large scale system studies, where the behavior of the system is known but the processes that produce the behavior are not. Another approach is to physically interpret and analyze the actual system. In other words, the variables and parameters in the mathematical model have certain physical meanings. The relationships between these variables and parameters can be derived analytically. Usually this approach is based on certain assumptions to simplify the analysis.

Experience has shown that the task of modeling is much easier if the designer knows something about the nature of the actual system and incorporates this \textit{a-priori} knowledge in modeling.

After a mathematical model is obtained, we proceed to
the next step.

2.2.2. **Formulation of a Digital simulator**

Since we are using the electronic digital computer as our simulation tool, the procedure to reproduce the output must be converted to a computer program to be implemented on that computer. We will refer this simulation program as "digital simulator". There are two basic approaches to obtain such a computer program which represents the developed mathematical model. The first one is to use a simulation language such as COGO, CSMP, STRESS, GPSS, ECAP, SIMSCRIPT [8, 45, 46]. The other is to implement the model using a procedure-oriented language such as FORTRAN, COBOL, PL/I etc. installed on the computer which is used for simulation. There are certain conveniences and inconveniences associated with each approach. When a specific simulation language is used, the method of solution is built into the computer program that interprets problem-oriented language statements. This enables the users to obtain computer solution to problem that they could not solve themselves.

As for the disadvantages, one of them is that the simulation problem must belong to the class which can be solved by that particular simulation language. Since the simulation language is designed for general cases, the resultant object module may not be very efficient. [13]

Most of these simulation languages use numerical integration to solve differential equations such as CSMP, Dynamo,
etc., and it usually requires the integration step to be selected as very small to reduce the approximation error. Therefore, it takes a lot of computer time to execute these numerical integrations even if the desired accuracy is not very high. On the other hand, the rounding error in the arithmetic operation on a computer increases significantly if the integration step is small. [52] Therefore, the accuracy of simulation by using this approach is essentially limited. However, the most serious disadvantage probably is that the simulation language (compiler or translator) is not available for the computer which is used for simulation.

The second approach of using a procedure-oriented language to program a simulation problem gives maximum flexibility. However, the user must have a good understanding of the procedure to obtain the solution and the nature of the system to be simulated. Throughout this thesis, we are interested in developing the algorithm to construct such a program from this approach.

In practice, the question of how difficult it is to program a model often influences how the model is constructed. The tasks of producing a model and a digital simulator are likely to be carried out in parallel rather than in series. Normally, a linear or linearized model is preferable. It is easy to be analyzed and simulated since the superposition principle can be applied for dynamic response study.

The general approach is given in section 2.4, and we
will discuss several new simulation techniques in Chapters III and IV respectively.

2.2.3. Validation

The validation of the computer program or digital simulator is an area requiring a good deal of judgement. Some test runs are executed and the inference made in establishing the model are checked by observing if the simulator behaves as expected.

There are 3 main error sources:
(i) Modeling error - the difference between the mathematical model and the actual system.
(ii) Conversion error - the error due to using discrete model to represent the continuous model.
(iii) Quantization error - the error due to coefficient truncation on model parameters or the rounding error in computer arithmetic operation.

Since many assumptions and simplifications may be made intuitively in constructing a model and programming it on the computer, the validation of the resulting simulator is necessary. Ideally, the modeling error and the conversion error should be separated. This may not be easy to do in certain cases. Hopefully, the maximum possible modeling and conversion error are bounded and given in an explicit form to facilitate the application of the simulation results.

The quantization error and rounding errors are neglected completely in this thesis for the reason of simplicity.
Normally, these errors are relatively smaller than the modeling and conversion errors for a nonrecursive digital filter type simulator. It can be greatly reduced by using double precision representation on the computer and a proper programming technique.\textsuperscript{[26,27,28,29]} The quantization error may cause numerical instability for a recursive digital filter type simulator.\textsuperscript{[29]} A great deal of research has been done in this area.\textsuperscript{[30-34]}

When a discrete simulator has been constructed by the above steps, certain experiments may be designed to solve the problem or to help making decisions.

Here we must stress an important concept that the use of simulation to solve problems must be based on the fundamental assumption that all the phenomena of the actual system are reproducible. People have found that the application of simulation in certain fields in social science such as economical systems is often not so sucessful as in most physical science cases. This is why a good historian might not be a good politician. The same economic problem repeats many times. However, the way to solve it may be different for each time. Failure to see this could result in a disastrous decision.

Another serious error source is the unexpected stimulus or input such as the occurrence of a war for a political system or a transient surge in an electric power system. It is very important to specify the class of inputs
used in a simulation study. Thoughout this thesis, for instance, the inputs are assumed to be bounded, piecewise absolutely continuous functions of time. Other kinds of inputs are not considered.

2.3. On Approximation Theory and Spline Functions

Since simulation doesn't require the exact solution, various approximation methods can be employed to construct a digital simulator. The way to simulate a system is not unique in nature. We are interested in finding the optimal simulation scheme to solve the problem which varies due to different applications and aspects. Therefore, in what sense the term "optimal" is used must be specified in each individual case.

Approximation problems are often defined in a metric space, or specifically, in a normed linear vector space. A pair \((W, d)\) is defined where \(W\) is a set and \(d\) is a real-valued function defined for pairs of points in \(W\) in such a way that the following postulates are satisfied for all \(x_1, x_2\) and \(x_3\) in \(W\)

\[
\begin{align*}
(i) & \quad d(x_1, x_2) = 0 \\
(ii) & \quad d(x_1, x_2) \geq 0 \quad \text{for } x_1 \neq x_2 \\
(iii) & \quad d(x_1, x_2) = d(x_2, x_1) \\
(iv) & \quad d(x_1, x_2) \leq d(x_1, x_2) + d(x_2, x_3)
\end{align*}
\]  
(2.3.)
The normed linear spaces for the setting of many problems of approximation in this thesis are the function spaces of real-valued functions on a real interval \((a,b)\) for which the distance function \(d\) is one of the following settings:

(i) If \(L^1\)-norm is used, then for two functions \(x_1\) and \(x_2\) in \(W\):

\[
d(x_1, x_2) = \| x_1 - x_2 \|_{L^1(a,b)}
= \int_a^b |x_1(t) - x_2(t)| \, dt
\]  

(2.4)

(ii) Similarly, for \(L^2\)-norm:

\[
d(x_1, x_2) = \| x_1 - x_2 \|_{L^2(a,b)}
= \left\{ \int_a^b \left| x_1(t) - x_2(t) \right|^2 \, dt \right\}^{1/2}
\]  

(2.5)

(iii) For \(L^\infty\)-norm:

\[
d(x_1, x_2) = \| x_1 - x_2 \|_{L^\infty(a,b)}
= \left\{ \int_a^b \left| x_1(t) - x_2(t) \right| \, dt \right\}^{1/\infty}
= \text{Ess sup}_{t \in [a,b]} \left| x_1(t) - x_2(t) \right|
\]  

(2.6)

(iv) For the case of a certain pseudo-norm

\[
d(x_1, x_2) = \| x_1(t) - x_2(t) \|_{A}
= \left\{ \int_a^b \left| A x_1(t) - A x_2(t) \right|^2 \, dt \right\}^{1/2}
\]  

(2.7)
where \( \mathcal{A} \) is a differential operator, i.e.

\[
\mathcal{A} = \sum_{i=0}^{q} a_i(t) D^i
\]  

(2.8)

where each \( a_i(t) \) is in \( C^i[a,b] \) and \( a_q(t) \) does not vanish on \( [a,b] \).

The condition for the existence of an optimal approximation is given in the following theorem.

**Theorem 2.1.** Let \( W \) denote a compact set in a metric space.

To each point \( x \) of the space, there corresponds a point in \( W \) of minimum distance from \( x \).

**Proof:** (See [24], pp. 4). The proof is omitted here.

Since a finite-dimensional linear subspace of a normed linear space is complete, it contains at least one point of minimum distance from a fixed point. Throughout this thesis, the "optimal approximating function" always means the function which minimizes a particular distance over a prescribed subset of functions. The uniqueness of an optimal approximation depends on the convexity of the subset.

**Theorem 2.2.** In a uniformly convex normed linear space, a finite-dimensional subspace contains a unique point closest to any given point. (See [24], pp. 20)

Among the various approximation schemes employed in modeling or formulating the digital simulator, we are particularly interested in applying the spline function approximation techniques to system simulation problems. The definition and
some of its properties are stated here briefly.

Let \( x \) be an absolutely continuous and bounded real function of \( t \) and \( \mathcal{A} \) be the differential operator defined by (2.8), i.e.

\[
\mathcal{A} = \sum_{j=0}^{q} a_j(t) D^j, \quad D = \frac{d}{dt}, \text{ where } a_j(t) \in C^j[t_0, t_N]
\]

and \( a_q(t) \geq \alpha > 0 \), and let

\[
\mathcal{A}^* = \sum_{j=0}^{q} (-1)^j D^j (a_j(t) ;)
\]

(2.9)

denote the formal adjoint of \( \mathcal{A} \). Let \( a = x_0 < x_1, \ldots < x_N = b \) be any partition of \([a, b]\) with \( N \geq q-1 \). It is also called mesh or knots and denoted by \( \triangle_t \). Then, we may define the generalized spline function \( S(\cdot) \), which interpolates \( x(\cdot) \) on \( \triangle_t \) as the unique solution \( S(\cdot) \) of the problem:

(i) \( \mathcal{A}^* \mathcal{A} S(t) = 0 \) for \( t_j < t < t_{j+1}, \quad j = 0, 1, \ldots, N-1 \)

(ii) \( S(t_j) = x(t_j) \quad \text{ for } 0 \leq j \leq N \)

(iii) \( S(t) \in C^{2q-2}[a,b] \)

(iv) \( \mathcal{A}(t) = 0 \quad \text{ if } a \leq t \leq t_1, \text{ or if } t_{N-1} \leq t \leq b \) (2.10)

Note: \( \dagger \) Throughout this thesis, we assume that

\[
\left[ \max_{0 \leq i \leq N} |x_i - x_{i-1}| / \min_{0 \leq i \leq N} |x_i - x_{i-1}| \right] \triangleq \infty
\]

\( \dagger \dagger \) For period

(iv') \( S(j)(t_0) = S(j)(t_0 + kP), \quad j = 0, 1, \ldots, 2q-2, k = 1, 2, \ldots \)
The existence and uniqueness of the interpolating spline defined above has been proved by many authors. (see for example [41], pp 105). Therefore, we cite the following theorem without proof.

**Theorem 2.3.** There is a unique solution $S(t)$ of (2.10) for each mesh $\Delta_\tau = \{ a = x_0 < x_1 < x_2 \ldots < x_N = b \}$ provided $N \geq q-1$.

The minimum norm property, best approximation property and the maximum error bound of spline interpolation are stated in theorem 2.4., 2.5., and 2.6. respectively.

**Theorem 2.4.** The unique solution $S$ of (2.10) uniquely minimizes the integral

$$\int_a^b (\mathcal{A} f)^2 \, dt$$

over all $f \in \mathcal{K}^{2q}[a,b]$ such that $f(t_j) = x(t_j)$, $j = 0, 1, \ldots N$.

**Proof:** See [41], pp. 105.

**Theorem 2.5.** If $x(t) \in \mathcal{K}^q(a,b)$, the unique solution of (2.10) minimizes the integral

$$\int_a^b |\mathcal{A} x(t) - \mathcal{A} S(t)|^2 \, dt$$

over all splines of degree $2q$ on $\Delta_\tau$.

**Proof:** See [42], pp. 201
Let
\[ \| \Delta_t \| = \max_j |t_j - t_{j-1}| \] (2.13)
then for \( \| \Delta_t \| \) sufficiently small, we have:

**Theorem 2.6.** If \( x(t) \in \mathcal{K}^q[a,b] \) and \( S(t) \) is the solution of (2.10), then
\[ \| D^\alpha (x - S) \|_{L^\infty[a,b]} \leq M_\alpha \| \Delta_t \|^q \alpha - \frac{3}{2} \| A x(t) \|_{L^2[a,b]} \] (2.14)
for all \( 0 \leq \alpha < q \), where \( M_\alpha \) is a constant which is independent of \( \{ x(t_k) \}_{k=0}^N \) and \( x(t) \).

**Proof:** See [24], pp. 118

**Theorem 2.7.** If \( x \in \mathcal{K}^{2q}(a,b) \) and \( S(t) \) is the unique solution of (2.10), then for all \( 0 \leq \alpha \leq q \),
\[ \| D^\alpha (f(t) - S(t)) \|_{L^\infty[a,b]} \leq M^*_\alpha \| \Delta_t \|^{2q - \alpha - \frac{3}{2}} \] 
\[ \cdot \| A^* A x(t) \|_{L^2[a,b]} \] (2.15)
where \( M^*_\alpha \) is a constant independent of \( \{ x(t_k) \}_{k=0}^N \) and \( x(t) \).

**Proof:** See [24], pp. 122

**Note:** There are other error bounds rather than the above error bounds between the interpolating spline and the actual function. For example, let \( x(t) \in \mathcal{K}^{2q}_p (a,b) \).
Theorem 2.8. If \( x \in \mathcal{P}^{2q}(a,b) \) and \( S(t) \) is the interpolating periodic polynomial spline of order \( 2q \), then

\[
\left\| D^\alpha (x(t) - S(t)) \right\|_{L^\infty[a,b]} = q(q-1)\ldots(\alpha+1) \left[ (q-1)! \right] 2^{-(q-\alpha-1)} \int_a^b D^{2q} x(t) \, dt \
\cdot \left\| \Delta_t \right\|^{2q-\alpha-1}, \quad 0 \leq \alpha \leq q-1.
\]  

(2.16)

Proof: See [39], pp. 171

The above desirable properties (Th. 2.3.-2.8.) are the partial reason why spline functions are employed to solve approximation problems in simulation.

The other advantages such as computational advantages, high efficiency of the resulting simulators, etc. will be discussed later.

2.4. Discrete System and Computer Simulation

Since digital computers perform their calculations discretely in time, the use of a digital computer for system simulation requires that time be a discrete variable. A system under study on a computer thus necessarily becomes a discrete time system.\(^{[22]}\)

The replacement of a continuous-time system by a discrete-time system must be carried out with caution and understanding if meaningful results for the continuous-time system are to be obtained from an analysis of the discrete
Various transform methods are developed to convert the transfer function of the continuous system, \( H(s) \), to the transfer function of a discrete system, \( D(z) \). The ideal discrete system which will produce the exact output of the continuous system is given by

\[
D_0(z) = H\left(\frac{1}{T} \log z\right),
\]

where \( T \) is the sampling interval.

Unfortunately, this ideal discrete system is not physically realizable since it is not a rational function of \( z^{-1} \) when \( H \) is a rational function of \( s \).

Other transformation methods are developed to give a realizable discrete transfer function with similar properties of the continuous system such as the well-known bilinear transform or impulse invariant methods.\(^{[1,2]}\) A finite impulse response discrete system or so-called "non-recursive digital filter" can be obtained by multiplying various window functions with time domain impulse responses\(^{[58]}\) truncating Fourier coefficients of the desired frequency response, or using Discrete Fourier Transform techniques to interpolate the frequency samples.

Whenever a discrete system is obtained, the input signal \( x(t) \) can be converted to discrete form by sampling, i.e. \( x_n = x(nT) \), and the output value of the linear, time-invariant discrete system at time \( nT \) is either obtained from
the discrete convolution:

\[ y_n = x_n * h_n \]
\[ = \sum_{m=-\infty}^{\infty} x_m h_{n-m} \]
\[ = \sum_{m=-\infty}^{\infty} x_{n-m} h_m \] \hspace{1cm} (2.18)

or from inverse z-transform

\[ y_n = z^{-1} \left\{ Z\{x_n\} \cdot D(z) \right\} \] \hspace{1cm} (2.19)

An important "sampling theorem" which is not only useful for reducing the number of points which are necessary to describe a bandlimited signal, but it is also a valuable interpolation scheme is stated as follows.

