UNIVERSITY OF SOUTHAMPTON
FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
School of Engineering Science

Frequency Domain Iterative Tuning for Active Noise and Vibration Control

by

Jian Luo

Thesis for the degree of Doctor of Philosophy

February 2008

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
SCHOOL OF ENGINEERING SCIENCE

Doctor of Philosophy

FREQUENCY DOMAIN ITERATIVE TUNING FOR ACTIVE NOISE AND VIBRATION CONTROL

by Jian Luo

In this thesis a new adaptive control method, called Iterative Tuning in the Frequency Domain (FD-IT), is proposed for Active Noise and Vibration Control (ANVC). This approach is a gradient based self-tuning method which completely relies on analysis of the frequency response of system dynamics and the spectrum of signals.

The new method is based on a new gradient estimation theory in the frequency domain. In this theory the gradient of the output spectrum with respect to controller parameters is expressed with the frequency response of dynamics and the spectrum of signals. When the performance gradient with respect to controller parameters can be expressed as some function of the signals' spectrum, it can be computed out completely in the frequency domain. Similar to audio compression, when the system's signals contain few frequencies, the computation of performance gradient can be greatly simplified by making "partial modelling" with respect to those frequencies.

According to the proposed theory, the new iterative tuning method, i.e., FD-IT, is developed for ANVC problems with periodic disturbances. It can tune the feedback/feed-forward controllers simultaneously with one experiment per iteration except some extra experiments for initial tuning. It covers both Single Input Single Output (SISO) and Multiple Input Multiple Output (MIMO) systems. Furthermore, it can be extended to nonlinear systems as well. Some issues about the implementation of the iterative method are discussed. Through the comparison with some other popular active control methods in ANVC, the advantages of the new method, including: the flexibility in selecting controllers, the simplicity in control structure, and the convenience in implementation, are emphasized.

The effectiveness and robustness of the proposed iterative tuning method are tested through simulated SISO and MIMO Linear Time Invariant (LTI) systems. Two simulated nonlinearities are used to illustrate the usefulness of the methods in nonlinear system as well. To show the practicability, the linear and nonlinear FD-ITs are implemented in an air-duct system with a PC-DSP based agent-architecture. All the results illustrate that FD-IT is an easy and effective approach to solve ANVC problems with periodic disturbances.

（知人者智，自知者明。
胜人者有力，自胜者强。）

—《道德经》，老子

He who knows other men is discerning
he who knows himself is intelligent.
He who overcomes others is strong
he who overcomes himself is mighty.

— "Tao Te Ching", by Lao Zi
(translated by James Legge)

Contents

List of Tables

List of Figures

Declaration of Authorship

I, Jian LUO, declare that the thesis entitled 'Frequency Domain Iterative Tuning for Active Noise and Vibration Control' and the work presented in it are my own. I confirm that:

- this work was done wholly or mainly while in candidature for a research degree at this University;
- where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- where I have consulted the published work of others, this is always clearly attributed;
- where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- parts of this work have been published as:
  - Jian Luo and S. M. Veres, (2007). Frequency domain iterative feedforward/Feedback tuning for MIMO ANVC. ALCOSP'07, Saint Petersburg, Russia. (In Press)
  - Jian Luo and S. M. Veres, (2007). Iterative feedforward/Feedback tuning in the frequency domain for ANVC. In ECC'07, Kos, Greece, pp.381-388.
  - S. M. Veres and J. Luo (2005). Active structural acoustic control of machine enclosures. IFAC'05, Prague, Czech Republik, (CD-ROM1).
  - S. M. Veres and J. Luo (2004). A class of BDI agent architectures for autonomous control. CDC'04, Paradise Island, Bahamas, pp.4746-4751.
  - S. M. Veres and J. Luo (2004). Formal verification of autonomous control agents. IASTED'04, Hawaii, USA, pp.140-144.

Signed: _____

Date:_____

Acknowledgements

First and for most, I would like to give my deepest gratitude to my supervisor, Sandor M. Veres. Without his guidance, support and encouragement, this research work can not come into reality. His knowledge and experiences have always guided my research work in fruitful directions.

Over the past years I have also enjoyed working with all students and staffs in in Electro-mechanical System group and Computational Engineering and Design Center (CEDC): Dr. Cheng Hock Tan, Dr. John K. Atkinson and Dr Andrew J. Chipperfield have given me great help for my research work.

Furthermore, I want to give a very big 'Thank you' to my good friends in Southampton, who made my life a pleasure during the research work udertaken. I am so lucky to meet you in Southampton. Among them are Cheng Gu, Cheng Lu, Fan Yang, Haoji Hu, Jian Liu, Jiyu Chen, Miao Yang, Shijun Wen, Tao Xing, Wu Hao, Xiang Liu, Xiaoqian Shi, Yu Yang and Zhaokai Ma.

Finally, my personal deep thanks must be extended to my family: my wife, Xiaoli Chen, my parents and my sister, who always stood with me during my PhD period and share my great happiness with the conferment of this degree. Their consistent support is always crucial in my whole life.

Nomenclature

SYMBOL	
C	Controller
$\mathbb{C}$	Set of Complex number
$\boldsymbol{d}$	Disturbance signals
experiment	When appeared together with 'iteration', it denotes a single tuning stage when changing the condition of ANVC, i.e., updating controller parameters and injecting extra signals; In Chapter 8, it denotes a complete tuning work in the experimental platform
F	Feed forward controller (or path)
G	Unknown dynamics in active noise and vibration control problems
H	Feedback controller (or path)
iteration	One iterative stage in the tuning with repeated operation
j	Complex imaginary unit, i.e., $\sqrt{(}-1)$
J	Control performance, average linear quadratic performance as default
N	Common period as default
$\boldsymbol{r}$	Reference signal
$\mathbb{R}$	Set of Real number
T	Closed loop dynamics
$\boldsymbol{u}$	Control actions
$\boldsymbol{u}_f$	Feed forward control actions
$\boldsymbol{u}_h$	Feedback control actions
$\boldsymbol{w}$	controller parameter vector
$\boldsymbol{w}_H$	Feedback controller parameter vector
$\boldsymbol{w}_F$	Feed forward controller parameter vector
$\boldsymbol{y}$	System output
z	Time delay operator
$\|\|\cdot\|\|_n$	n-order Norm of a vector, matrix or function
$\mapsto$	Mapping to
$:=$	Estimate
$\approx$	approximate
$\doteq$	Denote
$\in$	Belong to

ϕ_x	Spectrum of signal $\boldsymbol{x}$
Φ_P	Frequency response of dynamics P
P'	Derivative of dynamics P
M^{T}	Transport of matrix M
M^*	Conjugation of matrix M
$\bigcup$	Union
∇J	Gradient of function J
$\Delta \boldsymbol{x}$	Increment of variable $\boldsymbol{x}$
∞	Infinity

ABBREVIATION

ARMA	AutoRegressive, Moving Average model
ARX	AutoRegressive, Moving Average eXogenous model
BDI	Belief-Desire-Intention agent model
DFT	Discrete Fourier Transform
FD	Frequency Domain
FD-IT	Iterative Tuning in the Frequency Domain
FIR	Finite Impulse Response (filter)
FIR-FD-IT	Finite Impulse Response filter-based Iterative Tuning in the Frequency Domain
FRF	Frequency Response Function
FSF	Frequency Selective Filtered
FSF-FD-IT	Frequency Selective Filtered Iterative Tuning in the Frequency Domain
FELMS	Filter error (e) Least Mean Square method
FXLMS	Filter reference (x) Least Mean Square method
GFRF	Generalized Frequency Response Function
IFT	Iterative Feedback Tuning
IIR	Infinite Impulse Response (Filter)
ILC	Iterative Learning Control
LMS	Linear Mean Square method
LQ	Linear Quadratic
LS	Least Square method
LTI	Linear Time-Invariant system
LTV	Linear Time-Variant system
MIMO	Multi-Input Multi-Output system
MF-FDT	Model-Free Frequency Domain Tuning
NARMAX	Nonlinear Auto-Regressive, Moving Average eXogenous model
RLS	Recursive Least Square
SISO	Single-Input and Single-Output system
TD	Time Domain
TD-IFT	Iterative Feedback Tuning in the Time Domain
ZPK	Zeros-Poles-gain format filter

Chapter 1

INTRODUCTION

Vibration is one of the essential and common phenomenons in the physical world. In acoustics vibration is called sound or noise in the frequency range of human being's hearing. There are also structural vibrations out of the range of human hearing. Some sounds and vibrations are very useful. For instance, the voice is one of the basic methods to communicate information, and testing shock waves can be used to detect fault in equipments and structures. Some high levels of noise ($>$90dB) can have adverse effects and are considered as serious environmental pollution. Some vibration can cause damage to equipments and structures. With the development of industry and life in modern society, the problem to deal with unwanted noise and vibration has become more and more imperative in the last 30 years.

Traditionally, noise and vibration were reduced by passive methods, such as: spring damping system, damping plates, mufflers, sound absorbing materials, double-glazing windows, noise barriers etc. All these passive methods are mostly suitable for high frequencies and have various disadvantages. In the last two decades, active control of noise and vibration has emerged as an applicable approach to bridge the low-frequency technology gap.

1.1 Active Noise and Vibration control

Active noise and vibration control (ANVC) aims to attenuate the system output, that has unwanted levels of vibrations, by control signals that counteract disturbance signals by altering system dynamics or by cancellation. Altering system dynamics makes the system less excitable at certain frequency bands. If dynamical alterations are difficult to bring about by redesign, then control inputs can be provided to destructively interfere with vibrating system outputs.

Usually the signals in ANVC are sound waves and vibrations, that can be measured by microphones and accelerometers, respectively. The core task of ANVC is to design a control scheme, including implementation of control algorithm, set up of sensors and actuators, to produce destructive interference with the unwanted disturbance (noise or vibration) according to the mea-

sured reference and output signals. In the following subsections, the historical background is reviewed first. It is followed by a review of current methods for the development of ANVC technology. Design considerations of ANVC are briefly discussed in the last subsection.

1.1.1 Historical background

The idea of cancelling adverse noise and vibration by adding compensation signals is not new. This general idea of wave interference with anti-phase first emerged in optics in the 17th century. Initial physical principles of wave interference in acoustic field can be attributed to the research work of Lord Rayleigh in the late 1870's [116]. He produced interference of two sound fields using two electromagnetically synchronized tuning forks. This experimental result proved that the interference in acoustics performs in the same way as in optics, and it is the essentially common to all wave phenomenons.

The principle proved by Lord Rayleigh can be graphically described by Fig. 1.1. The top wave in Fig. 1.1 can be considered as original wave source, such as: unwanted noise or vibration, the bottom wave can be considered as control wave produced by a control actuator. They have almost same amplitude but opposite phase, i.e., the phase delay is π. The two waves are interfering and the residual wave is shown on the right hand side. There is nearly no wave left except some perturbation.