**Theorem 2.9.** If the Fourier transform of a function \( x(t) \) is zero above a certain frequency \( \Omega_x \), i.e. \( X(j\omega) = 0 \) for \( |\omega| \geq \Omega_x \), then \( x(t) \) can be uniquely determined from its sampled values if \( T \leq \frac{\pi}{\Omega_x} \), i.e.

\[ x(t) = \sum_{n=-\infty}^{\infty} x(nT) \frac{\sin(\Omega_p t - n\pi)}{\Omega_p t - n\pi}, \] \hspace{1cm} (2.20)

where \( \Omega_p = \pi/T \).

Another basic representation of certain signals is the Fourier series given by
The infinite set of complex coefficients \( \{ c_n \} \) are called the spectrum of \( x \). The truncated Fourier Series

\[
\hat{x}(t) = \sum_{k=-K}^{K} c_k e^{jk\frac{2\pi}{T}t}
\]

is an optimal approximation to \( x(t) \) given in (2.21) in the sense that the mean-square-error

\[
\varepsilon = \frac{1}{T} \int_0^T |\hat{x}(t) - x(t)|^2 \, dt
\]

is minimized with respect to any other function \( \tilde{x}(t) \) which is a linear combination

\[
\tilde{x}(t) = \sum_{k=-K}^{K} \tilde{c}_k e^{jk\frac{2\pi}{T}t}
\]
CHAPTER III
SYSTEM SIMULATION BASED ON FREQUENCY
DOMAIN APPROXIMATION BY SPLINE FUNCTIONS

3.1. Introduction
3.1.1. Problem Formulation

In this chapter, we consider the problem of simulating the dynamic response of a continuous system subjected to a band-limited or bandpass signal. Traditionally, the dynamic response of a continuous system is simulated on an analog computer. However, as the systems being analyzed become more and more complex, and the required accuracy and flexibility increase, interest has grown in the use of digital computers for system simulation problems. Moreover, when we use an analog computer to simulate a system, we actually bypass one of the most essential problems for almost all kinds of simulation. The problem is that usually we do not have a complete or accurate enough mathematical model describing the physical system under simulation. It thus becomes necessary to set up a mathematical model of the physical system from certain measurements or knowledge of its characteristics, such as the frequency response at certain frequencies, and then convert it to a discrete system which can be implemented on a digital computer to perform the simulation.

We assume that the frequency response of the continuous
system is analytically known for certain frequency intervals or given in a tabular form at a finite number of frequencies for $|\omega| \leq \Omega_x$. The values are called frequency samples for the later case. We also assume that the frequency response of the continuous system is bounded and absolutely continuous over $|\omega| \leq \Omega_x$.

The input signal $x(t)$ throughout this chapter is assumed to be bandlimited or bandpass in the sense that its Fourier transform $|X(\omega)|$ vanishes outside a certain frequency range $I_x$, where $I_x$ is a set of intervals in frequency domain for $|\omega| \leq \Omega_x$ (See Fig. 3.2), i.e. $I_x = \{ \omega : \omega_i \leq \omega \leq \omega_i' \text{ for } i = 1, 2, \ldots \text{ and } |\omega| \leq \Omega_x \}$. Strictly bandlimited signals, like ideal band-limited filter, are not physically realizable because of infinite time duration required for their complete description. However, there could be essentially bandlimited signals whose Fourier transform is negligible outside a bandwidth for analysis purpose.

Although we do not know the signal in a closed form, it is realistic to assume that we can sample the input signal at any instant. Sometimes this input is a simulated one instead of the one from a real information channel.

In this chapter, the dynamic response of the system is simulated by means of a non-recursive digital filter to operate on the samples of the input. (See Figure 3.3) The reason for using a non-recursive digital filter is because
it is easy to be programmed on a computer for on-line simulation, the rounding error is relatively minor, and it offers considerable flexibility in accuracy.

Suppose that the transfer function is given in the frequency intervals $I_x$. Let us assume that the complement of $I_x$ over $[-\Omega_p, \Omega_p]$ is not a null set, i.e. the outer measure**[62].

\[
\mathcal{M}^* I_p' = \mathcal{M}^* \{ \omega : \omega \in [-\Omega_p, \Omega_p] - I_x \} > 0 \quad (3.1)
\]

One such condition, for instance is $\Omega_p > \Omega_x$, where $\Omega_p$ is one half of the sampling frequency of the discrete system, i.e.

\[
\Omega_p = \pi/T \quad (3.2)
\]

and $T$ is the sampling interval in time domain. Then, we have

\[
\mathcal{M}^*(I_p') \\
\geq \mathcal{M}^*(I(-\Omega_p, \Omega_p) - I(-\Omega_x, \Omega_x)) \\
= 2(\Omega_p - \Omega_x) \\
> 0 \quad (3.3)
\]

**The outer measure of an interval is its length.
The frequency response of a digital simulator may be arbitrary for \( \omega \in \iota_p \) since there is no signal within those frequency intervals. We will call those intervals of \( \iota_p \) **Transition Gaps**. (See Figure 3.2) There can be numerous digital simulators which can duplicate the exact output at \( t = kT \), \( |k| = 0, 1, 2, \ldots \), if the digital simulator can have an infinite number of filter weights so that the output is obtained by their convolution with the samples of the input. In other words, the output of the system subjected to a signal \( x \) can be reconstructed by the following equation:

\[
y_k = \sum_{n=-\infty}^{\infty} c_n x_{k-n}, \quad |k| = 0, 1, 2, \ldots \quad (3.4)
\]

where \( c_n \) is the weighting sequence of the non-recursive digital filter served as a digital simulator and it is obtained by taking the Fourier series expansion of a "generating function" which will be defined later.

Because the convolution operator defined by (3.4) involves an infinite sum, it is not practical because we cannot carry out an infinite number of calculations in a finite time duration. The weighting sequence \( \{c_k\}_{k=-\infty}^{\infty} \) must be truncated so that a finite duration non-recursive digital filter is obtained for physical realization (See Fig. 3.3). Therefore, a conversion error due to replacing the infinite weighting sequence \( \{c_k\}_{k=-\infty}^{\infty} \) by a truncated weighting
sequence \( \{ c_k \}_{k=-K}^K \) is induced. Since there are numerous weighting sequences \( \{ c_k \} \) which satisfy the condition defined in equation (3.4), the maximum error due to truncating is different for different weighting sequences. How to optimally construct a digital simulator so that the bound of the error due to truncation of the corresponding weighting sequence can be minimized is one of the main goals of this chapter.

3.1.2. Outline of This Chapter

In this chapter, two simulation schemes using the spline approximation are developed — one for the case in which the frequency response is defined piecewise and the other in which it is defined pointwise. The description of our approach and the associated uniqueness, existence, and convergence properties of the resulting discrete simulator are given in Section 3.2. An error analysis shows that the rate of convergence of the discrete simulator designed by the conventional Fourier series truncation method depends on the smoothness property of the desired frequency response. We call this criterion the "Principle of Smoothness" and present it with the idea of improving the Fourier series truncation method by using this principle in Section 3.3. In Section 3.4.1, we propose a new simulation scheme based on the consideration in Section 3.3 by using the spline function approximation to obtain an efficient discrete simulator.
when the desired frequency response is piecewise defined.
The optimal property of minimizing an error bound and the
 corresponding error analysis of the resulting discrete simu-
lator are derived in Section 3.5. Another numerical
 algorithm based on the periodic spline function approximation
 is developed in Section 3.4.2 to simulate a continuous system
 whose frequency response is pointwise defined. The corre-
 sponding optimal property and error analysis of the respective
discrete simulator are given in Section 3.5.2. The applica-
tions and generalization of these simulation schemes are
discussed in Section 3.6. Some numerical experiments are
presented in next chapter to consolidate the theoretical
developments in this chapter.

3.1.3. Nomenclature

We have defined certain fundamental notations which will
be used throughout this thesis in Section 2.1. To make the
terminology clear, we summarize some of the notations used
in this chapter as follows:

1) \( \Omega_X \) = the bandwidth of the input signal \( x(\omega) \), i.e.
    \[ |X(\omega)| = 0 \text{ for } \omega \geq \Omega_X, \]
    where \( x(\omega) \) is the
    Fourier transform of \( x(\omega) \).

2) \( I_X \) = the set of intervals in frequency domain for
    \[ |\omega| \leq \Omega_X \text{ such that } |X(\omega)| = 0 \text{ for } \omega \in I_X. \]

3) \( \Omega_p \) = the half-period of the frequency response of a
    discrete system. Its value is defined by
\[ \Omega_p = \pi/T, \] where \( T \) is the sampling interval and \( \Omega_p \) is assumed to be greater than \( \Omega_x \).

4) \( H(j\omega) \) = the frequency response of a physical system at \( \omega \).

Let \( H(j\omega) = H_R(\omega) + jH_X(\omega) \), where
- \( H_R(\omega) \) is the real part of \( H(j\omega) \)
- \( H_X(\omega) \) is the imaginary part of \( H(j\omega) \)

5) \( \bar{H}(\omega) \) = the generating function defined in Definition 3.2. Let
- \( \bar{H}_R(\omega) \) = real part of \( \bar{H}(\omega) \)
- \( \bar{H}_X(\omega) \) = imaginary part of \( \bar{H}(\omega) \)

6) \( \bar{S}_n(\omega) \) = an \( n \)-th order spline generating function, which will be defined later.
- \( \bar{S}_R(\omega) \) = real part of \( \bar{S}_n(\omega) \)
- \( \bar{S}_X(\omega) \) = imaginary part of \( \bar{S}_n(\omega) \)

7) \( \mathcal{H} \) = the class of all possible function \( \bar{H}(\omega) \) for a given \( H(\omega) \).

8) \( X_D(z) \) = \( z \)-transform of the sampled signal, i.e.
\[ X_D(z) = \mathcal{Z} \{ x(nT) \}. \]

9) \( Y_D(z) \) = \( z \)-transform of the samples of the output, i.e.
\[ Y_D(z) = \mathcal{Z} \{ y(nT) \}. \]

10) \( I_p \) = the frequency interval \( [-\Omega_p, \Omega_p] \)

11) \( I'_p \) = the frequency interval \( [-\Omega_p, \Omega_p] - I_x \)

12) \( I_x \) = the frequency interval \( [-\Omega_x, \Omega_x] - I_x \)
3.2. Description of The General Approach

We consider here a simulation approach based on the assumption that the frequency response of a physical system $H(j\omega)$ is analytically known for $\omega \in I_X$. These results are immediately applicable to the scheme described in Section 3.4.1 and with minor changes to the scheme presented in Section 3.4.2.

We assumed that both $H_R(\omega)$ and $H_X(\omega)$ are bounded, piecewise absolutely continuous real valued functions and we define the well-known Dirichlet Condition as follows:

**Definition 3.1.** A real, bounded, periodic function $F(\omega)$ is said to satisfy the Dirichlet Condition if in any period, it has at most a finite number of local maxima and minima and a finite number of points of discontinuity. [56]

Then, a generating function which is used to obtain the weighting sequence $\{c_k\}_{k=-\infty}^\infty$ of the digital simulator is defined below.

**Definition 3.2.** A generating function $\tilde{H}(\omega)$ is a periodic complex function of $\omega$ with the following properties:

1. $\tilde{H}(\omega) = H(j\omega)$ for $\omega \in I_X$ (3.5.a)
2. $\tilde{H}(\pm 2k \sum_p + \omega) = H(j\omega)$ for $k = 1, 2, \ldots$ (3.5.b)
3. $\tilde{H}_R(\omega)$ and $\tilde{H}_X(\omega)$ satisfy the Dirichlet Condition (3.5.c)
4. $\tilde{H}_R(-\omega) = H_R(\omega)$ for all $\omega$ (3.5.d)
(5) \( \Re_X(\omega) = -\Re_X(\omega) \) all \( \omega \) \hspace{1cm} (3.5.e)

By our assumption that \( \Omega_p \) is greater than \( \Omega_x \). (See Figure 3.4). There always exist transition gaps for \( \Re_R(\omega) \) and \( \Re_X(\omega) \) which are shown as shadowed areas in Figure 3.4.

Obviously there can be numerous generating functions for a given transfer function \( H(j\omega) \). One such family, for example, is the set of generating functions whose \( \Re_R(\omega) \) and \( \Re_X(\omega) \) are constants in those transition gaps. Our digital simulator will be a non-recursive digital filter whose weights are defined below:

**Definition 3.3.** A time domain weighting sequence \( \{c_k\}_{k=-\infty}^{\infty} \) is defined as

\[ c_k = a_k + jb_k \] \hspace{1cm} (3.6)

where

\[ a_k = \frac{1}{2\Omega_p} \int_{-\Omega_p}^{\Omega_p} \Re_R(\omega) e^{j\omega kT} d\omega \] \hspace{1cm} (3.7)

\[ b_k = \frac{1}{2\Omega_p} \int_{-\Omega_p}^{\Omega_p} \Re_X(\omega) e^{j\omega kT} d\omega \] \hspace{1cm} (3.8)

Since \( \Re_R(\omega) \) and \( \Re_X(\omega) \) satisfy the Dirichlet Condition, according to Wiley[56], both \( \{a_k\}_{k=-\infty}^{\infty} \) and \( \{b_k\}_{k=-\infty}^{\infty} \) can be obtained from term-by-term integration of equations (3.7)
and (3.8). There are certain interesting properties of 
the weighting sequence, which we shall now describe.

**Proposition 3.1.** \( \{ c_k \} \) is a real number sequence.

**Proof:** Since \( \{ a_k \} \) and \( \{ b_k \} \) are the Fourier coefficients 
of the function \( \bar{R}_R(\omega) \) and \( \bar{R}_X(\omega) \) respectively, it can be 
shown that \( \{ a_k \} \) and \( \{ j b_k \} \) are real number sequences:

\[
a_k = \frac{1}{2\Omega_p} \int_{-\Omega_p}^{\Omega_p} \bar{R}_R(\omega) e^{j\omega k T} \, d\omega
\]

\[
= \frac{1}{2\Omega_p} \left[ \int_{-\Omega_p}^{\Omega_p} \bar{R}_R(\omega) \cos \frac{2\pi \omega k}{\Omega_p} \, d\omega \\
+ j \int_{-\Omega_p}^{\Omega_p} \bar{R}_R(\omega) \sin \frac{2\pi \omega k}{\Omega_p} \, d\omega \right]
\]

\[
= \frac{1}{2\Omega_p} \int_{-\Omega_p}^{\Omega_p} \bar{R}_R(\omega) \cos \frac{2\pi \omega k}{\Omega_p} \, d\omega
\]

= real constants. \hspace{1cm} (3.9)

By the same technique, we can show that

\[
b_k = \frac{1}{2\Omega_p} \int_{-\Omega_p}^{\Omega_p} j \bar{R}_X(\omega) \sin \frac{2\pi \omega k}{\Omega_p} \, d\omega
\]

= imaginary constants \hspace{1cm} (3.10)

Therefore by definition; \( c_n = \) real constant + real constant

= real constant.
Proposition 3.2. \( a_k = a_{-k}, \quad b_k = -b_{-k}, \quad k = 0, 1, \ldots \infty \).

**Proof:** Since \( \cos \omega kT = \cos \omega (-k)T \) and \( \sin \omega kT = -\sin \omega (-k)T \), substituting the above relations into the integration of (3.7) and (3.8) leads to the above conclusion.

**Proposition 3.3.** \( a_k \) and \( b_k \) can be calculated by half-range integration, i.e.

\[
\begin{align*}
a_k &= \frac{1}{\Omega_p} \int_{-\Omega_p}^{\Omega_p} \bar{H}_R(\omega) \cdot \cos \frac{2\pi \omega k}{\Omega_p} \ d\omega \\
b_k &= \frac{-i}{\Omega_p} \int_{-\Omega_p}^{\Omega_p} \bar{H}_X(\omega) \cdot \sin \frac{2\pi \omega k}{\Omega_p} \ d\omega
\end{align*}
\]  

(3.11)  

(3.12)

**Proof:** The result follows from the fact that both \( \bar{H}_R(\omega) \cdot \cos \omega kT \) and \( \bar{H}_X(\omega) \cdot \sin \omega kT \) are even functions of \( \omega \).

Q.E.D.

The above three propositions are very helpful in simplifying the numerical integration to evaluate the sequences \( \{ a_k \} \) and \( \{ b_k \} \). In practical applications, the sampling frequency \( 2\Omega_p \) can be normalized by properly scaling the time. For example, if \( r \) is a real constant, then

\[
f(rt) \leftrightarrow \frac{1}{|r|} \ F(\frac{\omega}{r})
\]  

(3.13)

where \( f(t) \) and \( F(\omega) \) constitute a Fourier Transform pair.

Once the weighting sequence \( \{ c_k \}_{k=-\infty}^{\infty} \) is obtained for a particular generating function \( R(\omega) \), the associated
digital simulator is defined as:

**Definition 3.4.** A *K*-th degree digital simulator is a non-recursive digital filter with $2K + 1$ weights whose transfer function $D(z,K)$ is

$$D(z,K) = \sum_{k=-K}^{K} c_k z^{-k} \quad (3.14)$$

where $c_k$ is defined by equation (3.6).

The output of the $K$-th degree digital simulator is

$$\bar{y}_n = \sum_{k=-K}^{K} c_k x_{n-k} \quad (3.15)$$

It can be proved that as $K$ approaches infinity, the output of the digital simulator converges to the actual output of the physical system, i.e.

$$y_n = \lim_{K \to \infty} \bar{y}_n \quad (3.16)$$

Let us state the following lemma first:

**Lemma 3.1.** If $\bar{R}(\omega)$ satisfies the Dirichlet Condition, then the summation $\sum_{k=-\infty}^{\infty} c_k e^{j\omega kT}$ converges to $\bar{R}(\omega)$ at all points where $\bar{R}(\omega)$ is continuous and converges to the average of the right- and left-hand limits of $\bar{R}(\omega)$ at points where $\bar{R}(\omega)$ is discontinuous.

**Proof:** This is a direct application of the theorem of
Dirichlet which is proved in reference [53] and is omitted here.

**Theorem 3.1.** If the Fourier transform of the input signal $X(\omega)$ vanishes for $|\omega| \leq \Pi'_0$, then

$$y_n = \lim_{K \to \infty} \sum_{K=-K}^{K} c_k x_{n-k}$$  \hspace{1cm} (3.17)

where $c_k$ is defined by equation (3.6).

**Proof:** Let

$$\bar{Y} = \sum_{n=-\infty}^{\infty} \bar{y}_n z^{-n}$$

$$= \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} x_k c_{n-k} z^{-n}$$

$$= \sum_{n=-\infty}^{\infty} x_n z^{-n} \sum_{k=-\infty}^{\infty} c_{k-n} z^{-(k-n)}$$

$$= X_D(z) \cdot D(z, \infty)$$  \hspace{1cm} (3.19)

Therefore, the frequency spectrum of the output of a $\infty$-th degree digital simulator is

$$\bar{Y}(z) \bigg|_{z=e^{j\omega T}}$$

$$= X_D(e^{j\omega T}) \cdot D(e^{j\omega T}, \infty)$$  \hspace{1cm} (3.20)

By Sampling Theorem (See [16], pp. 51), we have
\[ X_D(e^{j\omega T}) = \mathcal{F}\left\{ x(t) \cdot \sum_{k=-\infty}^{\infty} \delta(t - kT) \right\} \]

\[ = X(j\omega) \cdot \sum_{k=-\infty}^{\infty} \delta(\omega - 2k\Omega_p) \]  

(3.21)

where \( \delta \) is the Dirac delta function. Since \( \tilde{H}(\omega) \) satisfies the Dirichlet Condition, by lemma 3.1 we get

\[ D(e^{j\omega T}, \infty) = \tilde{H}(\omega) \]  

(3.22)

Therefore,

\[ \bar{Y}(z) \bigg|_{z=e^{j\omega T}} \]

\[ = X(j\omega) \cdot \sum_{k=-\infty}^{\infty} \delta(\omega - 2k\Omega_p) \cdot \tilde{H}(\omega) \]

\[ = (X(j\omega) \cdot H(j\omega)) \cdot \sum_{k=-\infty}^{\infty} \delta(\omega - 2k\Omega_p) \]

\[ = Y(j\omega) \cdot \sum_{k=-\infty}^{\infty} \delta(\omega - 2k\Omega_p) \]

\[ = \mathcal{F}\left\{ y(t) \cdot \sum_{k=-\infty}^{\infty} \delta(t - kT) \right\} \]

\[ = Y_D(z) \bigg|_{z=e^{j\omega T}} \]  

a.e.  

(3.23)

where \( Y(j\omega) \) is the Fourier transform of the actual output of the physical system and \( Y_D(z) = Z\{y_n\} \). So we have

\[ z\{\bar{y}_n\} = z\{y_n\} \]  

(3.24)
\[ \ddot{y}_n = y_n \quad (3.25) \]

Q.E.D.

In the above theorem, it is shown that the proposed simulation scheme is exact if we take infinite number of terms of \( c_k \) convolve it with the signal samples. As previously mentioned, it is not practical to sum up an infinite number of terms to calculate the convolution summation. The weighting sequence \( \{ c_k \}_{k=\infty}^{\infty} \) is truncated for \( |k| > K \) is a \( K \)-th degree digital simulator is used. In other words, we assumed the \( c_k \) approaches zero as \( K \) approaches infinity and let \( c_k = 0 \) for \( |k| > K \). Usually \( K \), which will be called "Truncation Index", is a properly selected positive integer such that the value of \( c_k \) is negligible for \( |k| > K \). However, when the convergence of \( c_k \) to zero is slow, it might take a large number of terms to reach a desired accuracy. The resulting digital simulator is inefficient. We want to investigate the effect due to truncation and find the optimal generating function in the sense that it will reduce the effect due to truncating the infinite weighting sequence.

To solve this problem, we establish the following theorems (3.2 and 3.3) in the next section.

3.3. Principle of Smoothness and Review of Previous Applications
The problem of how good a K-th degree simulator is depends on how fast the time domain weighting sequence \( \{ c_k \} \) converges to zero as K approaches infinity. To study this problem, let us define the envelope and its magnitude of a weighting sequence:

Definition 3.5. A \( \lambda \)-th order envelope is a number sequence:

\[
\left\{ \left| k \right|^{-\lambda} \right\}_{k=1}^{\infty}
\]  
(3.26)

where \( \lambda \) is a positive integer.

Definition 3.6. For a given periodic function \( F(\omega) \in \mathcal{P}_{\nu}(0,2\Omega_p) \), the \( \lambda \)-th order envelope magnitude, denoted by \( M_\lambda \), corresponding to \( F(\omega) \) with \( \lambda \leq \nu + 1 \) is a positive constant defined as

\[
M_\lambda = \frac{\Omega_p^{2\lambda-1}}{2^n \pi^{2\lambda}} \int_0^{\Omega_p} |F(\lambda)^{\nu}(\omega)|^2 \, d\omega, \quad 1 \leq \lambda \leq \nu + 1
\]  
(3.27)

Then, we can state the following important theorem.

Theorem 3.2. If a periodic function \( F(\omega) \in \mathcal{P}_{\nu}(0,2\Omega_p) \), \( \nu \geq 0 \), and \( \{ c_k \} \) denote the time domain weighting sequence of \( F(\omega) \) defined by equation (3.6), then the constants \( M_\lambda \) defined by equation (3.27) has the property that

\[
|c_k|^\lambda \leq M_\lambda \left| k \right|^{-2\lambda}, \quad \lambda \leq \nu + 1
\]  
(3.28)

Proof: By definition 3.3 and using the integration by
parts technique, we have

\[ |c_k| = \left| \frac{1}{2\Omega_p} \int_0^{2\Omega_p} F(\omega) e^{j\omega k T} d\omega \right| \]

\[ = \left| \frac{1}{2\Omega_p} \frac{1}{jkT} \left[ F(\omega) e^{j\omega k T} \left| \begin{array}{c} 2\Omega_p \\ 0 \end{array} \right. - \int_0^{2\Omega_p} F(1)(\omega) e^{j\omega k T} d\omega \right] \right| \]

\[ = \left| \frac{1}{2\Omega_p} \left( \frac{1}{jkT} \right)^2 \left[ F(1)(\omega) e^{j\omega k T} \left| \begin{array}{c} 2\Omega_p \\ 0 \end{array} \right. \\
- \int_0^{2\Omega_p} F(2)(\omega) e^{j\omega k T} d\omega \right] \right| \]

\[ \vdots \]

\[ = \left| \frac{1}{2\Omega_p} \left( \frac{1}{jkT} \right)^{\lambda} \int_0^{2\Omega_p} F(\lambda)(\omega) e^{j\omega k T} d\omega \right| \quad (3.29) \]

Since \( T = \frac{\pi}{\Omega_p} \),

\[ |c_k| = \left| \frac{1}{2\Omega_p} \left( \frac{1}{jkT} \right)^{\lambda} \right|^2 \cdot \left| \int_0^{2\Omega_p} F(\lambda)(\omega) e^{j\omega k T} d\omega \right|^2 \]

\[ \leq \frac{1}{4\Omega_p^2} \left( \frac{\Omega_p}{\pi} \right)^{\lambda} \left| k \right|^{-2} \cdot \int_0^{2\Omega_p} \left| F(\lambda)(\omega) e^{j\omega k T} \right| d\omega \]
\[
= \frac{1}{4} \frac{\Omega_p^{2\lambda-2}}{\pi^{2\lambda}} \cdot \left\{ \int_0^{2\Omega_p} \left| \mathcal{F}(\lambda)(\omega) \right|^2 \, d\omega \right\}^{2/|k|^2} \quad (3.30)
\]

By the Schwarz's inequality, we have

\[
\left\{ \int_0^{2\Omega_p} \left| \mathcal{F}(\lambda)(\omega) \right|^2 \, d\omega \right\}^{2/|k|^2} \leq 2\Omega_p \int_0^{2\Omega_p} \left| \mathcal{F}(\lambda)(\omega) \right|^2 \, d\omega \quad (3.31)
\]

(See reference [16], pp. 63)

Hence

\[
\left| e_k \right|^2 \leq \left[ \frac{1}{2} \cdot \frac{\Omega_p^{2\lambda-1}}{\pi^{2\lambda}} \right] \cdot \int_0^{2\Omega_p} \left| \mathcal{F}(\lambda)(\omega) \right|^2 \, d\omega / |k|^{2\lambda}
\]

\[
= M_{\lambda} / |k|^2 \quad (3.32)
\]

Q.E.D.

**Collary 3.2.1.** If both \( R_R(\omega) \) and \( R_R(\omega) \) of a generating function \( R(\omega) \) are in \( \mathcal{K}^{\nu}_p(\omega) \), there exist constants \( \tilde{M}_{\lambda} \) and \( \tilde{M}_{\lambda} \) such that

\[
\left| a_k \right|^2 \leq \tilde{M}_{\lambda} / |k|^{2\lambda}, \quad \lambda \leq \nu + 1 \quad (3.33)
\]

\[
\left| b_k \right|^2 \leq \tilde{M}_{\lambda} / |k|^{2\lambda}, \quad \lambda \leq \nu + 1 \quad (3.34)
\]

**Proof:** This is a direct application of Theorem 3.2 by defining the constants \( \tilde{M}_{\lambda} \) and \( \tilde{M}_{\lambda} \) to be
\[ \bar{M}_\lambda = \frac{1}{2} \frac{\Omega^{2\lambda-1}}{\pi^{2\lambda}} \int_0^\pi \left| \bar{H}_R(\lambda)(\omega) \right|^2 d\omega \]  

\[ \tilde{M}_\lambda = \frac{1}{2} \frac{\Omega^p}{\pi^{2\lambda}} \int_0^\pi \left| \bar{H}_X(\lambda)(\omega) \right|^2 d\omega \]  

(3.35)  

(3.36)  

The inequality defined by (3.35) and (3.36) is obtained from Theorem 3.2 by letting \( F(\omega) \) to be \( \bar{H}_R(\omega) \) and \( \bar{H}_X(\omega) \) respectively.

Q.E.D.

From above collary, it is obvious that

\[ |c_k|^2 \leq (\bar{M}_\lambda + \tilde{M}_\lambda)/|k|^{2\lambda} \]

\[ = M_\lambda/|k|^{2\lambda} \quad \text{with} \quad M_\lambda = \bar{M}_\lambda + \tilde{M}_\lambda \]  

(3.37)  

Since the \( L^2 \)-norm of the \( \lambda \)-th derivative of a function is a measurement of the smoothness of the function, the above theorem can be described in a more concise, though may less precise way as the "Principle of Smoothness".

**Principle of Smoothness:** For a real periodic function, the smoother it is, the faster its time domain weighting function converges to zero.

We shall reference this general statement as "smoothness Principle" which is an alternate way to say that the source of the Gibb's oscillation [56] is reduced by increasing the continuity of the derivative of a function. This
principle was applied to the design of the ideal low-pass or band-pass non-recursive digital filters as previously mentioned in Section 1.2. For example, M. A. Martin used a segment of a sine function to connect the passband and stopband of the frequency response of an ideal low-pass or band-pass filter in a transition gap to eliminate the discontinuities in \(|H(j\omega)|\) and \(|H^{(1)}(j\omega)|\). J. Ormsby improved the convergence rate of the filter weights by using a combination of piecewise palaboli and linear segments to remove the discontinuities of the frequency response of the desired filter. In other words, they made the generating function to be in \(K^2_p [-\Omega_p, \Omega_p] \) to reduce the error due to truncation.