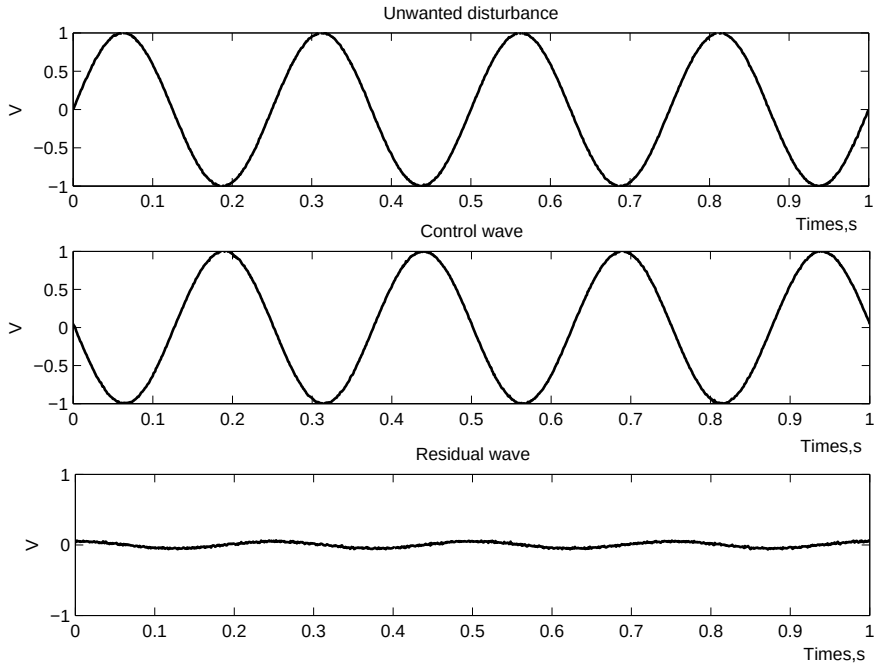

FIGURE 1.1: Principle of interference for ANVC

Essentially the objective of ANVC is to achieve a field with minimal wave amplitude by introducing some suitable interference waves. In the early 1930's, the idea of ANVC was summarized and patented by Lueg to control noise in an airduct [87]. This patent, with an appended

sheet of drawings, is generally considered as the first written document on active noise control. Since then, a great amount of work has been carried out to develop and develop methodologies for the control of noise and vibration in various application areas.

In 1950's, Olson [101, 102] performed the first laboratory experiments about ANVC and given a farsighted prediction of its application possibilities. Before the extension of computer development of digital signal proceeding (DSP) technology, ANVC was achieved by analogue equipment. The low-precision and complexity of analogue components prevented ANVC extension from laboratory to practical application [36] and therefore retarded the development of further theory.

During the 1970's the research discipline of ANVC was fueled by the widespread introduction of DSP technology in research and industry. K. Kido was probably the first to use computers in ANVC [67]. Since the 1990's versatile and high-performance DSP processors and computers have become popular and cheaper. Digital control for ANVC, based on DSP and computer technologies can realize fast, Multi-Input Multi-Output (MIMO) controllers with a reasonable cost [34]. They have speeded up the application of of ANVC into real engineering problems. ANVC has been widely used in vehicles[14, 22, 76, 81, 118], transport systems [55, 63, 118], rotating machinery[19, 28, 166, 171], high precision equipment [33, 38, 132, 163, 167] and consumer electronics [1, 31, 43, 44, 91, 126]. The most popular application in recent years seems to be in the structure [18, 39, 42, 49, 56, 61, 62, 84, 88, 100] and fluid areas [42, 66, 74, 127, 155, 158].

As a fast developing application topic, ANVC has utilized many techniques from different fields. Wavelet analysis has been used in ANVC to solve the system with complex disturbances [17, 19]. With the development of artificial intelligence (AI), some approaches from AI such as fuzzy logic controllers [79, 83, 112], Neural Network (NN) controllers [59, 77, 109, 111, 130, 135] and Genetic Algorithms (GA) [49, 154] have also been successfully used in ANVC. Stochastic control method [152] has also been adopted in ANVC problems [15, 41, 90, 122] . In order to suppress the disturbance in some large field or structures, distributed control in ANVC [40, 69, 137] has been a hot topic in recent years. Another popular practical approach is semi-active control which is the hybrid method combining active controls and passive suppressions [2, 66, 164, 168, 169].

1.1.2 Classification in ANVC

As the name of ANVC states, active noise control (ANC) and active vibration control (AVC) are two major topics in ANVC. Active noise control (ANC) aims to cancel unwanted sound. With man's increasing demand for a better living and working conditions, a lot of attention is paid to noise pollution and noise control is an active engineering field. Successful example products are noise cancellation headphones, headrests and noise cancellation speaker systems in vehicles. Active vibration control (AVC) aims to suppress mechanical vibrations at low frequencies which

are typically below 1kHz. Application areas of AVC typically includes mechanical systems, industry equipment, complicated structures and highrise building.

According to the physical features of disturbances, the disturbance signals in ANVC can be either broadband spectrum or narrow-band spectrum, with either stochastic signals or deterministic signals. When the disturbance source is unpredictable, the disturbance is often stochastic and broadband such as noise pollution in the environment and micro vibration in complicated equipment and structures. When the disturbance is excited by a few generators, it is always a periodic signal, such as the vibration caused by motors in industrial machineries.

From the aspect of the different controller structures, feedback and feed-forward controllers are the two most common and basic types. While feed-forward controller have the advantage of having a more stable performance, feedback controller are suitable in cases where no disturbance-reference signals are available. Some other controller structures such as Two Degrees of Freedom (TDF) controllers can be considered as a combination of these two types.

According to the time when controllers are designed, control methods for ANVC can be divided into two categories: off-line model based control designs and the on-line control designs.

Off-line model-based designs in ANVC are based on models which are known a priori or off-line identified. Its advantage is that it can deal with a broad frequency band of disturbances and, if conditions permit, it can yield high performance cancellation. This approach has been well developed and widely used in some instruments and platforms which have relative simple structures but high-precision requirement. As a generalized concept of 'off-line design', the methods of passive noise and vibration control can also be classified into an 'active' category if the passive components, such as sound absorbers and dampers, can be considered as a certain kind of 'controllers'.

On-line control design is developed to deal with the problems when off-line modelling can not present complete dynamics of control objectives under different work conditions. It will be discussed in detail in the literature review in the next Chapter.

1.1.3 Design consideration of ANVC

There are several aspects to be considered when designing a successful ANVC system.

The first important step in control design is to clarify the control objective. Considering the spectrum of disturbances (noise or vibration), the practical disturbance signals can be sorted as broad band signals which can include some stationary random signals and periodical signals which have finite discrete spectrum.

Regarding system dynamics, there are Linear Time Invariant (LTI) and Linear Time Variant (LTV) systems, Nonlinear Time Invariant (NTI) and Nonlinear Time Varying (NTV) systems.

Considering the input/output channels of the system, there are Single Input Single Output (SISO) and Multiple Input Multiple Output (MIMO) systems.

An important choice to be made is to select a proper control algorithm. While model based control methods may require some complicated computations to give a satisfactory model and robust bounds on uncertainties, they are more suitable for handling broadband control problems and off-line or time-invariant cases. Adaptive methods can track the change the system dynamics and tune controllers directly, they are widely used in realtime implementations in practice. Also some relatively new techniques, such as sliding model control, and neural network control, have been used for nonlinear ANVC.

The last stage of control design is to choose some proper hardware implementation. Controllers can be implemented either with a supervisor computer (such as a personal computer, PC) through serial ports, or directly with independent Digital Signal Processor (DSP) boards through some other I/O ports that are more suitable for some complicated problems such as the MIMO cases.

Another practical problem in hardware implementation is to find proper actuators/sensors that can provide the control actions and data acquisitions in the frequency band of interest. In noise control, microphones and speakers of various technologies are usually chosen as sensors and actuators, respectively. In vibration control, piezoelectric elements and accelerometers are the most commonly used sensors, there are varying choices for vibration actuators, such as shakers and tunable dampers.

1.2 Overview of the Thesis

As discussed above, the essential idea of ANVC is based on wave interference that is a concept in the frequency domain. While the most of popular control methods, such as the Filter-X LMS method, are based on analysis in the time domain, this thesis will propose a new adaptive control method for ANVC that is based on an analysis in the frequency domain.

The proposed adaptive control method of this thesis is based on a new gradient estimation theory which relies on frequency domain (FD) analysis of system dynamics and signals. The proposed adaptive control method is first tested on a simulation based platform to show its effectiveness. Experimental work is also implemented to demonstrate the practical performance of the algorithm.

1.2.1 Contribution of the thesis

The following subsections divide the contributions of this thesis into four areas: gradient estimation, iterative tuning algorithms, implementation issues and test work of the proposed control methodology.

The first contribution is the development of a new gradient estimation theory based on the analysis of the frequency response of the system dynamics and of the signals' spectrum. Using the frequency response of closed-loop dynamics, the relationship between the infinitesimal change in controller parameters and infinitesimal change in the spectrum of the system output is examined. When the control performance can be expressed in the frequency domain, the gradient of the control performance with respect to the controller parameters can be completely expressed within the frequency domain. The derivation of the proposed gradient estimate also reveals similarities with other gradient estimation methods found in the literature.

The second contribution is to develop iterative-tuning algorithms for ANVC problems with periodic disturbance. While the theory mentioned above is the general framework for most of the adaptive control problems, a detailed control algorithm is studied for the special case of ANVC with periodic signals. In this algorithm and its permutations, no explicit or complete dynamical model is required to be known a priori. There is only one initial experiment required to estimate the gradient in Single Input Single Output (SISO) cases. In Multiple Input Multiple Output cases (MIMO), the number of extra initial experiments is the same as the number of input channels of the secondary dynamics. During adaptation there is only one experiment per iteration which allows implementation in the form of an adaptive controller. Furthermore, the proposed algorithms can be used to tune feed-forward controllers as well as feedback controllers.

The third contribution is the extension the frequency domain iterative tuning method to nonlinear systems. While the nonlinear dynamics can be represented with the Generalized Frequency Response Function (GFRF) in the frequency domain, the nonlinear dynamics can be locally linearized as multiplication between a derivative matrix of GFRF and the spectrum vector of input changes. Similarly to the linear case, the gradient can be expressed via the nonlinear closed loop dynamics and it can be estimated through some experiments that realize small changes in the input spectrum.

The last contribution is implementation methods for the proposed iterative tuning method in simulation and on an experimental system. The proposed control algorithm is tested on three simulated platforms under SIMULINK, and an air duct system under laboratory conditions. All the results have proved the effectiveness and practicality of the proposed control algorithms in real engineering environments.

1.2.2 Structure of the thesis

There are nine chapters in this thesis. The main body of the thesis can be divided into three parts. Part one is the literature review in Chapter 2. Part two is the theoretical part of the proposed control method including Chapters 3, 4 and 5, Chapter 6 addresses the control problem of a nonlinear plant. Part three is on testing of the proposed control algorithms in simulated and experimental platform in Chapters 7 and 8. Chapter 9 draws conclusions and points to future work.

Considering the broad applicability, flexibility and practicality of discrete time control systems in ANVC, the control algorithms in this thesis are based on discrete models and controllers if not stated otherwise.

The detailed structure of the thesis for the rest of the chapters is as following:

In **Chapter 2** a literature review on adaptive control methods in ANVC is presented. As one of the most widely used control methods, Filtered reference Least Mean Square (filter-X LMS) method is first explained. Some permutations of this method, which are also based on the Least Mean Square (LMS) algorithm, are briefly reviewed. Some other recently developed adaptive control methods for ANVC, which are time domain Iterative Feedback Tuning (IFT) control and Model Free Frequency Domain Tuning (MF-FDT) control, are also reviewed here.

In **Chapter 3** the gradient estimation theory for SISO cases is introduced through system analysis in the frequency domain. Through analyzing the infinitesimal change of the output spectrum caused by the perturbation of the controller parameters, the gradient of the output spectrum is derived with respect to the tunable parameters of feedback and feed-forward controllers. Given the average quadratic performance criteria, using Parseval's Theorem, the gradient of the performance criterion with respect to the controllers parameters is completely expressed within the frequency domain formalism. This gradient estimate can then be used to tune both feedback and feed-forward controllers. Comparison with some other adaptive control methods in ANVC is made that clarifies the essential features of the proposed estimation theory.

In **Chapter 4** the iterative-tuning algorithm for SISO linear systems is based on the gradient estimate of the criterion that evaluates periodic signals. ANVC with periodic signals is especially suitable control problem of the proposed algorithm. Without known a priori model the proposed algorithm can estimate the partial Frequency Response Function (FRF) of the closed-loop dynamics through two different input-output experiments and make controller tuning adaptively. Some important issues in implementing the algorithm are discussed. Through introducing Frequency Selective Filters (FSF), the modified algorithm of FSF-FD-IT is developed to improve the performance and stability of an ANVC systems. The simplicity and flexibility of the obtained control structure and operation are compared to some other adaptive control methods in ANVC.

In **Chapter 5** the extension to MIMO systems is studied. The derivative matrix of extended Frequency Response Function is defined for MIMO systems. As expected there is a lot of similarity with the SISO case. The only major difference is the format of spectrum vectors of input and output signals and the frequency response matrix of the system. Having obtained a gradient estimate in the frequency domain, a practical tuning algorithm is straightforward in the MIMO case.

In **Chapter 6** the extension to nonlinear cases is studied. The derivative matrix of the frequency responses for nonlinear dynamics is studied which can be well explained by the Generalized Frequency Response Functions (GFRFs) for nonlinear systems. After this step, the estimate

technique for gradient estimation in the frequency domain for nonlinear system is almost the same as that in linear cases apart from some difference handling matrices and signal and spectrum vectors. Having yielded the gradient estimation theory, the practical tuning algorithms and some implementation issues are also discussed for the nonlinear case.

In **Chapter 7**, the simulation work based on SIMULINK is performed. The effectiveness of the algorithm is tested by three SIMULINK based examples, i.e., a SISO example, a MIMO example and a nonlinear example. The robustness of the algorithm against the error in the frequency domain is also shown through some extended examples.

In **Chapter 8**, an experimental systems is set up to test the proposed iterative-tuning algorithm. An air duct system is selected to implement the proposed control algorithm. The commonly used Texas Instrument TMS320C44 CPU based DSP board is used as a digital controller. Results are promising for the real engineering applications.

In **Chapter 9** conclusions are drawn. Advantages and characters of the proposed control method are emphasized . Future research directions are pointed out.

Chapter 2

LITERATURE REVIEW

This Chapter provides a literature review on current control methods in ANVC. It is divided into four parts. The first part is a brief panorama of control design in ANVC. Then the most commonly used Filter-X LMS (FX-LMS) method and its recently improved versions are discussed. Finally, two recent adaptive control methods in ANVC: the Iterative Feedback Tuning (IFT) control and Model-Free Frequency Domain Tuning (MF-FDT) control are discussed.

For simplicity, all the control methods are presented under SISO conditions in this Chapter.

2.1 Current development of ANVC

By now ANVC has achieved a lot both in theory and applications. The theoretical development of algorithms for ANVC can be split into two major classes. (1) model based control which aims to build a proper model for the system dynamics and use that for control; (2) direct adaptive control in which aggregated estimation and control is performed.

Today, the model-based control approach has been most flourishing research field in ANVC.

In the model-based control, mismatch between nominal models and real dynamics is one of the key problems. In general, it is often difficult to build a complete control system to meet all the changing conditions of a real system with an off-line model. The working conditions of a real plant and system are always affected by some uncontrollable factors, for example, the plant can be time varying in ANVC systems.

Researchers have developed some techniques aiming to either to build more reliable models or design controllers tolerating the mismatch which derived from two of the most important improvements in control theory: the concept of model uncertainty [73, 108, 129, 148] and robust control design [29, 170], which yields error-tolerant controllers for uncertain models with bounded descriptions of uncertainty. In the last ten years, the synergy and duality of modelling

and controller design [47, 51, 71, 123] has become one of the important topics about the model-based control. Especially, the concepts of model validation [72, 85, 157] and unfalsification control [70, 72, 121, 156] synergizes and dualizes system identification and control design in adaptive control [139, 143, 144, 145, 159] that make adaptive control more stable and robust. Nowadays, model uncertainty and robust control design has been considered more and more frequently in ANVC problem [16, 48, 57, 75, 84, 100, 153, 160].

Another important topic in model-based control is modelling nonlinear dynamics in that nonlinearity exists everywhere in the real world. While some researchers use model uncertainty to bound the mismatch between linear models and real nonlinearity [13], some new control methodologies are introduced into ANVC in complicated mechanical systems, such as: sliding mode control [80, 133, 134], switching control [86] and predictive control[162].

Direct adaptive control is one of the most successful methods in ANVC, which either requires no explicit modelling and updates controllers directly or performs online system identification and then design controllers with the online estimated model. A key to adaptive control is to find appropriate 'directions' to update controllers. The ability of self-improvement and automatically tracking the change of the system is the most attractive feature of this method [124]. Since most of real plants controlled by realtime ANVC have very complex dynamics and are varying under different working conditions, the online controller design method has been the mainstream approach in ANVC in industry and comfort applications.

One of the important milestones in algorithmic developments is the filter-x LMS method that emerged in early 1980s [97]. The filtered-x LMS (FXLMS) method is the most successful online design method in ANVC that has been widely used in commercial products for ANVC. Many variations of the filter-x LMS method have been developed to expand the applicability of initial idea of the filter-x LMS method. At the same time convergence and stability problems of adaptive control are also active research topics in the area of ANVC.

Some other adaptive control methods, such as Iterative Tuning method [146] and Model-Free Frequency Domain Tuning (MF-FDT) [95], are also developed for ANVC problems.

While this thesis will propose a new adaptive control method, some further discussion about the direct adaptive control in ANVC will be presented to provide a clearer understanding of direct adaptive methods in ANVC in the following parts of literature review. At first the classical filtered-x LMS method and some of its permutations are reviewed for their popularity. Afterwards, some recently developed adaptive control algorithms in ANVC are introduced.

2.2 Filter-X LMS method

In ANVC the filtered-x LMS (FXLMS) method [97] is the most widely used and well studied control algorithm today. It is a family member of adaptive filters where the reference signal is filtered through a model of the secondary path dynamics.

2.2.1 Prototype of FXLMS

In 1980, D. R. Morgan proposed Filtered-X LMS algorithm for active noise control in [97]. A block diagram of the Filtered-X LMS is given in Fig. 2.1.

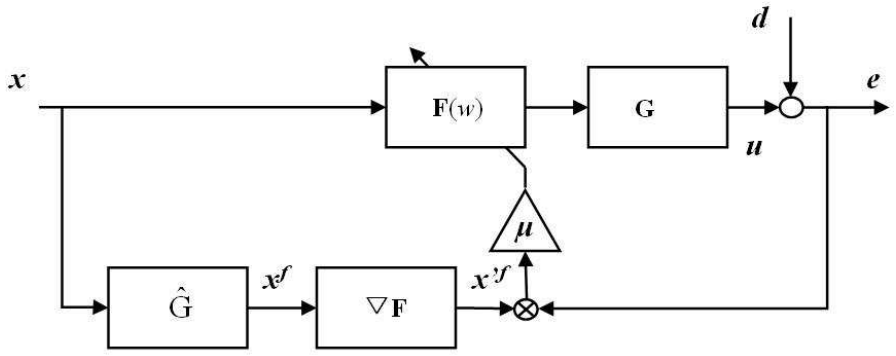

FIGURE 2.1: Block diagram for the FXLMS algorithm

Considering the feed-forward system in Fig. 2.1, the output vector can be expressed as

$$\boldsymbol{e}(\boldsymbol{w}) = \boldsymbol{d} + GF(\boldsymbol{w})\boldsymbol{x} \tag{2.1}$$

Given output series $\boldsymbol{e} = \{e(0), \ldots, e(L-1)\}$ with length L, the most commonly used average quadratic cost function is defined as

$$J(\boldsymbol{w}) \doteq \frac{1}{L}\sum_{t=0}^{L-1} e^2(t) \tag{2.2}$$

The objective of ANVC is to tune a parameter vector $\boldsymbol{w}$ of the controller $F(\boldsymbol{w})$ to minimize performance $J(\boldsymbol{w})$ as (2.2).

The gradient of performance $J(\boldsymbol{w})$ with respect to the parameter vector $\boldsymbol{w}$ can be written as

$$\nabla J(\boldsymbol{w}) = 2\sum_{t=0}^{L-1} e(t)\frac{\mathrm{d}e(t)}{\mathrm{d}\boldsymbol{w}}, \tag{2.3}$$

where

$$\frac{\mathrm{d}e(t)}{\mathrm{d}\boldsymbol{w}} = G\frac{\mathrm{d}F(\boldsymbol{w})}{\mathrm{d}\boldsymbol{w}}\boldsymbol{x}. \tag{2.4}$$

Due to the linearity of the system the order of cascaded dynamical blocks can swapped and Eqn. (2.4) can be rewritten as

$$\frac{\mathrm{d}e(t)}{\mathrm{d}\boldsymbol{w}} = \frac{\mathrm{d}F(\boldsymbol{w})}{\mathrm{d}\boldsymbol{w}}G\boldsymbol{x}, \tag{2.5}$$

which leads to

$$\nabla J(\boldsymbol{w}) = 2\sum_{t=0}^{L-1} e(t)\frac{\mathrm{d}F(\boldsymbol{w})}{\mathrm{d}\boldsymbol{w}}G\boldsymbol{x}, \tag{2.6}$$

According to (2.6), the tuning 'direction' of feed-forward controller $F(\boldsymbol{w})$ is the steepest descent, which can be directly calculated using the output signal $\boldsymbol{e}$ and the reference signal $\boldsymbol{x}$ passing through G. If the real dynamics G can be represented with a nominal model $\hat{G}$, it can be used to filter the reference signal $\boldsymbol{x}$.