By Theorem 3.2, we see that just selecting a generating function in \(K^2_p [-\Omega_p, \Omega_p] \) is not satisfactory since the constant \(M_\lambda\) has not been minimized in any sense. The motivation to reduce \(M_\lambda\) is due to the investigation of an error analysis described below:

As previously mentioned, a conversion error is induced due to truncating the infinite weighting sequence \(\{c_k\}_{k=-\infty}^{\infty}\) of the non-recursive digital filter. Let us denote this truncation error of a digital simulation of degree \(K\) at frequency \(\omega\) by \(E(\omega,K)\), i.e.

\[
E(\omega,K) = H(\omega) - D(\omega,K)
\]  
(3.38)
where

\[
D(\omega, K) = \sum_{k=-K}^{K} c_k z^{-k} \bigg|_{z=e^{j\omega T}} = D_R(\omega, K) + j D_i(\omega, K) \tag{3.39}
\]

To simplify the analysis, we separate the truncation error into real part and imaginary part:

\[
E(\omega, K) = E_R(\omega, K) + j E_i(\omega, K) \tag{3.40}
\]

By definitions 3.2 and 3.4, we have

\[
E_R(\omega, K) = H_R(\omega) - D_R(\omega, K)
\]

\[
= \left( \sum_{k=-\infty}^{\infty} a_k z^{-k} - \sum_{k=-K}^{K} a_k z^{-k} \right) \bigg|_{z=e^{j\omega T}} = \left\{ \sum_{k=-\infty}^{-K} a_k z^{-k} + \sum_{k=K}^{\infty} a_k z^{-k} \right\} \bigg|_{z=e^{j\omega T}} \tag{3.41}
\]

By Proposition 3.2, we have

\[
|E_R(\omega, K)| = 2 \left| \sum_{k=K}^{\infty} a_k \cos \omega kT \right|
\]

\[
\leq 2 \sum_{k=K}^{\infty} |a_k| |\cos \omega kT|
\]

\[
\leq 2 \sum_{k=-K}^{\infty} |a_k|
\]

\[
\leq 2 \sum_{k=K}^{\infty} \tilde{M}_k^{\frac{1}{\lambda}} |k|^{\lambda}, \quad \lambda \leq \nu + 1. \tag{3.42}
\]
where $\tilde{m}_\lambda$ is a constant defined in Corollary 3.2.1. For the imaginary part, we obtain by analysis

$$E_X(\omega, K) \leq \sum_{k=K}^{\infty} 2 \tilde{m}_\lambda / |k|^\lambda, \quad 1 \leq \lambda \leq \nu + 1$$

(3.43)

Let us define a constant $\theta(\lambda, K)$ as

$$\theta(\lambda, K) = \sum_{k=K}^{\infty} |k|^{-\lambda}, \quad k > 1, \lambda \geq 2$$

(3.44)

The value of $\theta(\lambda, K)$ can be evaluated easily by a computer or estimated by the follow expression:

$$\theta(\lambda, K) < \int_{K+1}^{\infty} (x)^{-\lambda} \, dx$$

$$= \frac{1}{1-\lambda} \cdot \left(\frac{1}{K+1}\right)^{1-\lambda}, \quad \lambda > 1$$

(3.45)

$$\theta(\lambda, K) > \int_{K}^{\infty} (x)^{-\lambda} \, dx$$

$$= \frac{1}{1-\lambda} \cdot \left(\frac{1}{K}\right)^{1-\lambda}, \quad \lambda > 1$$

(3.46)

A short table is given below which shows the values of $\theta(\lambda, K)$ with $\lambda = 2, 3, 4, 5, 6$ and $K = 5, 10, 15, 20$ respectively for a quick estimation of the error bound.
\( \theta(\lambda, K) \)

<table>
<thead>
<tr>
<th>( K )</th>
<th>5</th>
<th>10</th>
<th>15</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.2208</td>
<td>0.1046</td>
<td>0.06841</td>
<td>0.05074</td>
</tr>
<tr>
<td>3</td>
<td>0.2439 \times 10^{-1}</td>
<td>0.5523 \times 10^{-2}</td>
<td>0.2373 \times 10^{-2}</td>
<td>0.1312 \times 10^{-2}</td>
</tr>
<tr>
<td>4</td>
<td>0.3571 \times 10^{-2}</td>
<td>0.3866 \times 10^{-3}</td>
<td>0.1090 \times 10^{-3}</td>
<td>0.4482 \times 10^{-4}</td>
</tr>
<tr>
<td>5</td>
<td>0.5860 \times 10^{-3}</td>
<td>0.3041 \times 10^{-4}</td>
<td>0.5626 \times 10^{-5}</td>
<td>0.1718 \times 10^{-5}</td>
</tr>
<tr>
<td>6</td>
<td>0.1022 \times 10^{-3}</td>
<td>0.2549 \times 10^{-5}</td>
<td>0.3097 \times 10^{-6}</td>
<td>0.7023 \times 10^{-7}</td>
</tr>
</tbody>
</table>

Then, the error bound is given by the following theorem:

**Theorem 3.3.** If both \( R_R(\omega) \) and \( R_X(\omega) \) of a generating function \( R(\omega) \) are in \( \mathcal{F}_\nu[I_p], \nu \geq 1 \), then the error of a \( K \)-th degree simulator is bounded by

\[
| E_R(\omega, K) | \leq 2 \tilde{M}_\lambda^{\frac{1}{2}} \theta(\lambda, K), \quad 2 \leq \lambda \leq \nu + 1 \quad (3.47a) \\
| E_X(\omega, K) | \leq 2 \tilde{M}_\lambda^{\frac{1}{2}} \theta(\lambda, K), \quad \omega \in I_x \quad (3.47b)
\]

By equation (3.40), we have

\[
| E(\omega, K) | \leq (\tilde{M}_\lambda + \tilde{M}_\lambda)^{\frac{1}{2}} \cdot \theta(\lambda, K) \\
= 2 \cdot M_\lambda^{\frac{1}{2}} \cdot \theta(\lambda, K) \quad (3.47c)
\]

where \( M_\lambda \) and \( \tilde{M}_\lambda \) are defined by (3.35) and (3.36) respectively.
The $L^2$-norm of the error is bound by next theorem.

**Theorem 3.4.** If both $\bar{H}_R(\omega)$ and $\bar{H}_X(\omega)$ of a generating function $\bar{H}(\omega)$ are in $\mathcal{K}_p^\nu[I_p]$, $\nu \geq 1$, then the $L^2$-norm of the error of a $K$-th degree simulator is bounded by

$$
\int_0^{2\Omega_p} |E_R(\omega,K)|^2 d\omega \leq 4\Omega_p \bar{m}_\nu \cdot \Theta(2\lambda,K) \tag{3.48.a}
$$

$$
\int_0^{2\Omega_p} |E_X(\omega,K)|^2 d\omega \leq 4\Omega_p \bar{m}_\nu \cdot \Theta(2\lambda,K) \tag{3.48.b}
$$

$$
\int_0^{2\Omega_p} |E(\omega,K)|^2 d\omega \leq 4\Omega_p \bar{m}_\nu \cdot \Theta(2\lambda,K) \tag{4.48.c}
$$

$$
2 \leq \lambda \leq \nu + 1
$$

**Proof:** By equation (3.41)

$$
\int_0^{2\Omega_p} |E_R(\omega,K)|^2 d\omega
$$

$$
= \int_0^{2\Omega_p} \left| 2 \sum_{k=-K}^{\infty} a_k \cos \omega k T \right|^2 d\omega
$$

$$
\leq 4\Omega_p \cdot \sum_{k=-K}^{\infty} |a_k|^2
$$

$$
\leq 4\Omega_p \bar{m}_\nu \cdot \Theta(2\lambda,K) \tag{3.49}
$$

Equation (3.48.b) can be proved by the same technique and then equation (3.48.c) will be obtained from equation (3.40).

Q.E.D.
Since $\Omega_p$ and $\Theta(2\lambda,K)$ are constants for given $T$, $\lambda$ and $K$, both bounds of the $L^\infty$- and $L^2$-norms of the error of the $K$-th degree discrete simulator will be minimized if $\tilde{M}_\lambda$ and $\tilde{M}_\lambda$ are minimized. By definition 3.6, it is equivalent to minimizing the $L^2$-norm of the $\lambda$-th derivatives of the generating functions. Although this minimization problem can be solved by numerical methods [63-66], it will be solved more efficiently by using the polynomial spline interpolation approach due to the well-known minimum norm property of interpolating spline function [39], shown in the next section.

3.4 Two New Simulation Schemes Based on the Spline Approximation in the Frequency Domain

Two new simulation schemes are proposed in this section. The first scheme as based on the polynomial spline function approximation and the second scheme employs the periodic spline function approximation.

3.4.1. Scheme I - For Simulating System with Transfer Function Specified on Frequency Intervals

Suppose the desired frequency response of a discrete simulator is defined piecewise for $\omega \in I_x$. These desired frequency responses are assumed to be in $\Phi[I_x]$. To optimally select a generating function such that the associated $M_\mu$ is minimized, let us consider the problem of constructing the real part of the generating function first. We use a polynomial spline $P_i(\omega)$ of order $2\mu + 2$ to interpolate the frequency samples in the transition gap $[\omega_i, \omega_{i+1}]$
with the end conditions satisfy:

\[ P_i(\lambda) (\omega_i) = \bar{H}_R(\lambda) (\omega_i) \quad (3.50.a) \]

\[ P_i(\lambda) (\omega_i) = \bar{H}_R(\lambda) (\omega_{i+1}^*), \quad i = 0, 1, \ldots, \nu + 1 \quad (3.50.b) \]

The existence, uniqueness and properties of the interpolating spline have been discussed in section 2.3.

The same method can be applied to construct the imaginary part of the generating function by subsituting \( H_R(\omega) \) by \( H_X(\omega) \) in equations (3.50.a) and (3.50.b). Let us denote the resulting generating function by

\[ \hat{H}(\omega) = \hat{H}_R(\omega) + j \hat{H}_X(\omega) \quad (3.51) \]

A time domain weighting sequence \( \{ \hat{c}_k \} \) can be obtained by substituting equation (3.51) in to equation (3.6). The resulting K-th degree digital simulator is a non-recursive digital filter whose transfer function is given by

\[ \hat{D}(z,K) = \sum_{k=-K}^{K} \hat{c}_k z^{-k} \quad (3.52) \]

The time domain output is defined by

\[ \hat{y}_k = \sum_{i=-K}^{K} \hat{c}_k x_{k-i} \quad (3.53) \]
For on-line simulation problems, this digital simulator is realized as

$$\hat{D}(z, K) = \sum_{k=0}^{2K+1} c_{k-K} z^{-k}$$

(3.54)

In other words, the output is delayed by additional K pulses so that a causal discrete system can be obtained. The configuration of the discrete simulator is shown in Figure 3.3 The optimal property and error analysis of this scheme will be presented in section 3.5.1.

3.4.2. Scheme II — For Simulating System with Transfer Function Specified at Discrete Points

3.4.2.a. Method Description

Now, we study the simulation problem for the second assumption — that is the frequency response of the physical system is given in a tabular form at a finite number of frequencies \( \{ \omega_i \}^N_{i=0} \), \( N > 1 \), with the property that

\[ 0 = \omega_0 < \omega_1 < \omega_2 \ldots < \omega_N \leq \Omega_x \text{ and } \omega_i \in I_x, \text{ all } i. \]

Two periodic polynomial splines \( S_R(\omega) \) and \( S_X(\omega) \) of degree \( 2q-1 \), \( q \geq 1 \) are employed to approximate the real and imaginary parts of the actual frequency response on the mesh \( \Delta'_{2N} \) defined by:

\[ \Delta'_{2N} = \{ 0 \leq x_0 < x_1 < \ldots < x_{2N} < 2\Omega_p \} \], \text{ such that } \omega_{2N-i} = \omega_i, \quad i = 1, \ldots, N - 1 \]

(3.55)
We also assume that we know \textit{a-priori} that the frequency response of the physical system belongs to \( \mathcal{K}^{2q}(0, \Omega_x) \), \( q \geq 1 \). Then, the spline generating function is defined as:

**Definition 3.7.** A \( 2q \)-th order spline generating function \( \tilde{S}_{2q}(\omega) \) is a periodic complex function of \( \omega \), i.e. \( \tilde{S}_{2q}(\omega) = \tilde{S}_R(\omega) + j \tilde{S}_X(\omega) \), where \( \tilde{S}_R(\omega) \) and \( \tilde{S}_X(\omega) \) are real functions of \( \omega \), which satisfy the following conditions.

1. \( \tilde{S}_R(2q)(\omega) = 0 \) and \( \tilde{S}_X(2q)(\omega) = 0 \) \hspace{1cm} (3.55.a)

   for \( \omega \leq (0, 2\Omega_p) \) except at the knots \( \omega = \pm \omega_i \), \( i = 1, \ldots, \infty \).

2. \( \tilde{S}_R(\omega) \in C^{2q-1}(-\infty, \infty) \), \( \tilde{S}_X(\omega) \in C^{2q-1}(-\infty, \infty) \) \hspace{1cm} (3.55.b)

3. \( \tilde{S}_R(\omega_i) = \tilde{H}_R(\omega_i) \)
   \[ = H_R(\omega_i) \quad i = 0, 1, \ldots, N \]

   \( \tilde{S}_X(\omega_i) = \tilde{H}_X(\omega_i) \)
   \[ = H_X(\omega_i) \quad i = 0, 1, \ldots, N \]

   \( \tilde{S}_R(\omega_i) = \tilde{H}_R(\omega_i) \)
   \[ = H_R(\omega_{2N-i}) \quad i = N + 1, \ldots, 2N \]

   \( \tilde{S}_X(\omega_i) = \tilde{H}_R(\omega_i) \)
   \[ = -H_X(\omega_{2N-i}) \quad i = N + 1, \ldots, 2N \] \hspace{1cm} (3.55.c)
4. \( \tilde{S}_R(\pm 2k\Omega_p + \omega) = \tilde{S}_R(\omega) \)

\( \tilde{S}_X(\pm 2k\Omega_p + \omega) = \tilde{S}_X(\omega), \quad k = 1, 2, \ldots, \infty \quad (3.55.d) \)

Let the time domain weighting sequence associated with the spline generating function \( \tilde{S}(\omega) \) to be denoted as \( c_k^* \), i.e.

\[ c_k^* = a_k^* + j \ b_k^* \quad (3.57) \]

where

\[ a_k^* = \frac{1}{2\Omega_p} \int_0^{2\Omega_p} \tilde{S}_R(\omega) \cdot e^{j\omega kT} \ d\omega \quad (3.58.a) \]

\[ b_k^* = \frac{1}{2\Omega_p} \int_0^{2\Omega_p} \tilde{S}_X(\omega) \cdot e^{j\omega kT} \ d\omega \quad (3.58.b) \]

Since both \( \tilde{S}_R(\omega) \) and \( \tilde{S}_X(\omega) \) satisfy the Dirichlet condition, there is a unique weighting sequence \( \{ c_k^* \} \) defined by (3.56). The resulting K-th degree simulator is a (2K - 1)-th order non-recursive digital filter whose transfer function is denoted by

\[ D^*(z,K) = \sum_{k=-K}^{K} c_k^* z^{-k} \quad (3.59) \]

The simulated output at time \( t = kT \) of the K-th degree simulator is obtained by
\[ y_k = \sum_{i=-K}^{K} c_k^i x_{k-i} \]  \hfill (3.60)

A very efficient algorithm is developed below to calculate \( \{c_k^i\} \) if \( \bar{S}(\omega) \) is obtained.

3.4.2.b. An Algorithm to Calculate the Time Domain Weighting Sequence

In practical application, the weighting sequence may be calculated by the following method:

Suppose \( \bar{S}(\omega) = \bar{S}_R(\omega) + j \bar{S}_X(\omega) \) is the spline generating function. Due to the fact that \( \bar{S}_R(\omega) \) and \( \bar{S}_X(\omega) \) are in \( \mathbb{K}_p^{2q-1}[0,2\Omega_p] \), \( \bar{S}_R(2q-1)(\omega) \) and \( \bar{S}_X(2q-1)(\omega) \) are piecewise step functions. Therefore,

\[ \bar{S}_R(2q)(\omega) = \sum_{i=0}^{2N} A_i \delta(\omega - \omega_i) \]  \hfill (3.61)

\[ \bar{S}_X(2q)(\omega) = \sum_{i=0}^{2N} B_i \delta(\omega - \omega_i) \]  \hfill (3.62)

where \( \delta \) are Dirac delta function and \( A_i \) and \( B_i \) are constants.

By equation (3.73), and (3.74), we have

\[ a_k^i = \frac{\Omega_p^{2q-1}}{\pi^{2q}} \int_0^{\Omega_p} \bar{S}_R(2q)(\omega) \cos k\omega T \, d\omega \]

\[ = \frac{\Omega_p^{2q-1}}{(\pi k)^{2q}} \sum_{k=0}^{N} A_i \cos k\omega_i T \]  \hfill (3.63.a)
\[ b_k^* = \frac{\Omega_p}{(\pi k)^{2q}} \int_0^{\Omega_p} \mathbb{G}_R^{(2q)}(\omega) \sin k \omega_1 T \, d \omega \]
\[ = \frac{\Omega_p}{(\pi k)^{2q}} \sum_{k=0}^{N} B_i \sin k \omega_1 T \]  \hspace{1cm} (3.63.b)

This is a desirable result since no numerical integration is needed to compute the value of \( \{ c_k^* \} \).