Denoting a signal

$$\boldsymbol{x}^f = \hat{G}\boldsymbol{x}, \tag{2.7}$$

the update of controller $F(\boldsymbol{w})$ moves the parameter vector $\boldsymbol{w}$ along the negative gradient direction

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - 2\mu\sum_{t=0}^{L-1}\boldsymbol{e}(t)\nabla F(\boldsymbol{w})\boldsymbol{x}^f. \tag{2.8}$$

Normally a Finite Impulse Response filter (FIR) is used for ANVC for its simplicity and stability. The partial derivative of p-order FIR is

$$\frac{\partial F(\partial \boldsymbol{w})}{w_n} = z^{-n}, n = 0, \ldots, p \tag{2.9}$$

where z^{-n} is the time shifter with n sample time delay and w_n is the coefficient of z^{-n}.

Eqn. (2.9) can be expressed with the measurable signal $\boldsymbol{x}_t^f$ as

$$\nabla F(\boldsymbol{w})\boldsymbol{x}^f = \boldsymbol{x}_t'^f \doteq [x^f(t), x^f(t-1), \ldots, x^f(t-p)]^{\mathrm{T}}. \tag{2.10}$$

Eqn. (2.8) then can be written with simpler form as

$$\boldsymbol{w}_{i+1} = \boldsymbol{w}_i - 2\mu\sum_{t=0}^{L-1} e(t)\boldsymbol{x}_t'^f. \tag{2.11}$$

2.2.2 FXLMS with Online Secondary Path Estimate

In the original FXLMS method it was assumed that the model of the secondary path G is known a priori. In order to solve time-varying problems and to improve FXLMS, a realtime estimate of the secondary path was first reported in 1990 [131].

Without losing generality, the reference signal $\boldsymbol{x}$ is often obtained from a sensor with dynamics S. Assuming that the disturbance $\boldsymbol{d}$ can be expressed with the reference signal $\boldsymbol{x}$ filtered through

a inverted transfer function S_I, i.e. $\boldsymbol{d} = S_I\boldsymbol{x}$. Then the output signal in (2.1) can be rewritten as

$$\boldsymbol{e} = S_I\boldsymbol{x} + GF(\boldsymbol{w})\boldsymbol{x} \tag{2.12}$$

Assuming that the transfer functions G and P_I can be estimated by n_G and n_S-order FIR models with coefficients $\{g_0, \ldots, g_{(n_G-1)}\}$ and $\{s_0, \ldots, s_{(n_S-1)}\}$ respectively, it is convenient to introduce the following vector notations as

$$\Theta(t) = [g_0, \ldots, g_{(n_G-1)}, s_0, \ldots, s_{(n_S-1)}]^{\mathrm{T}} \tag{2.13}$$

and

$$\Phi(t) = [u(t), \ldots, u(t-n_G+1), x(t), \ldots, x(t-n_S+1)]^{\mathrm{T}}. \tag{2.14}$$

It is straightforward to to write the output signal as

$$e(t) = \Theta(t)\Phi^{\mathrm{T}}(t). \tag{2.15}$$

All the elements in the vector $\Phi(t)$ are obtainable through measurements or calculation and the output signal $e(t)$ can be measured. Only the elements in the vector $\Theta(t)$ are required to be estimated. A Recursive Least Square (RLS) algorithm [35] to estimate these unknown variables is

$$\hat{\Theta}(t+1) = \hat{\Theta}(t) + \mu\frac{\Phi(t)}{n + \Phi(t)\Phi^{\mathrm{T}}(t)}[e(t) - \hat{\Theta}(t)\Phi^{\mathrm{T}}(t)] \tag{2.16}$$

where μ is the adaption gain and the small, positive constant n is introduced to avoid division by 0. The convergence of the above algorithm can be guaranteed by the condition $0 < \mu < 2$.

Therefore, the recursive equation is solved by minimizing the cost function

$$J = \frac{1}{L}\sum_{t=0}^{L-1} E\{[\hat{\Theta}(t) - \hat{\Theta}(t-1)]^2\}, \tag{2.17}$$

with the constraint $e(t) = \Theta(t)\Phi^{\mathrm{T}}(t)$.

Using (2.16) models are obtained for the primary and secondary path and the FXLMS method is applied by (2.17).

2.2.3 Permutations of Filtered-X LMS

Being a family of adaptive control methods for ANVC, the FXLMS method has a lot of variations for different control applications.

In 1983 the filtered error LMS method (FELMS) was proposed for ANVC in [128]. As its name implies, FELMS filters the error output signals with a certain dynamics, which is generally the estimated model of secondary path dynamics.

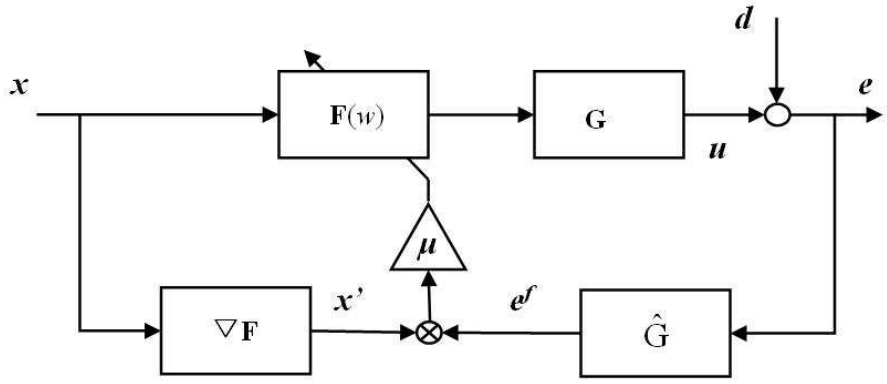

FIGURE 2.2: Block diagram of FELMS method

As a member of the family of FXLMS, the deduction process to deliver $\nabla J(\boldsymbol{w})$ is almost the same as that of FXLMS, which is referred in [128]. As shown in Fig. 2.2, given a p-order FIR feed-forward controller F, the update rule of controller parameter vector can be expressed as

$$\boldsymbol{w}_{i+1} = \boldsymbol{w}_i - 2\mu \sum_{t=0}^{L-1} e^f(t)\boldsymbol{x}'_t. \tag{2.18}$$

where $\boldsymbol{x}'_t := \frac{\partial F(\boldsymbol{w},x(t))}{\partial \boldsymbol{w}} = [x(t),\ldots,x(t-p)]^{\mathrm{T}}$ and $e^f(t) = \hat{G}\boldsymbol{e}$ is the filtered error.

Based on the idea of FELMS, FXLMS can be extended to feedback control for ANVC. The block diagram of feedback control with FELMS is shown in Fig. 2.3.

Given a feedback controller H, the update rule for the controller parameter vector can be expressed as

$$\boldsymbol{w}_{i+1} = \boldsymbol{w}_i - 2\mu \sum_{t=0}^{L-1} e^f(t)\frac{\partial H(\boldsymbol{w},e(t))}{\partial \boldsymbol{w}}. \tag{2.19}$$

where $e^f(t) = \hat{K}\boldsymbol{e}$ is $\hat{K}$ filtered error and $\hat{K}$ is the nominal closed-loop dynamics, i.e.,

$$K = \frac{G}{1-GH}. \tag{2.20}$$

After the 1990s, the research of FXLMS focused on the improvement of FXLMS algorithm since FXLMS and FELMS have covered the structures of most control systems. In order to satisfy the online demands and hardware limitations, one of the most important topics aimed to reduce the computation and memory requirements. Some discussion of this can be found in [3, 6, 25, 26, 27, 99, 150]. Another active topic is the study of stability conditions, robustness

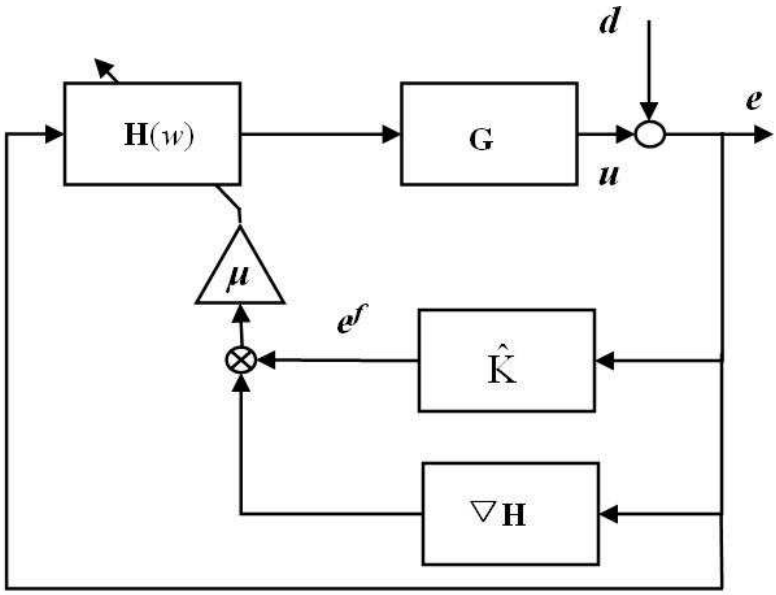

FIGURE 2.3: Block diagram of feedback control with FELMS

and convergence of FXLMS, which can be referred in [20, 96].

Adjoint FXLMS is proposed in [150] which can be used to control nonlinear dynamics. When the NARMAX model or Volterra/Wiener function model is introduced to represent the nonlinear dynamics, filtered-x LMS has been used to implement nonlinear ANVC [21, 23, 136].

2.3 Iterative Feedback Tuning Control

Apart from FXLMS, other methods of adaptive control for ANVC, such as self-tuning control, were not widely used and well developed until the end of the 1990s. The Iterative Feedback Tuning (IFT) method for ANVC is one of the successful examples of extending the 'negative-gradient tuning principle' of FXLMS.

2.3.1 Iterative feedback tuning in the time domain

In the middle of 1990s, IFT [52, 54] was first reported to solve servo control problems. It is essentially a self-tuning method in the time domain, which solves the gradient of control performance in the time domain.

A general SISO and LTI system with tunable feedback controller $H(\boldsymbol{w}_H)$ and feed-forward controller $F(\boldsymbol{w}_F)$ can be graphically represented as shown in Fig.2.4. It can be used to illustrate either ANVC problems where $\boldsymbol{d}$ is an unknown disturbance, or servo control problems where $\boldsymbol{d}$ is considered as the desired trajectory signals.