3.5. Optimal Property and Error Analysis

3.5.1. Optimal Property and Error Analysis for Scheme I

Since the real part and imaginary part of the generating function \( \hat{H}(\omega) \) obtained by Scheme I method are in \( \mathcal{L}^2[0, 2\Omega_p] \). By theorem 3.2, the corresponding weighting sequence \( \{ \hat{a}_k \} \) and \( \{ \hat{b}_k \} \) are bounded respectively by

\[ |\hat{a}_k|^2 \leq \hat{M} / |k|^{2\nu} \]  \hspace{1cm} (3.64.a)

\[ |\hat{b}_k|^2 \leq \hat{M} / |k|^{2\nu} \]  \hspace{1cm} (3.64.b)

where

\[ \hat{M}_\nu = a_\nu \int_0^{2\Omega} \left| \hat{H}_R(\nu)(\omega) \right|^2 d\omega \]  \hspace{1cm} (3.65)

\[ \hat{M}_\nu = a_\nu \int_0^{2\Omega} \left| \hat{H}_X(\nu)(\omega) \right|^2 d\omega \]  \hspace{1cm} (3.66)
\[ a_\nu = \frac{\Omega_p^{2\nu-1}}{2 \pi^{2\nu}} \]  

(3.67)

Due to the well-known smoothness property of spline function interpolation stated in theorem 2.4, we know that

\[ \int_{I_X} |\hat{H}_R(\nu)(\omega)|^2 \, d\omega \]  

(3.68)

and

\[ \int_{I_X} |\hat{H}_X(\nu)(\omega)|^2 \, d\omega \]  

(3.69)

are minimized over all functions \( F(\omega) \in \mathcal{K}^n(I_X) \). [39]

Therefore, we have the following theorem.

Theorem 3.5. If \( H_R(\omega) \in \mathcal{K}^\nu(I_X) \), then, the \( \nu \)-th envelope magnitude \( \hat{M}_\nu \) defined by equation (3.56) is minimized over those of all possible \( H_R(\omega) \in \mathcal{K}_p^{\nu} [0, 2\Omega_p] \). The same property holds for the imaginary part.

Proof. By definition

\[ \hat{M}_\nu = a_\nu \int_0^{2\Omega_p} |H_R(\nu)(\omega)|^2 \, d\omega \]

\[ \leq a_\nu \int_{I_X} |H_R(\nu)(\omega)|^2 \, dx + a_\nu \int_{I_X} |H_R(\nu)(\omega)|^2 \, d\omega \]

(3.70)

Since the first term of the above equation is a constant and
\[ \int_{I_p} \left| \hat{H}_R(\nu)(\omega) \right|^2 d\omega = \min_{\hat{H}_R(\omega) \in \mathcal{K}_p(\nu)} \int_{I_p} \left| R_R(\nu)(\omega) \right|^2 d\omega \]

(3.71)

The conclusion follows. The imaginary part can be proved in an analogous way.

Q.E.D.

By theorem 3.3, the error bound is directly proportional to the value of \( (\overline{M} + \tilde{M}) \). Therefore, theorem 3.4 reveals that the error bound of the frequency response designed by Scheme I is minimized.

Due to theorem 3.4, the above theorem also disclosed that the bound of the \( L^2 \)-norm of the error of the resulting discrete simulator is minimized in frequency domain since \( \overline{M} \) and \( \tilde{M} \) are minimized.

Now we want to study the error bound in time domain. Let \( \hat{e}_k \) denote the error between the actual output of Scheme I and the simulated output of a \( K \)-th degree simulator:

\[ \hat{e}_k = |y_k - \hat{y}_k| \quad (3.72) \]

where \( \hat{y}_k \) is defined by equation (3.53).

Suppose that the Fourier transform of the band-limited signal \( x(t) \) is \( X(\omega) \) and \( \int_{-\Omega_x}^{\Omega_x} |X(\omega)| \; d\omega < \beta < \infty \), then we have:

**Theorem 3.6.** If both \( R_R(\omega) \) and \( R_X(\omega) \) are in \( \mathcal{K}_p^J[0, 2\Omega_p] \),
the time domain error $\hat{e}_k$ defined by (3.72) is bounded by

$$\hat{e}_k \leq \frac{2^{\frac{1}{K}} \cdot \theta(\lambda,K)}{\Omega_p} \cdot \int_{x} |X(\omega)| \, d\omega, \quad 2 \leq \lambda \leq \nu + 1$$

(3.73)

where $\hat{M}_\lambda = \hat{M}_\lambda + \hat{N}_\lambda$

**Proof:** By definition

$$\hat{e}_k = |y_k - \hat{y}_k|$$

$$= \left| \frac{1}{2\Omega_p} \int_{\Omega_p} \hat{H}(\omega) X(\omega) e^{j\omega kT} d\omega - \frac{1}{2\Omega_p} \int_{\Omega_p} \hat{D}(\omega,K) X(\omega) e^{j\omega kT} d\omega \right|$$

$$\leq \frac{1}{2\Omega_p} \int_{\Omega_p} |\hat{H}(\omega) - \hat{D}(\omega,K)| \cdot |X(\omega)| e^{j\omega kT} d\omega$$

$$\leq \frac{1}{\Omega_p} \sup_{\omega \in \Omega_p} |\hat{H}(\omega) - \hat{D}(\omega,K)| \cdot \int_{0}^{\Omega_p} |X(\omega)| d\omega$$

$$\leq \frac{2^{\frac{1}{k}} \cdot \theta(\lambda,K)}{\Omega_p} \cdot \int_{x} |X(\omega)| \, d\omega, \quad \lambda = \nu + 1$$

(3.74)

From the above theorem for a given $\nu$ and $K$, the error bound in time domain is also minimized since $\hat{M}_\lambda$ is. Moreover, this error bound may be further reduced by increasing
$\nu$ and $K$ to reduce the error due to truncation.

3.5.2. Optimal Property and Error Analysis for Scheme II

The optimal property and error analysis for Scheme II are similar to those of Scheme I. Suppose that

$$\tilde{R}^*(\omega) = \tilde{R}^*_R(\omega) + j \tilde{R}^*_X(\omega)$$

(3.75)

is a generating function which interpolates the frequency samples of $\tilde{R}(\omega)$ on $\Delta_{2N}$ defined in (3.66) and both $\tilde{R}^*_R(\omega)$ and $\tilde{R}^*_X(\omega)$ are in $\mathcal{K}^q_p[0,2\Omega_p]$. Let $\tilde{M}^*_\lambda$ and $\tilde{M}^*_X$ denote the magnitude of $\lambda$-th order envelope of $\tilde{R}^*_R(\omega)$ and $\tilde{R}^*_X(\omega)$ respectively. By definition, for $\nu \leq 2q + 1$

$$\tilde{M}^*_\lambda = \frac{1}{2} \Omega_p^{2\lambda - 1} \frac{2\Omega_p}{\pi} \int_0^\infty |\tilde{R}^*_R(\lambda)(\omega)|^2 d\omega$$

(3.76)

$$\tilde{M}^*_X = \frac{1}{2} \Omega_p^{2\lambda - 1} \frac{2\Omega_p}{\pi} \int_0^\infty |\tilde{R}^*_X(\lambda)(\omega)|^2 d\omega, \lambda \leq 2q$$

(3.77)

Then, we have Theorem 3.7. Let $\Delta_{2N}$: $0 = x_0 < x_1 < \ldots < x_{2N} < 2\Omega_p$ be a given mesh defined by (3.55). Then of all functions $\tilde{R}^*_R(\omega) \in \mathcal{K}^q_p[0,2\Omega_p]$ which interpolate to $\{\tilde{R}^*_R(\omega_i)\}^N_{i=0}$, $\tilde{S}^*_R(\omega)$ defined by (3.55) uniquely minimizes the corresponding $\tilde{M}^*_q$. The imaginary part $\tilde{S}^*_X(\omega)$ has the same property as the real part.

Proof: This is a direct application of theorem 2.4 by
replacing the differential operator with \( \left( \frac{d}{d\omega} \right)^q \). Then, among all functions \( \tilde{H}_R^i(\omega) \) in \( \mathcal{K}_p^{2q} [0, 2\Omega_p] \) which interpolate to \( \{ \tilde{H}_R(\omega_i) \}_{i=0}^K \), \( \tilde{S}_R(\omega) \) is the unique function which minimizes

\[
\int_0^{2\Omega_p} |\tilde{H}_R^{(q)}(\omega)|^2 \, d\omega
\]

(3.78)

The imaginary part can be proved by analogy.

These optimal properties and the computational advantage given in 3.4.2.b give us a picture of the highlights of using spline function to interpolate the frequency samples.

To evaluate the error bound for Scheme II, we consider the two major error sources — the approximation error (modeling error) and the truncation error (conversion error) as previously mentioned in Chapter II.

Let us denote the generating function in \( \mathcal{K}_p^{2q} [0, 2\Omega_p] \) which has the property that the L\(^1\)-norm of its 2q-th derivative over the period \([0, 2\Omega_p]\) is globally minimized among all other generating functions in \( \mathcal{K}_p^{2q} [0, 2\Omega_p] \) as

\[
\tilde{H}_{2q}(\omega) = \tilde{H}_R(\omega) + j \tilde{H}_X(\omega)
\]

(3.79)

where \( \tilde{H}_R(\omega) \) and \( \tilde{H}_X(\omega) \) are the real and imaginary part of \( \tilde{H}_{2q}(\omega) \).
Then, we define the approximation error of a $2q$-th order spline generating function to $\overline{S}_{2q}(\omega)$ as

$$A^*(\omega) = \overline{S}_{2q}(\omega) - \overline{S}_{2q}(\omega) = A_R^*(\omega) + j A_X^*(\omega)$$  \hspace{1cm} (3.80)

where $A_R^*(\omega)$ and $A_X^*(\omega)$ are real functions.

By theorem 2.7, for $\omega \in \Omega_X$, we have

$$|A_R^*(\omega)| \leq q! (q-1)! 2^{-(2q-1)} p \int_0^{2\Omega} |\overline{H}_R(2q)(\omega)| d\omega \cdot \|\Delta_{2N}^i\|^{2q-1}$$

$$= \tilde{\mu}_q \cdot \|\Delta_{2N}^i\|^{2q-1}$$ \hspace{1cm} (3.81)

Similarly

$$|A_X^*(\omega)| \leq q! (q-1)! 2^{-(2q-1)} p \int_0^{2\Omega} |\overline{H}_X(2q)(\omega)| d\omega \cdot \|\Delta_{2N}^i\|^{2q-1}$$

$$= \tilde{\mu}_q \cdot \|\Delta_{2N}^i\|^{2q-1}$$ \hspace{1cm} (3.82)

Let the error due to truncation be denoted by

$$T^*(\omega, K) = \overline{S}_{2q}(\omega) - D^*(z, K) \bigg|_{z = e^{j\omega T}}$$

$$= T_R^*(\omega, K) + j T_X^*(\omega, K)$$ \hspace{1cm} (3.83)

where $D^*(z, K)$ is defined by equation (3.59).
From theorem 3.3, we have

$$T_R^*(\omega, K) \leq 2 M^{1/2}_\lambda \theta(\omega, K)$$  \hspace{1cm} (3.84)

$$T_X^*(\omega, K) \leq 2 \tilde{M}^{1/2}_\lambda \theta(\omega, K)$$  \hspace{1cm} (3.85)

where $M^*$, $M'$ and $\theta(\lambda, K)$ are defined by equations (3.76), (3.77) and (3.44) respectively. Let us denote the error in frequency domain by

$$E^*(\omega, K) = E_R^*(\omega, K) + j E_X^*(\omega, K)$$  \hspace{1cm} (3.86)

Then, we have

**Theorem 3.8.** The error of a $K$-th order discrete simulator designed by Scheme II method using a $2q$-th spline generating function is bounded by

$$E^*(\omega, K) \leq \tilde{E}^* \quad \omega \in I_X$$  \hspace{1cm} (3.87)

where $\tilde{E}^*$ is a constant defined as

$$\tilde{E}^* = \left[ \left[ \mu_q \| \Delta_{2N}^{2q-1} \| + 2 M^{1/2}_\lambda \cdot \theta(\lambda, K) \right]^2 + \left[ \mu_q \| \Delta_{2N}^{2q-1} \| + 2 \tilde{M}^{1/2}_\lambda \cdot \theta(\lambda, K) \right]^2 \right]^\frac{1}{2}$$  \hspace{1cm} (3.88)

**Proof:** Since

$$E_R^*(\omega, K) = A_R^*(\omega) + T_R^*(\omega, K)$$  \hspace{1cm} (3.89)
\[ |E_R^r(\omega, K)| \leq |A_R^r(\omega)| + |T_R^r(\omega, K)| \]

\[ \leq \mu_q \| \Delta \|_{2N}^{2q-1} + 2 \tilde{M}_\lambda^{\frac{1}{2}} \cdot \theta(\lambda, K) \]  

(3.90)

Similarly,

\[ |E_X(\omega, K)| \leq \mu_q \| \Delta \|_{2N}^{2q-1} + 2 \tilde{M}_\lambda^{\frac{1}{2}} \cdot \theta(\lambda, K) \]  

(3.91)

The conclusion is obtained from equations (3.86), (3.90) and (3.91).

Q.E.D.

As for the error bound in time domain, it can be obtained directly from the above theorem. Suppose that

\[ \int_0^{\Omega} |X(\omega)| \, d\omega < \infty, \]

then the time domain error \( e^r_k \) is bounded by

\[ e^r_k = |y_k - y^r_k| \]

\[ = \frac{1}{\Omega} \left| \int_0^{\Omega} \bar{H}(\omega) X(\omega) e^{j\omega T} \, d\omega \right| \]

\[ - \int_0^{\Omega} D^r(e^{j\omega T}, K) X(\omega) e^{j\omega T} \, d\omega \right| \]

\[ \leq \frac{1}{\Omega} \int_0^{\Omega} |\bar{H}(\omega) - D^r(e^{j\omega T}, K)| \cdot |X(\omega)| \, d\omega \]

\[ \leq \frac{E}{\Omega} \int_0^{\Omega} |X(\omega)| \, d\omega \]  

(3.92)
Therefore, both simulation errors are bounded and the bounds can be reduced by increasing $q, K$ and decreasing $\| \Delta_{2N} \|.$

3.6. Generalization and Conclusion

Although the simulation techniques described in this chapter are based on the assumption that the input signal is band-limited or band-pass signal, the scheme can be used to the case that the signal is not strictly band-limited. In this case, the simulation error may increase and certain judgement is required to evaluate the results of simulation.

An interesting problem of whether can we use generalized splines instead of the polynomial spline or not is answered below: the generalized spline can be applied to the simulation problems by changing the basis functions of the polynomial spline to be the set of linearly independent functions 

\[
\{ \Phi_i(t) \}_{i=1}^{2q} \] 

such that $\mathbf{A} \mathbf{A}^T \Phi(t) = 0$, where $\mathbf{A}$ is defined by (2.8). By doing so, the resulting generalized spline function will be smoothest in the sense that $\int_0^{2\Omega} | \hat{\Phi}^2(q \omega) |^2 d\omega$ is minimized. The corresponding error analysis and optimal properties can be obtained in an analogous fashion by replacing the differential operator $\mathbf{A} = d^q/d\omega^q = D^q$, of polynomial splines by $\mathbf{A} = \sum_{i=0}^{q} \alpha_i D^i$. These optimal properties are not so easy to interpret as in the polynomial spline approximation case.

In this chapter, two simulation schemes based on
frequency response approximating by spline functions have been proposed. The first one is for the case that the frequency response of the physical system is piecewise defined over the range of interest: \( \omega \in I_x \). The second case is that the given frequency response is given in a tabular form. Since the physical system is subjected to a band-limited signals whose value is defined only at sampling instants, a non-recursive digital filter type simulator is used because it gives an excellent magnitude and phase control over the range of interest. Both problems are solved very efficiently by the spline approximation with the optimal property that certain bound of the error due to converting the continuous system to a discrete system is minimized. Several numerical examples are given in next chapter to illustrate the method developed here.
Fig. 3.1 Block Diagram of System Simulation by Digital Computer
Fig. 3.2. Bandpass Signal and Transition Gaps in the Frequency Domain
Fig. 3.3. Configuration of Discrete Simulator

(a) Off-Line Simulation Scheme

(b) On-Line Simulation Scheme
CHAPTER IV

NUMERICAL EXPERIMENTS ON SYSTEM SIMULATION

PROBLEMS FROM FREQUENCY DOMAIN APPROACH

4.1. Numerical Examples

Three numerical examples were studied on the Rice Burroughs 5500 computer. Example 1 illustrates the highlights of applying the Schemes I and II proposed in Chapter III to simulate a physical system whose frequency response is either defined on disjoint frequency intervals or at discrete frequencies. Examples 3 and 4 show that the application of Scheme I to the non-recursive digital filter design problems turns out to be efficient and satisfactory too.