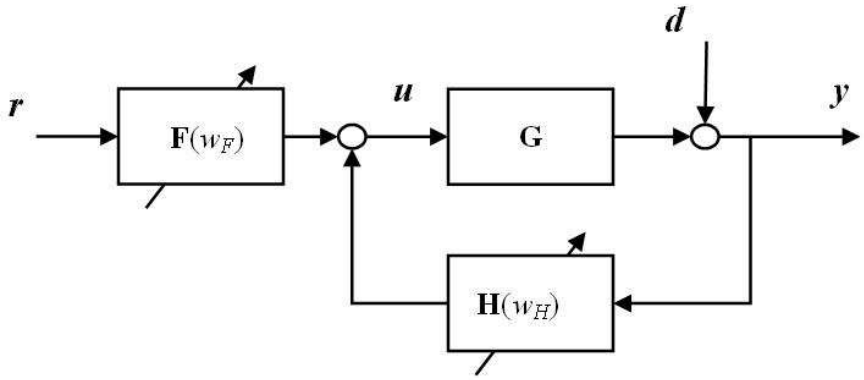

FIGURE 2.4: Block diagram of a general LTI control system

Considering an ANVC system as Fig.2.4, the output $y(t)$ can be represented as

$$y(t) = \frac{GF}{1 - GH} r(t) + \frac{GH}{1 - GH} d(t) \tag{2.21}$$

In the IFT [54] the key idea is to estimate the partial derivatives of $y(t)$ with respect to $\boldsymbol{w}_H$ using the following equation:

$$\frac{\partial y(t)}{\partial \boldsymbol{w}_H} = \frac{\partial H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} \frac{G}{1 - GH} y(t) \tag{2.22}$$

Similarly, the estimation of partial derivatives of $y(t)$ with respect to $\boldsymbol{w}_F$ is given in [93] as

$$\frac{\partial y(t)}{\partial \boldsymbol{w}_F} = \frac{\partial F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} \frac{G}{1 - GH} r(t) \tag{2.23}$$

Assuming an averaged quadratic performance criterion for SISO system such as

$$J(\boldsymbol{w}) := \frac{1}{N} \sum_{t=0}^{N-1} y^2(t) \tag{2.24}$$

it is straightforward to write

$$\frac{\partial J}{\partial \boldsymbol{w}_H} = \frac{2}{N} \sum_{t=0}^{N-1} y(t) \frac{\partial y(t)}{\partial \boldsymbol{w}_H} \tag{2.25}$$

and

$$\frac{\partial J}{\partial \boldsymbol{w}_F} = \frac{2}{N} \sum_{t=0}^{N-1} y(t) \frac{\partial y(t)}{\partial \boldsymbol{w}_F} \tag{2.26}$$

As described in Fig. 2.5 and Fig. 2.6, by injecting $y(t)$ and $r(t)$, $\frac{\partial y(t)}{\partial \boldsymbol{w}_H}$ and $\frac{\partial y(t)}{\partial \boldsymbol{w}_F}$ are the outputs of the close-loop system $\frac{G}{1-GH}$ followed by post-filters $\frac{\partial H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}$ and $\frac{\partial F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}$, respectively.

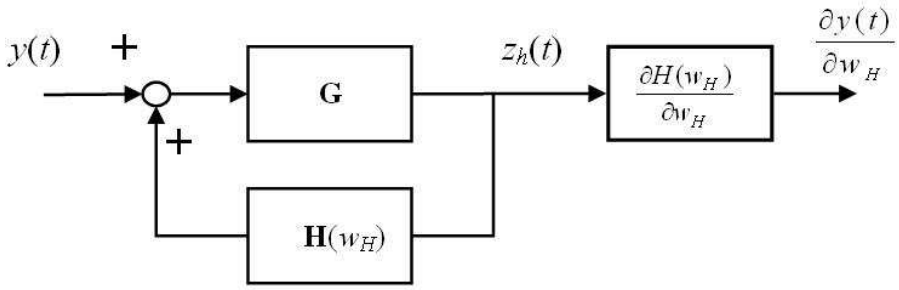

FIGURE 2.5: Block diagram of feedback tuning

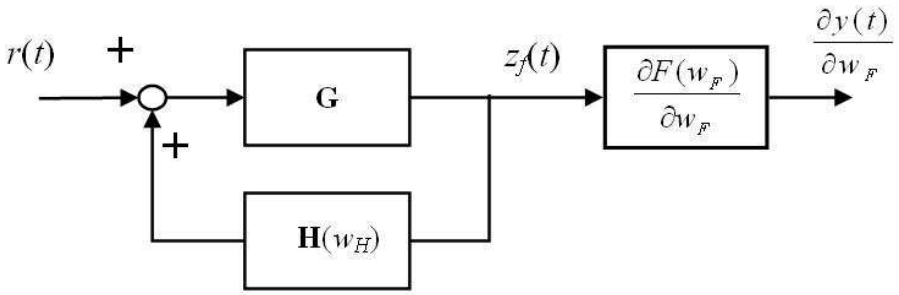

FIGURE 2.6: Block diagram of feed-forward tuning

Note that $H(\boldsymbol{w}_H)$ and $F(\boldsymbol{w}_F)$ are always known by designers, post-filtering by filters $\frac{\partial H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}$ and $\frac{\partial H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}$ can be computed after obtaining $z_h(t)$ and $z_f(t)$ as shown in Fig. 2.5 and Fig. 2.6, which avoids multiple real experiments for all parameters in $\boldsymbol{w}_H$ and $\boldsymbol{w}_F$. Therefore, in IFT, after one normal experiment, two extra gradient experiments are performed to produce $z_H(t)$ and $z_F(t)$, which lead to the gradient as

$$\frac{\partial J}{\partial \boldsymbol{w}_H} = \frac{2}{N}\sum_{t=0}^{N-1} y(t)\frac{\partial H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} z_h(t) \tag{2.27}$$

and

$$\frac{\partial J}{\partial \boldsymbol{w}_F} = \frac{2}{N}\sum_{t=0}^{N-1} y(t)\frac{\partial F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} z_f(t) \tag{2.28}$$

In IFT the methodology is relatively simple and practicable. Given an initial stable controller, the controller can be tuned completely through experimental data without any model of the system.

In this scheme feedback and feed-forward controllers are tuned separately based on (2.22) and (2.23), which requires two additional experiments. Also using the normal experiments, altogether IFT had to perform $3 \times N_T$ experiments to implement N_T-step tuning iterations.

In order to distinguish it from the proposed frequency domain method, the method illustrated in Fig. 2.5 and Fig. 2.6 is called time domain iterative feedback tuning (IFT) in the rest of this thesis.

2.3.2 Online IFT for ANVC

The approach described in the previous subsection is the most basic and general idea of IFT. Note that in Fig. 2.5 and Fig. 2.6 there is no disturbance $\boldsymbol{d}$, which means that the disturbance is switched off in the gradient experiments, the previous method is only suited for off-line tuning for ANVC. In the case of periodic disturbances, this problem can be solved by using identical disturbances in different periods and self-tuning method can be adopted to perform online tuning for ANVC.

In [93] online IFT was introduced to control ANVC with periodic disturbances and has been tested in a series of laboratory experiments [92]. In this approach, it is not necessary to switch off the disturbance signal and therefore some modification of the experiments described by (2.22) and (2.23) has to be performed for the online tuning application.

In the online application disturbance $\boldsymbol{d}$ has to be considered in the gradient experiments. Considering periodic disturbance $\boldsymbol{d}$ with common period N, in order to cancel the impact of $\boldsymbol{d}$, (2.21) can be rewritten as

$$\frac{GF}{1-GH}r(t)+\frac{G}{1-GH}d(t+N)-y(t)=0 \tag{2.29}$$

It is straightforward to rewrite (2.22) and (2.22) as following:

$$\frac{\partial y(t)}{\partial \boldsymbol{w}_H}=\frac{\partial H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}(\frac{G}{1-GH}y(t)+\frac{GF}{1-GH}r(t)+\frac{G}{1-GH}d(t+N)-y(t)) \tag{2.30}$$

$$\frac{\partial y(t)}{\partial \boldsymbol{w}_F}=\frac{\partial F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}(\frac{G}{1-GH}r_p(t)+\frac{GF}{1-GH}r(t)+\frac{G}{1-GH}d(t+N)-y(t)) \tag{2.31}$$

As illustrated in Fig. 2.7 and Fig. 2.8, the online gradient experiments for $H(\boldsymbol{w}_H)$ and $F(\boldsymbol{w}_F)$ produce $\boldsymbol{z}_h$ and $\boldsymbol{z}_f$ respectively. Using (2.25) and (2.26), the online estimate of the gradient of the performance $J(\boldsymbol{w}_H,\boldsymbol{w}_F)$ is straightforward.

Noting the assumption of LTI system, the frequency response function (FRF) can be considered independent with respect to the frequency. Frequency Selective Filter based on IFT [93] (FSF-IFT) was also developed for multi-tone disturbances, in which narrow bandwidth filters were used to split one complicated tuning task for all multi-tone disturbance into multiple tuning

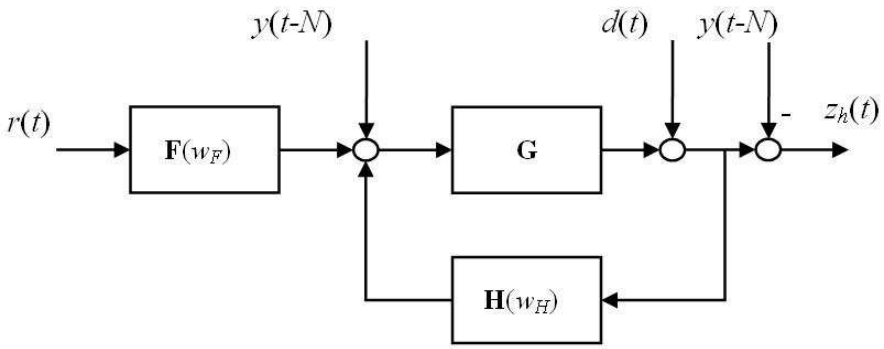

FIGURE 2.7: Gradient experiment for H in online IFT

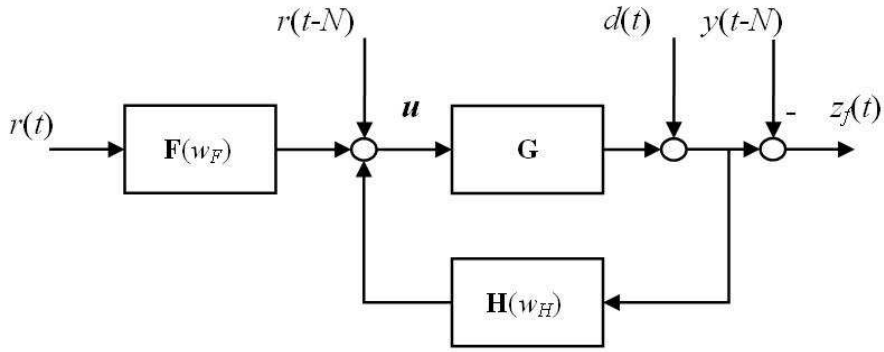

FIGURE 2.8: Gradient experiment for F in online IFT

subtasks for each of the dominant tones of the disturbance spectrum. In FSF-IFT, it is not necessary to wait a long common period of all multi-tones to perform gradient experiments but to perform gradient experiments for each single tone after its period. While FSF-IFT gives a more complicated structure by using some extra frequency selective filters (FSF), its advantage is obviously that it it results in a more effective tuning process.