The frequency responses of the digital simulators designed by our schemes are compared with those results designed by the other three existing methods. They are the conventional Fourier series coefficient truncation method \[^{2,59}\], discrete Fourier transform method \[^{1,72}\] and the Chebyshev approximation method \[^{51,70,76}\]. These methods will be called simply the Fourier method, the DFT method and the Chebyshev method respectively. For the reason of simplicity, we will call the resulting discrete simulators designed by these methods and our schemes the Fourier simulator, the DFT simulator, the Chebyshev simulator, the Scheme I simulator and the Scheme II simulator etc.
**Example 4.1.** In our first example, the physical system has the transfer function

\[
H(\omega) = \frac{1}{1 + 4\omega^2} + j \frac{2\omega}{1 + 4\omega^2}
\]

(4.1)

The input signal \(x(t)\) is band-limited such that \(|X(\omega)| = 0\) for \(|\omega| > \Omega_x = 2.8274\) and the sampling interval \(T\) is 1.

Figure 4.1 shows the real and imaginary parts of the desired frequency response of the discrete simulator. It also gives the 20 equispaced frequency samples for the Scheme I and DFT methods. Figures 4.2 and 4.3 show respectively the \(L^\infty\)- and \(L^2\)-norms of the real part of the frequency domain errors of the discrete simulators designed by Scheme I, Scheme II, Fourier and DFT methods at different filter lengths. Figures 4.4 and 4.5 are the \(L^\infty\)- and \(L^2\)-norms of the errors of the imaginary part. Figure 4.6 shows the time domain weighting sequences of the 21st order discrete simulators designed by these four different methods.

This example shows clearly that the errors associated with our schemes converges to zero fast.

**Example 4.2.** An ideal low-pass filter whose pass-band is \([0,0.2]\) and stop-band is \([0.3,0.5]\) is designed by Scheme I using a cubic spline in the transition gap \([0.2,0.3]\).

The sampling frequency is normalized to one.

This example has been widely discussed by many
authors\[2,3,70,71,76\]. Our results are presented together with the results of the Fourier, the DFT and the Chebyshev methods for comparison. These results are shown in Figures 4.7, 4.8 and 4.9 respectively.

Note that the error of the DFT simulator would not improve as the number of filter length increases. The error of the Fourier simulator goes down slowly. The error of the Scheme I simulator has a convergence rate (to zero) which is comparable to the convergence rate of the chebychev simulator as the filter length increases.

One principal merit of our method is that the time required to evaluate the coefficients of the discrete simulator on the digital computer is extremely short. It takes about 1.2 seconds to obtain all coefficients for 50 discrete simulators with length ranging from 3 to 101 by our Scheme I on the Rice Burroughs 5500 computer. (See Appendix B). However, to design a single Chebyshev simulator with length 95 by Parks-McClellan approach\[70\] would require about 200 seconds on the same computer. Another merit of our method can be explained by the next example.

Example 4.3. A band-eliminate filter with the frequency response defined below is required

\[
D(\omega) = \begin{cases}
0.6 & \omega \in [0,0.31] \\
0.0 & \omega \in [0.15,0.5] \\
1.0 & \omega \in [0.35,0.5] 
\end{cases}
\] (4.2)

Note: † See the note in page 4
The sampling frequency is normalized to be one. This filter would be difficult to be designed by the Chebyshev method [59,70]. The error of the DFT simulator for this example is very significant so that it is not applicable either. Therefore, only the Fourier simulator and the Scheme I simulator are designed and the results are shown in Figures 4.10, 4.11 and 4.12 respectively. It is obvious that the $L^\infty$- and $L^2$-norms of the error of the Scheme I simulator converges to zero much faster than that of the Fourier simulator when the length of the digital filter increases.

4.2. Comparison and Conclusion

Although most of the digital filter design methods can be applied to system simulation problems, few of them would give us satisfactory results for the actual application. The main reason is because most of the existing digital filter design methods assume that the desired frequency response of the digital filter is completely defined. That is an unrealistic condition for system simulation problems as previously discussed in Chapters II and III.

Among various digital filter design techniques, the one which is called "Frequency Sampling Method" is an applicable approach. The method proposed by L. Rabiner and R. Schafer [50] is based on Discrete Fourier Transform Techniques to allow the designer to specify values of the filter's frequency responses at equispaced frequencies, and
hence derive an approximation to the desired continuous frequency response. The merit of this method is that it automatically generates a simulator with a finite time duration impulse response and zero error at the sampling frequencies. However, the frequency responses between two interpolating points may differ significantly from the desired frequency response. Therefore, the feasibility of this method which depends upon individual case is determined empirically. A more general way to simulate a given physical system by using sine and cosine functions to interpolate those frequency samples is described briefly in Appendix A.

When the frequency response of a system is piecewise defined, the recently developed non-recursive digital filter design method by Chebyshev approximation may be used to obtain an optimal discrete simulator whose frequency response error is minimized in Chebyshev sense. O. Hermann[3], T. W. Parks[70] and E. Hofstetter[71] developed some numerical algorithms to solve the problem of finding a best approximation to the ideal low-pass filter in Chebyshev sense. However, when the number of transition gaps increases and the shape of the desired frequency response varies, it might become complicated and difficult to obtain the optimal non-recursive digital filter by these methods. Even with the simple low-pass filter design case, the algorithm to search the optimal Chebyshev approximation is time-consuming.
As previously discussed in Example 4.2, it would require about 200 seconds to design a length 95 filter by Parks-McClellan algorithm on Rice Burroughs 5500 computer. However, 50 non-recursive digital filters with length ranging from 3 to 101 will be designed within 1.2 seconds by Scheme I and 4.5 seconds by Scheme II on the same computer (See the programs in FORTRAN IV listed in Appendix B and C).

From these numerical examples, we may conclude that the proposed spline simulation schemes are simple and efficient methods to solve system simulation problems. As previously mentioned, the convergence rates of the $L^\infty$- and $L^2$-norms of the error of the spline simulator are much better than those of the DFT simulator and Fourier simulator. Another valuable application is to the design of non-recursive digital filters when the desired frequency response is defined on disjoint frequency intervals. The resulting digital filter is a sub-optimal one in the sense that certain error bound is minimized.
Example 4.1

Fig. 4.1 Frequency Response of the Physical System

Magnitude

Frequency Samples of the Real Part of $H(j\omega)$

Frequency Samples of the Imaginary Part of $H(j\omega)$
Fig. 4.2 \( \| \text{Error for Real Part} \|_{L^\infty[I_x]} \) vs. Filter Length
Example 4.1

- △ Fourier Simulator
- □ Scheme I Simulator
- ○ Scheme II Simulator
- ◆ DFT Simulator

Fig. 4.3 \( \| \text{Error for Real Part} \|_{L^2[I_x]} \) vs. Filter Length
Fig. 4.4 \[ \|\text{Error for Imaginary part}\|_{L^2[I_x]} \text{ vs. Filter Length} \]
Fig. 4.5 $\|\text{Error for Imaginary Part}\|_{L^2[I_x]}$ vs. Filter Length
Example 4.1

Fig. 4.6 Magnitude of the Time Domain Impulse Response
Filter Length = 21
Example 4.2

\[ \text{Magnitude} \]

\[ \triangle \text{ Fourier Simulator} \]
\[ \square \text{ Scheme I Simulator} \]
\[ \bigcirc \text{ DFT Simulator} \]

\[ 0.0 \quad 0.1 \quad 0.2 \quad 0.3 \quad 0.4 \quad 0.5 \]

\[ -0.4 \quad -0.2 \quad 0.0 \quad 0.2 \quad 0.4 \quad 0.6 \quad 0.8 \quad 1.0 \quad 1.2 \]

Normalized Frequency

**Fig. 4.7** Frequency Response of Low-pass Filter,
Filter Length = 19 (The sample points shown here are some from more points actually used in the computation, and they are picked to emphasize the important feature of the curves plotted).
Fig. 4.8. \( \| Error \|_{L^\infty[I_X]} \) vs. Filter Length
Fig. 4.9 $\|\text{Error}\|_{L^2[I_x]}$ vs. Filter Length
Example 4.3

Fig. 4.10. Frequency Response of Band-Eliminate Filter, Filter Length = 21
Fig. 4.11 $\|\text{Error}\|_{L^\infty[I_x]}$ vs. Filter Length
Fig. 4.12 $\|\text{Error}\|_{L^2[L_x]}$ vs. Filter Length
CHAPTER V
TIME DOMAIN SIMULATION
BASED ON THE SPLINE APPROXIMATION

5.1 Problem Formulation

The attention in this chapter will be restricted to the simulation of a given linear, time-invariant asymptotically stable system using a time domain approach. The scheme developed here will be for off-line simulation as in the case of seismic signal processing; the input signals are recorded on magnetic tape and the time period of interest is limited from zero to a few seconds. The problem may be described as follows:

Suppose we are given a system:

\[ w(t) = Aw(t) + Bx(t) \]
\[ y(t) = Cw(t) \]  \hspace{1cm} (5.1)

where \( w(t) \), \( x(t) \) and \( y(t) \) denote, respectively, real \( m_1 \)-, \( m_2 \)-, and \( m_3 \)-vectors representing the state, the input and the output at time \( t \); and \( A \), \( B \), \( C \) are, respectively \( m_1 \times m_1 \), \( m_1 \times m_2 \), and \( m_3 \times m_1 \) matrices with real constant coefficients.
The input signals are a set of bounded continuous functions and are assumed to be known only at sampling instants $t_n = nT$, $n = 0, 1, 2, \ldots, N$. Without loss of generality, we assume that the signals are zero for $t < 0$.

Normally, the output is simulated for the time interval from 0 to $NT$. If the signal is periodic with period $NT$ or the signal vanishes for time greater than $NT$, then one may consider that the output is to be simulated over a time interval from zero to infinity.

Here a new computational scheme for evaluating the dynamic response of a continuous system by applying spline approximation in time domain is described. In section 5.2, we review some previous results on time domain simulation. In section 5.3, a new off-line simulation scheme based on the spline approximation is proposed. Some interesting properties of the proposed scheme are discussed in section 5.4. An illustrative example and conclusions are given in sections 5.5 and 5.6, respectively.

5.2. Review of Previous Results

If $\Phi(t)$ denotes the state transition matrix for the differential system defined in (5.1) without the forcing term, then the value of the output at time $t$ may be expressed as
\[ y(t) = C \{ \Phi(t) w(0) + \int_0^t \Phi(u - t) B x(u) \, du \} \] (5.2)

There are many computation algorithms to calculate \( y(t) \) by equation (5.2). For instance, M. L. Liou \textsuperscript{71} developed a computation scheme if \( x(t) \) is a set of step or delta functions. A recursive formula is derived to calculate the convolution integral. B. Capehart \textsuperscript{13} proposed another algorithm to compute (5.2) if the dynamic system which generates the signal \( x(t) \) is known. By converting the original forced system to a new higher order unforced system, he reduced the simulation problem to calculating the new transition matrix. The above results are not applicable here since the only knowledge which we have about the signals is their sampled values. Therefore, certain approximation methods were developed to evaluate the system output.

Since \( \Phi \) is known in closed form, the output can be duplicated by a digital processor using the typical scheme shown in Fig. 3.1, which was first introduced by C. Greaves and J. Cadzow\textsuperscript{[19]}. Suppose the value of the output of the simulator is denoted by \( \tilde{y}(t) \) and let \( \tilde{y}_k = \tilde{y}(kT) \). C. Greaves and J. Cadzow designed a digital processor which minimizes the mean square error \( E\{ |y_n - \tilde{y}_n|^2 \} \). The resulting digital processor required a rational approximation to the spectrum of the signal \( \Phi_{XX}(s) \). If \( \Phi_{XX}(j\omega) \) is band-limited, their results led to the conventional impulse invariant and
bilinear transformation designs which are discussed by
C. Kao

An alternate approach which was first introduced by
R. Vich is to use certain interpolation schemes to approxi-
mate the convolution integrand in equation 5.2.

The Gregory formula is used by R. Vich to obtain
approximations up to any desired order to the convolution
integrand corresponding to the physical system. He derived
D(z) corresponding to a given approximation by taking the
z-transformation of the summation approximating the convo-
olution integral. His zero order approximation (rectangular
approximation) leads to the same result as the impulse
invariant method.

Harrison and Leon[21] used a similar approach to deduce
difference equations that describe the digital filter in
the time domain. These equations depend upon a free
parameter the value of which is to be selected empirically.

An improved time domain simulation method based on the
approximation of the convolution integrand associated with
the physical system by a simple cubic spline is developed
by de Figueiredo and Kao. After integrating and taking
z-transforms of the resulting sums, a formulation which gives
the transfer function D(z) of the desired digital simulator
was developed. Due to the fact that the time domain simula-
tion involves a numerical quadrature, it can be shown that
the approximation of the convolution integrand by an inter-
polating spline leads to a result which is optimal in Sard's sense according to Schoenberg's Theorem. This scheme also possesses the other variational and convergence properties pertaining to splines. The most remarkable gain is that the simulation error converges to zero at the order of $T^5$, where $T$ is the sample period. A similar analysis disclosed that the simulation error by impulse invariant method is of the order of $T^2$. The resulting simulator, which is the combination of a recursive digital filter and a nonrecursive digital filter after $z$-transformation, can easily by realized on a digital computer.

R. de Figueiredo and A. Netravali also developed a digital simulation scheme by applying a spline approximation technique to reconstruct the input signal at each sampling instant. The simulator will operate on the samples of the input and minimize the maximum error between its output and the corresponding output of the physical system with respect to all the possible inputs of a prescribed class. Optimal properties of the splines are used as a problem of approximation of linear functionals. The resulting output is a linear combination of the sampled values of the signal with time (or index) varying weighting sequence and therefore the $z$-transformation technique was not used. The weighting sequence depends upon the dynamic system which generates the signal. A more general case of this approach is in progress.

Here an off-line simulation technique using only one
interpolation spline for each signal is developed. This is the main difference from the scheme just discussed who used N interpolation splines for a signal.

5.3. Description of the Method

For the reason of simplicity, let us consider the single input and output continuous system first. The multiple input and output system can be simulated in the same fashion. The transition metrix $\Phi(t)$ now becomes a scalar function $h(t)$ which is the impulse response of the system and has the following form:

$$h(t) = \sum_{m=1}^{M} a_m e^{b_m t} \quad (5.3)$$

where $a_m$ and $b_m$ are constants and $b_m$ has negative real part for all $m$.