2.4 Model-free Frequency Domain Tuning Control

In the previous sections the mentioned adaptive control methods in ANVC are all based on analysis in the time domain. The control methods based in the frequency domain have some essential advantages in the application of ANVC since the physical foundations of ANVC are anti-phase compensation in the frequency domain.

In [94, 95], a Model-Free Frequency Domain Tuning (MF-FDT) method was proposed for ANVC with periodic disturbances.

Considering an LTI ANVC system as illustrated in Fig. 2.9, the output of the system can be expressed in a vector format

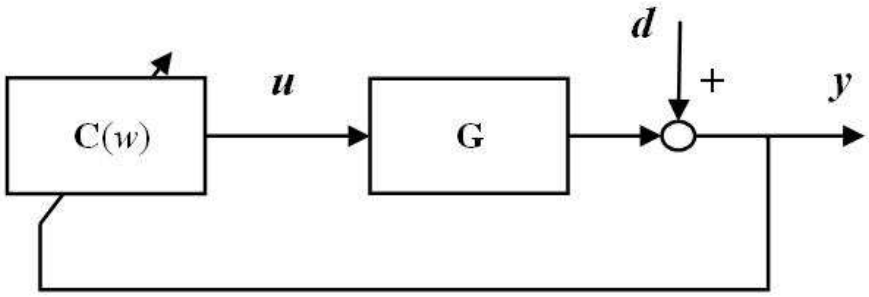

FIGURE 2.9: Block diagram of MF-FDT for ANVC

$$\boldsymbol{y} = \boldsymbol{d} + G\boldsymbol{u} \tag{2.32}$$

which has optimal solution of control action as

$$\boldsymbol{u} = -\frac{\boldsymbol{d}}{G} \tag{2.33}$$

Considering expressions of $\boldsymbol{y}$, $\boldsymbol{d}$,$\boldsymbol{u}$ and G in the frequency domain, all signals and systems can be represented by complex gains as

$$\begin{aligned} \boldsymbol{y} &= y_r + jy_i, \\ \boldsymbol{d} &= d_r + jd_i, \\ G &= G_r + jG_i, \\ \boldsymbol{u} &= u_r + ju_i, \end{aligned} \tag{2.34}$$

It is straightforward to rewrite (2.32) with the complex gains as

$$\boldsymbol{y} = \begin{bmatrix} y_r \\ y_i \end{bmatrix} = \begin{bmatrix} d_r \\ d_i \end{bmatrix} + \begin{bmatrix} G_r & -G_i \\ G_i & G_r \end{bmatrix} \begin{bmatrix} u_r \\ u_i \end{bmatrix} = \boldsymbol{d} + \boldsymbol{G}\boldsymbol{u} \tag{2.35}$$

where $\boldsymbol{y}$, $\boldsymbol{d}$ and $\boldsymbol{u}$ are matrix formats of signals, and $\boldsymbol{G}$ is the matrix format of system dynamics.

An average quadratic cost function is to be minimized. The control criteria in the frequency domain are given by

$$J(\boldsymbol{u}) := \frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\boldsymbol{y} = \frac{1}{2}(y_r)^2 + \frac{1}{2}(y_i)^2 \tag{2.36}$$

From (2.35), it is straightforward to get the gradient of J with respect to control action $\boldsymbol{u}$ as

$$\nabla J(\boldsymbol{u}) = \boldsymbol{G}^{\mathrm{T}}\boldsymbol{y} \tag{2.37}$$

In (2.37), while $\boldsymbol{y}$ is measurable through experiments, the remaining problem is how to calculate the gradient without any prior information about the secondary path dynamics. In order to solve the above problem, similar to IFT, a two-stage tuning approach is designed to get derivative information about ∇J:

1. The first stage is to perform such common experiment as illustrated in Fig.2.9,
2. The second stage is to perform such an extra gradient experiment as designed in Fig. 2.10.

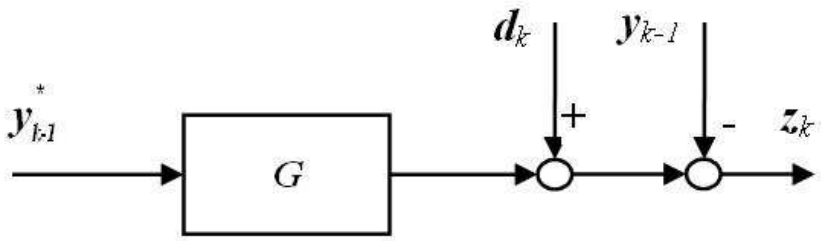

FIGURE 2.10: Gradient experiment being injected complex conjugate output in MF-FDT

In Fig. 2.10, the input of the gradient experiment, i.e., $\boldsymbol{y}_k^*$, is the complex conjugate of the previously recorded output in the common experiment. The index k indicates the data from the common experiment and the index $k+1$ refers to the current gradient experiment. Therefore, the output of the gradient experiment is given by

$$\boldsymbol{z}_k = \begin{bmatrix} z_r \\ z_i \end{bmatrix}_k = \begin{bmatrix} d_r \\ d_i \end{bmatrix}_{k+1} - \begin{bmatrix} d_r \\ d_i \end{bmatrix}_k + \begin{bmatrix} G_r & -G_i \\ G_i & G_r \end{bmatrix} \begin{bmatrix} y_r \\ -y_i \end{bmatrix}_k \tag{2.38}$$

Note that the disturbance is periodic, $\boldsymbol{d}_k$ and $\boldsymbol{d}_{k+1}$ are identical and eliminate each other, and (2.38) can be rewritten as

$$\begin{bmatrix} z_r \\ z_i \end{bmatrix}_k = \begin{bmatrix} G_r & -G_i \\ G_i & G_r \end{bmatrix} \begin{bmatrix} y_r \\ -y_i \end{bmatrix}_k \tag{2.39}$$

By obtaining the complex gain of $\boldsymbol{z}$ the gradient of $J(\boldsymbol{u})$ becomes

$$\nabla J(\boldsymbol{u}_k) = \begin{bmatrix} z_r \\ z_i \end{bmatrix}_k \tag{2.40}$$

Using a steepest descent method with a proper step size μ, the update rule for the feedback control action is

$$\begin{bmatrix} u_r \\ u_i \end{bmatrix}_{k+1} = \begin{bmatrix} u_r \\ u_i \end{bmatrix}_k - \mu \begin{bmatrix} z_r \\ z_i \end{bmatrix}_k \tag{2.41}$$

As shown in Fig. 2.9, although controller C looks like a feedback controller in the control structure, MF-FDT can be considered as a member of the family of Iterative Learning Controllers for the control action $\boldsymbol{u}$ is determined from the output $\boldsymbol{y}_{k-1}$ in the previous iteration but not feedback the output data $\boldsymbol{y}_k$ in the current iteration. Controller C acts as a signal generator and otherwise can be considered as a feed-forward controller with some fixed reference in current iteration.

This method has some advantages over FXLMS method as it

- avoids the requirement of reference signals, which are sometimes unavailable in the practice;
- reduces the computation by tuning the spectrum of control action $\boldsymbol{u}$ directly;
- does not demand prior knowledge of the secondary path G.

RLS algorithm has also be used in MD-FDT to avoid extra gradient experiments which is similar to RLS algorithm used in FXLMS and the detailed implementation is stated in [107].

2.5 Discussion about the Limitation

Although above three methods have been all successfully applied into real platform, they have their own limitations.

While FXLMS only tunes feedforward controllers, FELMS only tunes feedback controllers. At the same time, FXLMS requires nominal model $\hat{G}$ which is generally identified off-line. When using RLS online estimation, FXLMS is not true gradient based tuning any more due to the adaptive approaching of RLS algorithm.

IFT method has more complex structure which requires additional path to inject extra signals. At the same time, it requires more complicated operations in implementation which requires extra experiments in order to make each tuning.

MF-FDT method has the same drawbacks as ITF. It requires additional signal injection and extra gradient experiment.

In the following part of the thesis, an new iterative tuning method, FD-IT, will be developed to overcome above limitations, especially the drawbacks of IFT and MF-FDT.

2.6 Summary

This Chapter presented a brief review of control design for Active Noise and Vibration control (ANVC). After giving a short review about the background and developing the history of ANVC, three important adaptive control methods for ANVC are discussed in detail, i.e. the FXLMS method, the IFT method and the MF-FDT method.

FXLMS is an adaptive filtering based control method in time domain that requires nominal model of the secondary path and is mainly used to tune feed-forward controllers. IFT method is a self-tuning control method in the time domain that requires additional path to inject extra signal and extra experiments to 'produce' a gradient of the control criterion. MF-FDT is a model free self-tuning control method in the frequency domain that requires additional path and extra experiments in implementations.

Chapter 3

GRADIENT ESTIMATE IN THE FREQUENCY DOMAIN

This Chapter proposes a new gradient estimation theory to tune both feedback controllers and feed-forward controllers. The method is completely based on the analysis of system dynamics and signals in the frequency domain. Considering its representation in the frequency domain, the approach is especially suitable for control problems with finite-spectrum signals, such as ANVC with periodic disturbances.

There are three parts in the Chapter. First a general framework of ANVC is represented with a generalized dynamical model with both feedback and feed-forward controllers and the control problem is represented as a mathematical optimization problem to minimize average quadratic cost functions in the time domain. Secondly, a new gradient estimation theory is stated in the frequency domain through analyzing the spectrum of signals and frequency responses of system dynamics. Finally, comparisons with other gradient estimation methods are also presented.

3.1 Problem Setting of ANVC

This section presents a general framework for the ANVC problems addressed in this thesis. Fundamental equations and performance functions are defined and the essence of self-tuning control is formulated in the time domain. The proposed framework can be extended to some general control problems as discussed in the end of this section.

3.1.1 Control model and mathematical description about ANVC

Consider a SISO LTI ANVC system with feedback and feed-forward controllers, Fig. 3.1 gives a schematic description.