Since the signal is given in a tabular form, an interpolation scheme is required to reconstruct either the convolution integrand or the signal so that the convolution integral described in (5.2) can be carried out. Here we use a generalized natural spline to reconstruct the signal. The reason to select spline function as the interpolation function is mainly due to its high convergence rate properties and the optimal interpolation property described in Chapter II. The computational advantage and the fact that spline approximation is relatively insensitive to round-off errors
serve as additional reasons. Computation simplification might not be so important from the mathematical point-of-view however, it is crucial in practical applications. The interpolating spline is obtained as follows:

Let $\mathcal{A}$ be the differential operator defined by (2.9) and $\mathcal{A}^*$ is its formal adjoint. Let $\{\phi_i(t)\}_{i=1}^{2q}$ be a set of linearly independent functions (which we will call representing functions) such that:

$$\mathcal{A}^*\mathcal{A} \cdot \phi_i(t) = 0 \quad 0 < t < T \quad \text{for} \quad i = 1, 2, \ldots, 2q.$$  \hfill (5.4)

Let $t_n$ denote $nT$ and $S(t)$ be defined as

$$S(t) = \sum_{i=1}^{2q} r_{in} \phi_i(t - t_n) \quad t_n \leq t \leq t_{n+1} \quad \text{for} \quad n = 0, 1, 2, \ldots, N-1.$$  \hfill (5.5)

where $r_{in}$ are real constants obtained by substituting (5.5) into the following equations and solving the resulting linear system,

$$S^{(\mu)}(t_n) = x^{(\mu)}(t_n) \quad n = 0, 1, 2, \ldots, N-1$$
$$\nu = 0, 1, 2, \ldots, 2q-2$$ \hfill (5.6)

$$\mathcal{A} S(t) = 0 \quad \text{for} \quad 0 \leq t \leq T$$
$$\text{or} \quad (N - 1)T \leq t \leq NT$$ \hfill (5.7)

In other words, $S$ is a generalized $\mathcal{A}$-spline
interpolating the data points \( x(t_n) \). \[35\]

When the spline function \( S(t) \) is obtained, the output of the continuous system is approximated by

\[
\bar{y}(t) = c_0 h(t) + \int_0^t h(t-u) S(u) \, du \tag{5.8}
\]

where \( c_0 \) is the initial state of the system.

Although the simulated output is given for the whole time range \([0, NT]\), usually only those output values at the same sampling instants the input i.e. \( \{y(nT)\}_{n=1}^N \) are needed. A computation method based on the concept of digital filtering and z-transformation technique is developed to simplify the calculation of the convolution integral when we substitute by \( nT \) in equation 5.8., i.e.

\[
\bar{y}(nT) = c_0 h(nT) + \int_0^{nT} h(nT-u) S(u) \, du \tag{5.9}
\]

Since using numerical integration to compute \( \bar{y}(nT) \) by equation 5.9 not only increases the computing time and simulation cost, additional conversion error will also be induced. The following scheme will compute the exact value of \( \bar{y}(t) \) at \( t = nT, n = 0, 1, \ldots, N \), without using numerical integration. The method is described below.

Substituting the impulse response \( h(t) \) defined in (5.3) and \( S(t) \) into (5.9) we get
\[
\tilde{y}(nT) = c_0 \ h(nT) + \sum_{k=1}^{n} \sum_{i=1}^{2q} r_{ik} \cdot \int_{(k-1)T}^{kT} \left\{ \phi_i [u - (k - 1)T] \cdot b_m (nT - u) \right\} du \\
= c_0 \sum_{m=1}^{M} a_m e^{b_m nT} + \sum_{k=1}^{n} \sum_{i=1}^{2q} r_{ik} \\
\cdot \int_{(k-1)T}^{kT} \phi_i [u - (k - 1)T] \cdot e^{b_m (nT - u)} du \\
= c_0 \sum_{m=1}^{M} a_m e^{b_m nT} + \sum_{m=1}^{M} \int_{(k-1)T}^{kT} \phi_i [u - (k - 1)T] \cdot e^{b_m (nT - u)} du \\
(5.10)
\]

Define

\[
\tilde{w}_m(nT) = c_0 \ a_m e^{b_m nT} + a_m \sum_{k=1}^{n} \sum_{i=1}^{2q} r_{ik} \int_{(k-1)T}^{kT} \left\{ \phi_i [u - (k - 1)T] \cdot e^{b_m [(n - 1)T - u]} \right\} du \\
= a_m e^{b_m T} \left\{ c_0 e^{b_m (n-1)T} + \sum_{k=1}^{n-1} \sum_{i=1}^{2q} r_{ik} \right. \\
\cdot \int_{(k-1)T}^{kT} \phi_i [u - (k - 1)T] \cdot e^{b_m [(n - 1)T - u]} du \right\} \\
+ a_m \sum_{i=1}^{2q} r_{in} \int_{(n-1)T}^{nT} \phi_i [u - (n - 1)T] \cdot e^{b_m (nT - u)} du \\
= e^{b_m T} \left\{ \tilde{w}_m [(n - 1)T] + a_m \sum_{i=1}^{2q} r_{in} \int_{0}^{T} \phi_i (u) \cdot e^{-b_m u} du \right\} \\
(5.11)
\]
Let

\[ v_{im} = a_m \cdot \int_0^T \Phi_i(u) e^{-b_m u} \, du \quad (5.12) \]

Taking z-transform of (5.11), we get

\[ \tilde{w}_m(z) = e^{b_m T} z^{-1} \tilde{w}_m(z) + \sum_{i=1}^{2q} R_i(z) \cdot v_{im} \quad (5.13) \]

where \( R_i(z) = \sum_{n=0}^{\infty} r_{in} z^{-n} \)

Hence,

\[ \tilde{w}_m(z) = \frac{1}{1 - e^{b_m T} z^{-1}} \cdot \sum_{i=1}^{2q} R_i(z) \cdot v_{im} \quad (5.14) \]

To simplify the notation, let us define

\[ \bar{a}_m = e^{b_m T} \quad (5.15) \]

then

\[ \tilde{Y}(z) = \sum_{m=1}^{M} \tilde{w}_m(z) = \sum_{m=1}^{M} \frac{1}{1 - \bar{a}_m z^{-1}} \cdot \sum_{i=1}^{2q} R_i(z) v_{im} \quad (5.16) \]

In matrix form, we have
\[ Y(z) = \left[ \frac{1}{1 - a_1 z^{-1}}, \frac{1}{1 - a_2 z^{-1}}, \ldots, \frac{1}{1 - a_M z^{-1}} \right] \]

\[
\begin{bmatrix}
v_{1,1} & v_{1,2} & \cdots & v_{1,2q} \\
v_{2,1} & v_{2,2} & \cdots & v_{2,2q} \\
\vdots & \vdots & \ddots & \vdots \\
v_{M,1} & v_{M,2} & \cdots & v_{M,2q}
\end{bmatrix}
\begin{bmatrix}
R_1(z) \\
R_2(z) \\
\vdots \\
R_{2q}(z)
\end{bmatrix}
= A(z) \cdot V \cdot R(z)
\] (5.17)

where

\[ A(z) = \left[ \frac{a_1}{1 - b_1T z^{-1}}, \frac{a_2}{1 - b_2T z^{-1}}, \ldots, \frac{a_M}{1 - b_MT z^{-1}} \right] \]

(5.18)

\[
\begin{bmatrix}
v_{1,1} & v_{1,2} & \cdots & v_{1,2q} \\
v_{1,2} & v_{2,2} & \cdots & v_{2,2q} \\
\vdots & \vdots & \ddots & \vdots \\
v_{M,1} & v_{M,2} & \cdots & v_{M,2q}
\end{bmatrix}
\] (5.19)

\[ R(z) = \text{col} \left[ R_1(z), R_2(z), \ldots, R_{2q}(z) \right] \]

(5.20)

with \( R_i(z) = Z \{ r_{in} \} \) with respect to \( n \), where \( r_{in} \) is defined
by (5.5) for $0 \leq n \leq N$, and $r_{in} = 0$, for $n < 0$ and $n > N$, $r_{in}$ is defined by

\[ r_{in} = 0, \quad \text{for the non-periodic case} \]

\[ r_{i}(kN + n) = r_{in}, \quad k = \pm 1, \pm 2, \ldots, \quad \text{for the periodic case} \quad (5.21) \]

The simulator (5.17) can be realized on computer by using the structure shown in Fig. 5.1.

5.4. Convergence and Error Analysis

The approximated output $\tilde{y}_n$ obtained by the computation scheme proposed above will be exact if $x(t) = S(t)$, a.e.. In other words, the only error source of the above scheme is the modeling or approximation error between the actual signal and the interpolating spline. To investigate the convergence rate of the results for the discrete system approaching the output of the actual system, we make the following analysis which gives the error bound of the output.

Let $e_n$ denote the error between the simulated output and the actual output at time $t = nT$, i.e.

\[ e_n = |y(nT) - \tilde{y}_n| \quad (5.22) \]

where $\tilde{y}_n$ is defined by (5.17). Then, we have

**Theorem 5.1.** If $x(*) \in \mathcal{K}^{2q}[0,NT]$, $q \geq 1$ and $\mathcal{A}$ is the
q-th order differential operator of the interpolating spline function, then the error defined in (5.2) is bounded by

$$e_k \leq K_0 \cdot k \cdot T^{2q+\frac{1}{2}} \cdot \|x^* x(t)\|_L^2[0,NT]. \quad (5.23)$$

where $k_0$ is a constant independent of $x(t)$ and $\{x(kT)\}_{k=0}^N$.

**Proof:** If $x(*) \in K^{2q}[0,NT]$, by J. Jerome and R. Varga's theorem [40], that

$$\|D^\infty(x(t) - S(t))\|_{L^\infty(0,NT)} \leq \lambda_\alpha T^{2q - \alpha - \frac{1}{2}} \cdot \|x^* x(t)\|_{L^2[0,NT]} \quad (5.24)$$

where $\lambda_\alpha$ is a constant independent of $x(t)$ and $\{x(kT)\}_{k=0}^N$, hence we have

$$\sup_{t \in [0,NT]} |x(t) - S(t)| \leq \lambda_\alpha T^{2q - \frac{1}{2}} \cdot \|x^* x(t)\|_{L^2[0,NT]} \quad (5.25)$$

Define

$$e_{mk} = |w_m(kT) - \bar{w}_m(kT)|, \quad (5.26)$$

where $w_m(kT) = c_0 a_0 e^{b_m kT} + a_m \int_0^{kT} x(u - kT) e^{b_m u} du$ and $w_m(kT)$ is defined by (5.11). We have
\[ e_{mk} = \left\{ a_m \int_0^T e^{b_m t} x(kT - t) \, dt + e^{b_m T} w_m \left[ (k-1)T \right] \right\} \]

\[ - \left\{ a_m \int_0^T e^{b_m t} S(kT - t) \, dt + e^{b_m T} \bar{w}_m \left[ (k-1)T \right] \right\} \]

\[ \leq \left| a_m \int_0^T e^{b_m t} \left[ x(kT - t) - S(kT - t) \right] \, dt \right| \]

\[ + \left| e^{b_m T} \right| \left| w_m ((k-1)T) - \bar{w}_m ((k-1)T) \right| \]

\[ \leq \left| a_m \int_0^T \left| e^{b_m t} \right| \left| x(kT - t) - S(kT - t) \right| \, dt \right| \]

\[ + \left| e^{b_m T} \right| \cdot e_m (k-1) \]  \hspace{1cm} (5.27)

Substituting (5.25) into (5.27), we have

\[ e_{mk} < \left| a_m \right| \int_0^T \bar{K}_0 \left[ e^{T^q} - \frac{1}{2} \right] \, dt + e_m (k-1) \]

\[ = \bar{K}_0 \left| a_m \right| T^{2q} + \frac{1}{2} + e_m (k-1) \]  \hspace{1cm} (5.28)

Since

\[ e_{ml} = a_m \int_0^T e^{b_m t} \left| x(T - t) - S(t - t) \right| \, dt \]

\[ \leq \bar{K}_0 \left| a_m \right| T^{2q} + \frac{1}{2} \]  \hspace{1cm} (5.29)
by induction, we know that

\[ e_{mk} \leq K_0 |a_m| |e^{b_m T}_k|^{2q + \frac{1}{2}} \]  \hspace{1cm} (5.30)

\[ e_k \leq \sum_{m=1}^{M} e_{mk} \]

\[ = K_0 \sum_{m=1}^{M} |a_m| |e^{b_m T}_k|^{2q + \frac{1}{2}} \]

\[ = K_0 k T^{2q + \frac{1}{2}} \]  \hspace{1cm} (5.31)

Q.E.D.

**Theorem 5.2.** If \( x(\cdot) \in \mathcal{K}^q \{0, NT\} \) and is a \( q \)-th order differential operator, then, the discrete error

\[ e_k \leq K_0^* k \cdot \| A^* A x(t) \|_{L^2[0, NT]} \cdot T^{q + \frac{1}{2}} \]  \hspace{1cm} (5.32)

where \( K_0^* \) is a constant independent of \( x(t) \) and \( \{x(kT)\}^N_{k=0} \).

**Proof:** By J. Jerome's Theorem, if \( x \in \mathcal{K}^q [0, NT] \),

\[ \| D^\alpha (x(t) - S(t)) \|_{L^\infty [0, NT]} \]

\[ \leq \lambda_\alpha^T (q - \alpha - Y_2) \cdot \| A^* A x(t) \|_{L^2[0, NT]} \]  \hspace{1cm} (5.33)

where \( \lambda_\alpha \) is a constant independent of \( x(t) \) and \( \{x(nT)\}^N_{n=0} \).
The same procedure is used to prove theorem 5.2.

\[ \text{Q.E.D.} \]

From theorem 5.1 we see that the error bound is of the order of \( T^{2q+\frac{3}{2}} \) if \( x(\cdot) \in K^{2q} 0,NT \) and it increases linearly with \( t \). This is a favorite factor for transient analysis applications since usually we are interested in the initial overshoot of the output of a physical system. The small error at the initial period of simulation is desirable.

By the results of R. de Figueiredo and A. Netravali[36], \( \sup |e(NT)| \) is minimized as a linear functional based on the knowledge of the sampled value of the signal when the signal \( x \) belongs to the class such that \( \|A \cdot x(t)\|_{L^2[0,NT]} \leq \beta < \infty \). The differential operator \( A \) for the interpolating spline can be chosen from this criterion.

If the cubic spline is used to interpolate the signal and \( x(t) \in C^4[0,NT] \), then \( |S(t) - x(t)| = o(T^4) \)[39] and the same analysis will show that \( e_n = O(T^5) \), i.e., the same order as when a cubic spline is used to approximate the convolution intergrand. The numerical examples in the next section illustrate the method developed so far.

For the case in which \( x(t) = 0 \), \( t > NT \), the period of simulation can be extended to infinity by letting \( r_{in} = 0 \) for \( n > N \). The associated simulation error \( e_n \) for \( t > NT \) is
bounded and approaches zero as time increases, we have

\[ e_n = e_n h((n-N)T), \quad n > N \]  \hspace{1cm} (5.34)

Since

\[ \lim_{n \to \infty} h(t) = 0 \]  \hspace{1cm} (5.35)

it follows that

\[ \lim_{n \to \infty} e_n = e_N \lim_{n \to \infty} h((n-N)T) \]

\[ = 0 \]  \hspace{1cm} (5.36)

5.5 An Illustrative Example

An example is given here to illustrate the method. Suppose that the transfer function of the continuous system is \(1/(1 + 1.666s)\) and the input signal is

\[
\begin{align*}
0 \leq t & \quad \text{if } 0.5t \\
0 & \quad \text{if } t < 0
\end{align*}
\]  \hspace{1cm} (5.37)

The input is sampled at \(T = 0.5\) and a natural cubic spline is used to interpolate it. The total period for simulation is \([0,10]\).