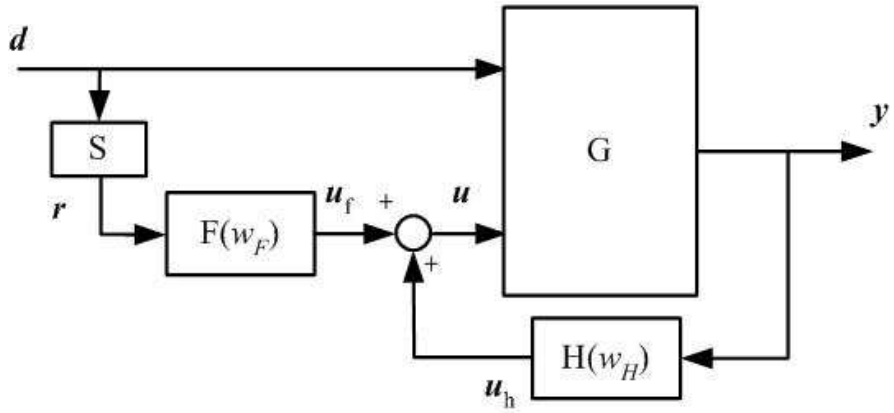

FIGURE 3.1: Block diagram of an ANVC system with feedback and feed-forward controllers

The measured output, which is affected by the disturbance $\boldsymbol{d} \in \mathbb{R}^{n_d}$, is represented by $\boldsymbol{y} \in \mathbb{R}^{n_y}$. G is the unknown plant dynamics with inputs $\boldsymbol{d}$ and $\boldsymbol{u} \in \mathbb{R}^{n_u}$, and produces output $\boldsymbol{y}$. It can be described as

$$\boldsymbol{y} = G(\boldsymbol{d}, \boldsymbol{u}) \tag{3.1}$$

The control signals from the feed-forward controller F and feedback controller H are denoted by $\boldsymbol{u}_f \in \mathbb{R}^{n_u}$ and $\boldsymbol{u}_h \in \mathbb{R}^{n_u}$, respectively. The tunable control system C comprises the parameterized feed-forward controller F and the feedback controller H:

$$\begin{aligned} C(\boldsymbol{w}, \boldsymbol{r}, \boldsymbol{y}): \qquad F: \boldsymbol{u}_f &= F(\boldsymbol{w}_F, \boldsymbol{r}) \\ H: \boldsymbol{u}_h &= H(\boldsymbol{w}_H, \boldsymbol{y}) \\ \boldsymbol{u} &= \boldsymbol{u}_f + \boldsymbol{u}_h \end{aligned} \tag{3.2}$$

which can be tuned by adjusting their parameter vectors in $\boldsymbol{w} := \{\boldsymbol{w}_F, \boldsymbol{w}_H\} \in \mathbb{R}^{n_w}$.

It is assumed that the disturbance-detection (also called "reference" in the literature) signal $\boldsymbol{r} \in \mathbb{R}^{n_r}$ is obtained through an unknown dynamics S from $\boldsymbol{d}$. While the output signal $\boldsymbol{y}(t)$ is measurable and recordable, the disturbance signal $\boldsymbol{d}$ cannot be measured directly.

If the system has steady output $\boldsymbol{y} = \{y(0), \ldots, y(N-1)\}$ with length of N sampling periods then the control performance criterion is defined as the average quadratic performance of a length N output sequence:

$$J(\boldsymbol{w}) := \frac{1}{N} \sum_{n=0}^{N-1} q(n) y^2(n), \tag{3.3}$$

where $q(n)$ is a series of a priori known weight factors varying with the index n.

Eqn.(3.3) can be expressed in the vector format as

$$J(\boldsymbol{w}) := \frac{1}{N}\boldsymbol{y}^{\mathrm{T}}(t)Q\boldsymbol{y}(t), \tag{3.4}$$

where $Q := \mathrm{diag}[q(0), \ldots, q(N-1)]$ is the a priori known weight matrix.

The objective of controller tuning is to tune the controller parameters $\boldsymbol{w}_F$ and $\boldsymbol{w}_H$ to minimize the performance criterion (3.3). The optimization problem can be formulated as

$$\begin{aligned} \min: \quad & J(\boldsymbol{w}) \text{ in (3.3)}, \\ s.t. \quad & \begin{cases} \text{Eqn. (3.1)}, \\ \text{Eqn. (3.2)}. \end{cases} \end{aligned} \tag{3.5}$$

Without losing generality, the dynamics represented by (3.1) and (3.2) are always assumed to contain differentiable functions. Hence, one of the effective methods to solve such optimization problem as (3.5) is by the steepest descend method which is based on the gradient estimate $\nabla J(\boldsymbol{w})$.

For example using Newton's method [10], the steepest descend update strategy is

$$\boldsymbol{w}_{k+1} = \boldsymbol{w}_k - \mu \nabla J(\boldsymbol{w}_k), \tag{3.6}$$

where $\boldsymbol{w}_k$, $\boldsymbol{w}_{k+1}$ is the controller parameter vector in the kth and $(k+1)$-th control iteration respectively, and a proper positive real value μ is the step size.

In general the problem of minimizing $J(\boldsymbol{w}_F, \boldsymbol{w}_H)$ is not necessarily convex. The tuning method only finds a suboptimal solution at a local minimum. This suboptimal solution of the problem is given as a solution of

$$\nabla J(\boldsymbol{w}_F^o, \boldsymbol{w}_H^o) = \boldsymbol{0} \tag{3.7}$$

There is and abundance of literature available on global optimization [32, 104, 106, 110]. To summarize, under the above proposed framework the key task to is to find out the gradient of ∇J in order to solve the ANVC problem.

3.1.2 Extensions to general control problems

Although the above framework is based on ANVC problems, it can be generalized to most control problems without difficulties.

Considering servo control problems, the block diagram of a typical servo system is shown in Fig. 3.2.

Compared with the feedback system described in Fig. 3.1, Fig. 3.2 has only slight differences in that the unknown disturbance, $\boldsymbol{d}$, in Fig. 3.1 is substituted with desired track reference, $\boldsymbol{r}$, in

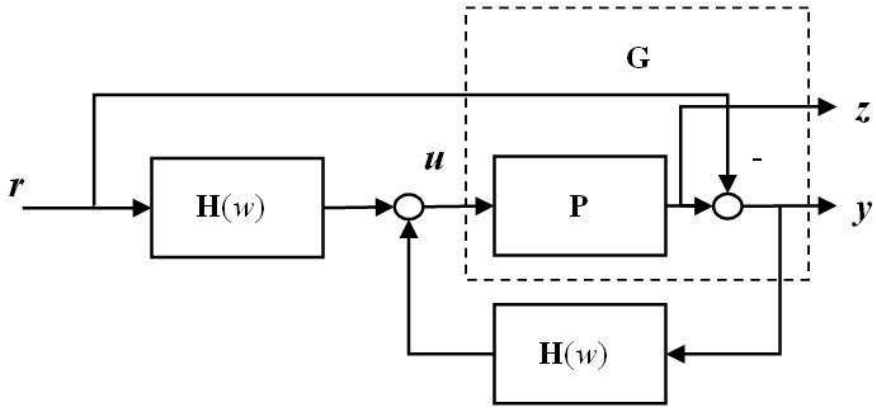

FIGURE 3.2: Block diagram of a servo control system with feedback controller

Fig. 3.2 and there are real output $\boldsymbol{z}$ and error signal $\boldsymbol{y}$ presented as outputs. The mathematical model of the servo control problem can be represented in terms of (3.5) as well.

3.2 Gradient Estimate in the Frequency Domain

As mentioned above, the gradient estimate is the key step in adaptive control. In this section, the above mentioned general framework for ANVC in the time domain is described from the aspect of the frequency domain, and a new gradient estimation technique is proposed as based on representations completely in the frequency domain.

3.2.1 Descriptions in the frequency domain

In this subsection some preliminary assumptions are presented about the dynamics considered in the frequency domain. Some notation is also defined and some equations are set up for further discussion.

Consider a SISO discrete system as described by Fig. 3.1, there is an N-length output data set $\boldsymbol{y} := \{y(0); \ldots; y(N-1)\} \in \mathrm{R}^N$. Let $\omega_m := \frac{2\pi m}{N}, m = 0, \ldots, N-1$ denote m-th discrete frequency for an N-length time sequence. $\boldsymbol{\phi}_y := \{\phi_y(\omega_0); \ldots; \phi_y(\omega_{N-1})\} \in \mathbb{C}^N$ is the discrete spectrum of the N-length time sequence $\boldsymbol{y}$, i.e. $\boldsymbol{\phi}_y \doteq \mathrm{DFT}(\boldsymbol{y})$. There are similar notations of $\boldsymbol{\phi}_d$, $\boldsymbol{\phi}_r$, $\boldsymbol{\phi}_{uf}$ and $\boldsymbol{\phi}_{uh}$.

In the frequency domain the plant G can be described with Frequency Response Functions (FRF) $\{\boldsymbol{\phi}_d, \boldsymbol{\phi}_u\} \in \mathbb{C}^{2N} \mapsto \boldsymbol{\phi}_y \in \mathbb{C}^N$ as:

$$\boldsymbol{\phi}_y = \Phi_G(\boldsymbol{\phi}_d, \boldsymbol{\phi}_u) \tag{3.8}$$

and the controller system C is described as function of $\{\boldsymbol{w}, \boldsymbol{\phi}_r, \boldsymbol{\phi}_y\} \in \mathbb{C}^{n_w+2N} \mapsto \boldsymbol{\phi}_u \in \mathbb{C}^N$:

$$\begin{aligned} \Phi_C(\boldsymbol{w}, \boldsymbol{\phi}_r, \boldsymbol{\phi}_y): \qquad \Phi_F: \boldsymbol{\phi}_{uf} &= \Phi_F(\boldsymbol{w}_F, \boldsymbol{\phi}_r) \\ \Phi_H: \boldsymbol{\phi}_{uh} &= \Phi_H(\boldsymbol{w}_H, \boldsymbol{\phi}_y) \\ \boldsymbol{\phi}_u &= \boldsymbol{\phi}_{uf} + \boldsymbol{\phi}_{uh} \end{aligned} \tag{3.9}$$

It is assumed that Φ_G, Φ_F and Φ_H are differentiable functions with respect to the spectra of input arguments, i.e., $\{\phi_y, \phi_r\}$ and the tunable parameters, i.e., $\{\boldsymbol{w}_H, \boldsymbol{w}_F\}$.

Considering differentiable functions, local linearization can be be performed in case of infinitesimal increments. For the ANVC system as in Fig. 3.1, the local linearization of Φ_G can be described as:

$$\Delta\boldsymbol{\phi}_y \approx \frac{\mathrm{d}\boldsymbol{\phi}_y}{\mathrm{d}\{\boldsymbol{\phi}_d, \boldsymbol{\phi}_u\}} \begin{bmatrix} \Delta\boldsymbol{\phi}_d \\ \Delta\boldsymbol{\phi}_u \end{bmatrix} \tag{3.10}$$

Since $\boldsymbol{\phi}_d$ can be considered as fixed for the stationary disturbance assumptions, only the case that $\Delta\boldsymbol{\phi}_d = \mathbf{0}$ is needed to be discussed.