The differential operator \(\mathcal{A}\) for cubic spline is \(D^2\). If we chose the representing functions \(\{\phi_i(t)\}_{i=1}^4\) to be:

\[ \phi_i(t) = \frac{t(t - T)^2}{T^2} \]  \hspace{1cm} (5.38)
\[ \phi_2(t) = \frac{t^2(t - T)}{T^2} \quad (5.39) \]

\[ \phi_3(t) = \frac{(t - T)^2(2t + T)}{T^3} \quad (5.40) \]

\[ \phi_4(t) = \frac{t(3T - t)}{T^3} \quad (5.41) \]

Then, for \( t \in [kT, (k+1)T] \), \( k = 0, 1, 2, \ldots, N-1 \), we have

\[ S(t) = m_k \cdot \phi_1(t - t_k) + m_{k+1} \cdot \phi_2(t - t_k) \]

\[ + x_k \cdot \phi_3(t - t_k) + x_{k+1} \cdot \phi_4(t - t_k) \quad (5.42) \]

where \( s_k = x(kT) \) and \( m_k \) is the derivative of the interpolating spline at \( t = kT \). The value of \( m_k \) can be obtained by using the algorithm proposed by J. Ahlberg, et al [39] or the program given in Appendix B. The resulting discrete simulator is shown in Fig. 5.2. The output shown in Fig. 5.3, is compared with the one obtained from the simulator designed by impulse invariant method (normalized at \( \omega = 0 \)) which has an excellent magnitude response with respect to the actual system. The phase error is the main reason for the large error in the simulation period for the impulse invariant simulator. The error of spline simulator is so small that it is difficult to tell from the diagram.

Since the coefficients of the representing functions \( \{r_{in}\} \) are obtained by a spline interpolating program shown in Fig. 5.1, each \( r_{in} \) is a function of \( \{x_n\}_{n=1}^N \) and \( i \). Hence when implementing a system that sends the actual input samples \( \{x_n\}_{n=1}^N \) to \( \{y_n\}_{n=1}^N \), the time-invariant recursive filter which sends \( \{r_{in}\}_{i=1, \ldots, 2q, n=1, \ldots, N} \) to \( \{y_n\}_{n=1, \ldots, N} \) discussed
in this chapter would be preceded by an interpolating program which maps all of \( \{x_n\}_{n=1}^N \) into each \( r_{in} \).

5.6. Generalization and Conclusion

Although the above analysis was carried out for scalar inputs, it can be easily generalized to vector inputs at the cost of notational complexity. A set of \( \mathcal{A} \)-spline functions \( S(t) = \text{Col}(S_1(t), S_2(t), \ldots, S_{m_2}(t)) \) may be used to interpolate the input \( x(t) = \text{Col}(x_1(t), x_2(t), \ldots, x_{m_2}(t)) \). By using equation (5.2), a set of recursive formula can be derived as in the scalar case and the same error analysis can be applied.

Due to the optimal properties of the spline approximation and its computational ease, the spline function is shown to be a valuable tool for system simulation from the time domain approach. Usually, the spline function is employed to approximate either the sampled input or the sampled convolution integrand.

A computational algorithm is developed here to simulate a continuous system by applying the spline function to interpolate the sampled input. If the input signal is in \( K^\nu[0, NT], \nu \geq 1 \), the maximum error is bounded over the period of simulation study.

Since different time domain simulation schemes using the spline approximation possess various optimal and convergence properties pertaining to spline, the choice of the simulation scheme depends upon the applications. As mentioned earlier, the scheme developed here is very suitable for off-line system transient analysis or prediction due to the property that its maximum possible error increases with time.
Fig. 5.1.a Proposed Computation Scheme

Fig. 5.1.b Structure of $D_1(z)$ in Fig. 5.1.a
Fig. 5.2. (a) Configuration of Spline Simulator

(b) Configuration of Impulse Invariant Simulator

\[ V_{11} = 0.01743 \quad V_{12} = -0.01851 \quad V_{13} = 0.20303 \]

\[ V_{14} = 0.22842 \quad \bar{a}_1 = \exp(-0.3) \quad A = 1/(1 - \exp(-0.3)) \]
Fig. 5.3 Output of the Digital Simulators
Fig. 5.4 | Error in the Time Domain| vs. Time
CHAPTER VI

CONCLUSIONS AND PROPOSALS FOR FURTHER RESEARCH

6.1. **Summary and Conclusion**

Problems of simulating the dynamic response of a linear, time-invariant system based on the spline approximation have been explored. As previously mentioned, the simulation scheme varies with different applications and given conditions. In general, the solutions to the simulation problems can be divided into two groups — those based on frequency domain methods and those based on time domain methods. In the present work, spline functions have been applied in both there approaches.

In Chapter III, two new simulation methods based on the spline approximation in frequency domain are presented. When the input signal is bandlimited with the sampling frequency greater than the Nyquist rate, the transfer function or frequency response of the discrete simulatory which duplicates the exact value of the output of the actual system at sampling instants is not unique. These transfer functions can be represented by means of a infinite duration non-recursive digital filter whose impulse response amounts of its Fourier coefficients. However, a finite duration non-recursive digital filter must be used for physical realization. Therefore, a truncation error is induced. It is found that the smoother the frequency response is, the
faster the error in the spectrum converges to zero. Thus, the maximum conversion error due to truncation will be minimized if we select the transfer function with the smoothest frequency response. Due to the smoothness property of spline functions, the polynomial spline is the optimal interpolating function not only in the sense that it is the best approximation function\[^{39,42}\], but such that the worst possible error of the resulting simulator due to truncation is reduced. The error bound for both modeling and conversion error are derived separately. The numerical examples for Chapter IV are given in Chapter IV.

In Chapter V, simulation schemes based on the spline approximation in time domain are discussed. The impulse response of the actual system and the values of the inputs at the sampling instants are assumed given. The spline function usually is used to interpolate the convolution integrand or reconstruct the input signal. Here a new method to reconstruct the input signal is proposed so that its pseudo norm of the error function is minimized. Error bounds for this simulation scheme are also derived.

Summing up, two simulation techniques and one computational scheme have been developed in this thesis and they represent new solutions to a class of current simulation problems.

6.2. **Suggestions for Further Studies**
System simulation by means of computer becomes more and more difficult, especially for modern day complicated, nonlinear, time-variant or distributed parameter systems. To simulate such systems by computer requires different kinds of knowledge in digital and analog circuit theory, control systems, sampled-data control, computer science, numerical analysis, approximation theory, statistics, probability theory, and the specific field in which computer simulation is applied.

Actually, all numerical methods solving the differential equations of a system can be viewed as a wide-sense system simulation.\cite{43,60} New techniques are required to apply spline function to simulate those nonlinear, time-variant or distributed parameter systems. The results in Chapter III could be extended to a linear, distributed parameter system whose input is bandlimited\cite{57} How to select the mesh points of an interpolating spline optimally to approximate a step function so that an ideal low-pass or bandpass filters can also be designed by the methods proposed in Chapter III would be another interesting topic.\cite{51}

The application of spline functions to nonlinear system\cite{58} also ought to be extended to nonlinear system simulation problems.

It seems to us that much can be done in this area. We hope that more significant results will be developed to make the digital simulation an even more powerful and efficient problem-solving technique in the near future than is now.
APPENDIX A

THE FREQUENCY DOMAIN SIMULATION METHOD BASED ON THE FINITE DURATION IMPULSE RESPONSE DIGITAL FILTER DESIGN TECHNIQUE

It is possible to interpolate $N$ frequency samples of a transfer function $H$ by means of a non-recursive digital filter with $2N - 1$ pulses. The method is described as follows:

Suppose we are given a mesh in frequency domain

$$\Delta = \{0 = \omega_1 < \omega_2 < \omega_3 \ldots < \omega_N\}$$

and the frequency samples

$$\left\{H(j\omega_i) = H_R(\omega_i) + jH_X(\omega_i)\right\}_{i=1}^{N},$$

where $H_R(\omega_i)$ and $H_X(\omega_i)$ are the real part and imaginary part of the value of the transfer function at frequency $\omega_i$, we have the following theorem:

**Theorem A-1** If $T > 2\pi/\omega_N$ and $H_X(\omega_1) = 0$, there exists a nonrecursive digital filter $D(z)$:

$$D(z) = \sum_{k=-(N-1)}^{N-1} c_k z^{-k} \quad N \geq 1 \quad \text{(A.1)}$$

such that

$$D(e^{i\omega_i T}) = H(j\omega_i), \quad i = 1, 2, \ldots, N$$

**Proof:** Since
\[ D(e^{j\omega_i T}) = \sum_{k=1-N}^{N-1} c_k e^{-j\omega_i kT} \]

\[ = c_0 + \sum_{k=1}^{N-1} (c_{-k} + c_k) \cos k \omega_i T \]

\[ + j \sum_{k=1}^{N-1} (c_{-k} - c_k) \sin k \omega_i T \]

\[ = H_R(\omega_i) + j H_X(\omega_i) \quad i = 1, 2, \ldots, N \quad (A.2) \]

Therefore, two systems of linear equations can be derived.

\[ H_R(\omega_i) = \sum_{k=0}^{N-1} a_k \cos k \omega_i T, \quad i = 1, 2, \ldots, N \quad (A.3) \]

\[ H_X(\omega_i) = \sum_{k=1}^{N-1} b_k \sin k \omega_i T, \quad i = 2, 3, \ldots, N \quad (A.4) \]

Since \( \{\cos k \omega T\}_{k=0}^{N-1} \) is a linearly independent set of \( N \) functions for \( \omega \in \left[ 0, \frac{\pi}{T} \right) \) and \( \{\sin k \omega T\}_{k=0}^{N-1} \) is a linearly independent set of \( (N-1) \) functions for \( (0, \frac{\pi}{T}) \) and these two systems of linear functions (A.3) and (A.4), are nonsingular, there is one and only one solution for the constants \( \{a_k\}_{k=0}^{N-1} \) and \( \{b_k\}_{k=1}^{N-1} \). By comparing (A.3), (A.4) with (A.2), we obtain a unique set \( \{c_k\}_{k=1-N}^{N-1} \) defined by

\[ c_0 = a_0 \quad (A.5) \]
\[ c_k = \frac{a_k + b_k}{2} \quad k = 1, 2, \ldots, N-1 \quad (A.6) \]

\[ c_{-k} = \frac{a_k - b_k}{2} \quad k = 1, 2, \ldots, N-1 \quad (A.7) \]

Q.E.D.

This is a generalized approach of the digital filter design using frequency sampling technique. If \( \Delta \) is an equispaced mesh, this method becomes the ordinary discrete Fourier transform design\(^2\) or the so-called "Type I Frequency Sampling Design" according to L. R. Rabiner and R. W. Schaffer\(^{50}\). It can be realized either by a nonrecursive structure or by a recursive one. (See [50], pp. 204)

The advantage of this approach is that the frequency response of the digital filter has the exact values at the sampling instants. However, between frequency samples, the continuous frequency response may differ significantly from the desired frequency response.
C-----IDEAL LOW-PASS FILTER DESIGNED BY SCHEME I
C-----PASSBAND IS 0 = 0.2, STOPBAND IS 0.3 = 0.5

START

DIMENSION A(60)
DATA Y1,Y2,Y3,Y4,PI/1,0,0,0,0,0,0,0,0,0,0,1/
DPI=PI/10.
X1=4,0*DPI
X2=6,0*DPI
T=2,0,3,4,1593
X2=X2*X1
A1=X2*X2
B1=X2*X2

C-----HERMITE POLYNOMIAL
A2=3,0*X2*X2
B2=2,0*X2
C1=Y2*Y3*X2*Y1
C2=Y4*Y3
C=Y3
D0=Y1
C1=AA
C2=-3,0*AA*X1+BB
C3=-3,0*AA*X1*X1+BB*2,0*X1(CC
C4=-AA*X1*X1+BB*X1*X1+CC*X1*DD
WRITE(6,1005)C1,C2,C3,C4

1005 FORMAT(* CUBIC COEFF. ARE*4E16+7)
DW=PI/1000.
AN=FLOAT(N=1)*T
A(1)=0,2
W=400,0*DW
DO 10 I=401,600
A(1)=A(1)+(C1+W+W+C2+W+W+C3+W+C4)*DW
10 W=W+DW
A(1)=(A(1)-0,5*DW)/PI
H=1,2*C1+2,C2
D=6,C1
DO 14 N=2,50
AN=FLOAT(N=1)*T
A(N)=(SIN(0,2*AN)+SIN(0,3*AN))*AN*H +(COS(0,2*AN)-COS(0,3*AN))*D
14 A(N)=A(N)/(AN*4*PI)
DO 12 I=1,50
12 WRITE(6,1011)I,A(I)

1011 FORMAT(I5,E16,8)
I2=TIME(2)
I3=TIME(3)
WRITE(6,1111)I2,I3

1111 FORMAT(218)
CALL EXIT

END
APPENDIX C

A SAMPLE PROGRAM FOR SCHEME II

START

DIMENSION V(25), P(25), Q(25), U(25), X(50), S(25), T(25), A(25)

DIMENSION SD(25), C(60)

FA(W) = 1./(1.+4.*W*W)

SA = 0.5

PI = 3.1415926

SC = 0.5

SB = 2.0

NM = 20

NMF = NM/2 + 1

NA = NM + 1

TT = 1.

TW = 2.*0.3.1415926/TT

SW = TW/NM

I = 1

DO 10 I = 1, NMF

II = NM - I + 2

AI = FLOAT (I - 1)

A(I) = FA(AI*SW)

A(I11) = A(I)

10 CONTINUE

J = NM - 1

DO 15 J = 1, J

12 I = I + 2

15 SD(1) = 3.*(A(I) - 2.*A(I1) + A(I2))/(-SW*SW)

SD(NA) = 3.*(A(NM) - 2.*A(NA) + A(2))/(-SW*SW)

SD(1) = 3.*(A(NM) - 2.*A(1) + A(2))/(-SW*SW)

V(NM + 1) = 0.

P(1) = SA

U(1) = 0.

S(1) = 1.

Q(1) = 0.

T(NM + 1) = 1.

DO 20 K = 2, NA

P(K) = SA*Q(K - 1) + SB

Q(K) = SC/P(K)

U(K) = (SD(K) - SA*U(K - 1))/P(K)

20 CONTINUE

S(K) = SA*S(K - 1)/P(K)

DO 30 T = 1, NM

K = NA + T

T(K) = Q(K)*T(K + 1) + S(K)

30 CONTINUE

V(K) = Q(K)*V(K + 1) + U(K)

X(NA) = (SD(NA) - SC*V(2) - SA*V(NM))/(-SC*T(2) + SA*T(NM) + SB)

DO 40 K = 2, NM

40 X(K) = T(K)*X(NA) + V(K)

X(1) = X(NA)
WRITE(6,1200)
FORMAT("THE SECOND DERIVATIVE OF SPLINE FUNCTION ARE "
MS DS A")
DO 60 I=1,NA
WRITE(6,1600)I,X(I),SN(I),A(I)
1600 FORMAT(T4,3F15.8)
WRITE(6,1201)
1201 FORMAT("TIME DOMAIN PULSES ARE"
SS=SW/100.
AS=0.
DO 81 J=1,10
DO 80 K=1,100
DA=SSW*FLOAT(K-1)
DN=SW-DA
SS=(X(J)*DN**3+X(J+1)*DA**3)/(6.*SW)+(A(J)-X(J)*SW**2/6.)*DN+
1*(A(J+1)-X(J+1)*SW**2/6.*DA)/4
80 AS=AS+SS
81 CONTINUE
C(1)=AS*SSW/PI
DO 96 N=1,50
AN=FLOAT(N)*TT
K=N+1
CC=0.
DO 93 I=1,NM
AI=FLOAT(I)
93 CC=CC-(X(I+1)-X(I))*(2.*PI*FLOAT(N)**4)
96 C(K)=CC/(2.*PI*FLOAT(N)**4)
DO 97 T=1,50
97 WRITE(6,120)T,C(1)
120 FORMAT(T4,F16.7)
12 T=TIME(2)
T3=TIME(3)
WRITE(6,1234)T2,T3
1234 FORMAT(2I10)
500 CALL EXIT
END
REFERENCE


46. O. Dahl and K. Nygaard, "SIMULA, A Algol Based Simulation Language", *ACM Comm.*, Vol. 9, No. 9, September 1966

47. L. Wayne, *Simulation: The Road to Coexistence*, Datamation, April 1971


57. M. R. Weinberger, "Optimal Inversion and Digitization for Distributed Parameter Systems Whose Inputs Have Bandlimited Time-Dependence", AVCO Corporation, Wilmington, Massachusetts, 1970