We introduce now notations as

$$\Phi_{G'} := \Phi_{Gu'} := \frac{\partial \Phi_G(\boldsymbol{\phi}_d, \boldsymbol{\phi}_u)}{\partial \boldsymbol{\phi}_u} \quad \in \mathbb{C}^{N\times N} \tag{3.11}$$

and

$$\Phi_{H'} := \Phi_{Hu'} := \frac{\partial \Phi_H(\boldsymbol{w}_H, \boldsymbol{\phi}_{uh})}{\partial \boldsymbol{\phi}_{uh}} \quad \in \mathbb{C}^{N\times N} \tag{3.12}$$

which $\Phi_{G'}$ and $\Phi_{H'}$ are used more frequently in order to simplify expression if no confusion.

At the same time the derivative matrixes with respect to parameter vectors $\{\boldsymbol{w}_F, \boldsymbol{w}_H\}$ can be denoted by

$$\Phi_{H'_w} := \frac{\partial \Phi_H(\boldsymbol{w}_H, \boldsymbol{\phi}_{uh})}{\partial \boldsymbol{w}_H} \in \mathbb{C}^{N\times n_{w_H}} \tag{3.13}$$

and

$$\Phi_{F'_w} := \frac{\partial \Phi_F(\boldsymbol{w}_F, \boldsymbol{\phi}_{uf})}{\partial \boldsymbol{w}_F} \in \mathbb{C}^{N\times n_{w_F}} \tag{3.14}$$

The infinitesimal increments of plant dynamics G in the frequency domain with respect to $\Delta\boldsymbol{\phi}_u$ can be written as

$$\Delta\phi_y \approx \Phi_{G'}(\Delta\phi_{uf} + \Delta\phi_{uh}) \tag{3.15}$$

3.2.2 Gradient estimate in the frequency domain

In order to estimate the gradient of the performance criterion J with respect to the control parameter vector $\boldsymbol{w}$, the technique of perturbation analysis is adopted to find out the relationship from the change of controller parameters to the change of output spectra.

Considering small updates of the parameters, i.e., $\boldsymbol{w}$, i.e., $\boldsymbol{w}_F \rightarrow \boldsymbol{w}_F + \Delta\boldsymbol{w}_F$ and $\boldsymbol{w}_H \rightarrow \boldsymbol{w}_H + \Delta\boldsymbol{w}_H$, small increments of the control action in the feed-forward path $\Delta\phi_{uf}$ are obtained as

$$\Delta\phi_{uf} = \Phi_{F'_w}\Delta\boldsymbol{w}_F. \tag{3.16}$$

There are two changing variables in the feedback path when tuning feedback controllers: the change of output spectrum $\Delta\phi_y$ and the change of parameters $\Delta\boldsymbol{w}_H$, which leads to

$$\Delta\phi_{uh} = \Phi_{H'}\Delta\phi_y + \Phi_{H'_w}\Delta\boldsymbol{w}_H. \tag{3.17}$$

According to (3.15), it is straightforward to write the increment equation as

$$\Delta\phi_y = \Phi_{G'}(\frac{\partial\Phi_F(\phi_r, \boldsymbol{w}_F)}{\partial\boldsymbol{w}_F}\Delta\boldsymbol{w}_F + \frac{\partial\Phi_H(\phi_y, \boldsymbol{w}_H)}{\partial\boldsymbol{w}_H}\Delta\boldsymbol{w}_H + \Phi_{H'}\Delta\phi_y) \tag{3.18}$$

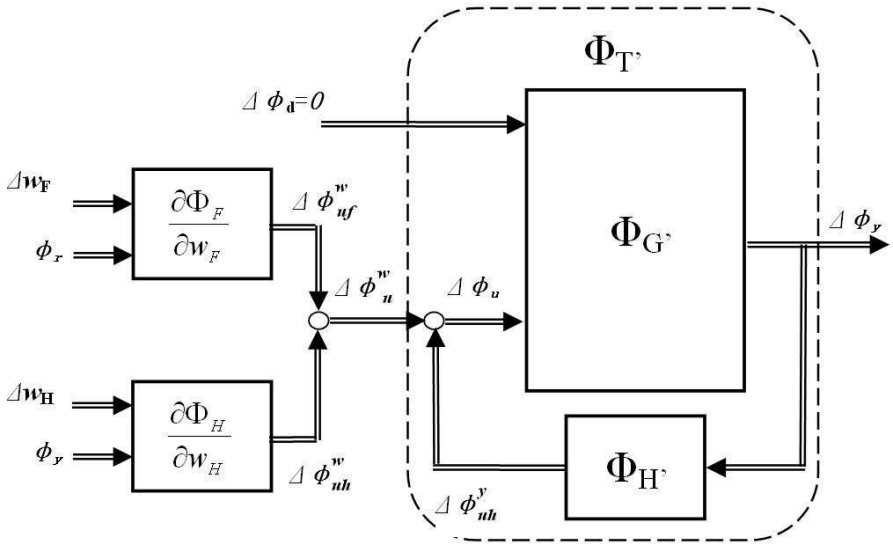

FIGURE 3.3: Block diagram of infinitesimal increment in frequency domain

Denoting

$$\Delta\boldsymbol{\phi}_{uh}^{w} := \frac{\partial \Phi_H(\boldsymbol{\phi}_y, \boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}\Delta\boldsymbol{w}_H \tag{3.19}$$

$$\Delta\boldsymbol{\phi}_{uh}^{y} := \Phi_{H'}\Delta\boldsymbol{\phi}_y \tag{3.20}$$

$$\Delta\boldsymbol{\phi}_{u}^{w} := \Delta\boldsymbol{\phi}_{uh}^{w} + \Delta\boldsymbol{\phi}_{uf} \tag{3.21}$$

$$\tag{3.22}$$

Equation (3.18) can be illustrated in Fig 3.3.

As illustrated in (3.18), the dynamics G is considered as an unknown control object and the input/output increment mapping is $\Delta\boldsymbol{\phi}_u \mapsto \Delta\boldsymbol{\phi}_y$.

Note that the physical infinitesimal increment in the path of feedback controller H comprises two parts:

- $\boldsymbol{\phi}_{uh}^{w}$ caused by the change of controller parameter $\Delta\boldsymbol{w}_H$;
- $\boldsymbol{\phi}_{uh}^{y}$ caused by the change of output $\Delta\boldsymbol{\phi}_y$.

If $(I - \Phi_{G'}\Phi_{H'})^{-1}$ exists, the input/output mapping $\{\Delta\boldsymbol{\phi}_{uf}^{w} + \Delta\boldsymbol{\phi}_{uh}^{w}\} \mapsto \Delta\boldsymbol{\phi}_y$ can be rewritten from (3.18) as

$$\Delta\boldsymbol{\phi}_y = (I - \Phi_{G'}\Phi_{H'})^{-1}\Phi_{G'}(\Delta\boldsymbol{\phi}_{uf}^{w} + \Delta\boldsymbol{\phi}_{uh}^{w}) \tag{3.23}$$

Considering the closed-loop system $T := \{G, H\}$ as the unknown plant to be controlled, $(I - \Phi_{G'}\Phi_{H'})^{-1}\Phi_{G'}$ is the partial derivative matrix of Φ_T with respect of the change of the spectrum of input signals.

Introducing the notation

$$\Phi_{T'} := (I - \Phi_{G'}\Phi_{H'})^{-1}\Phi_{G'} \tag{3.24}$$

and using (3.23), the partial derivative of $\boldsymbol{\phi}_y$ with respect to controller parameters $\boldsymbol{w}_H$ and $\boldsymbol{w}_F$ can be written as

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_H} = \Phi_{T'}\frac{\partial \Phi_H(\boldsymbol{\phi}_y, \boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} \tag{3.25}$$

and

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_F} = \Phi_{T'}\frac{\partial \Phi_F(\boldsymbol{\phi}_r, \boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} \tag{3.26}$$

In (3.25) and (3.26), controllers H and F are known by designers. Therefore, the key item of gradient estimation in the frequency domain is to estimate the closed-loop dynamics $\Phi_{T'}$.

3.2.3 Gradient of the performance criterion

After getting the gradient of the output spectrum, the gradient of performance criterion with respect to parameters is easy to obtain if the performance can be represented in terms of the spectra.

Considering the average quadratic performance (3.3), according to Parseval's Theorem [103], it is straightforward to write the frequency domain format as

$$J = \frac{1}{N}\sum_{i=0}^{N-1}\phi_y^*(\omega_i)\phi_Q(\omega_i)\phi_y(\omega_i) = \frac{1}{N^2}\boldsymbol{\phi}_y^*\Phi_Q\boldsymbol{\phi}_y \tag{3.27}$$

where $\phi_Q(\omega_i)$ and Φ_Q represent the element and matrix format in the frequency domain of the weighting matrix Q in (3.4).

The derivative of performance J with respect to the controller parameters can be written in the frequency domain format as

$$\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_i} = \frac{2}{N^2}\boldsymbol{\phi}_y^*\Phi_Q\Phi_{T'}\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i} \tag{3.28}$$

While the output $\boldsymbol{y}$, the controller H and F are all known, $\boldsymbol{\phi}_y$ and $\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}$ are both available in (3.28). The key to gradient estimation turns out to be to find an estimate of $\Phi_{T'}$.

Remark 3.2.1. As mentioned in the beginning of this subsection, the criterion function $J(w)$ is not limited to the average quadratic cost function. The gradient of the generalized performance $J(\boldsymbol{w})$, including $\boldsymbol{u}$ too, can be also obtained by following the proposed approach.

Consider the cost function $J_G(\boldsymbol{w})$ in generalized minimum variance control [161] as

$$J_G(\boldsymbol{w}) = \boldsymbol{y}^{\mathrm{T}}Q\boldsymbol{y} + \boldsymbol{u}^{\mathrm{T}}R\boldsymbol{u} \tag{3.29}$$

where $\boldsymbol{y}$ and $\boldsymbol{u}$ are finite length measured output and control vectors, respectively, and Q and R are a priori known weighting matrixes.

According to Parseval's Theorem [103], (3.29) can be represented in frequency domain as

$$J_G(\boldsymbol{w}) = \frac{1}{N^2}(\boldsymbol{\phi}_y^*\Phi_Q\boldsymbol{\phi}_y + \boldsymbol{\phi}_u^*\Phi_R\boldsymbol{\phi}_u) \tag{3.30}$$

where Φ_Q and Φ_R represent the weighting matrices Q and R in (3.29), respectively.

Similar to the previous deduction process using (3.25) and (3.26), it is straightforward to obtain the gradient of the generalized criterion $J_G(\boldsymbol{w})$ in the frequency domain as follows:

$$\frac{\partial J_G(\boldsymbol{w})}{\partial \boldsymbol{w}_i} = \frac{2}{N^2}(\boldsymbol{\phi}_y^* \Phi_Q \Phi_{T'} + \boldsymbol{\phi}_u^* \Phi_R) \frac{\partial \Phi_C(\boldsymbol{w}, \boldsymbol{\phi}_y, \boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i} \tag{3.31}$$

Remark 3.2.2. In order to ease the description of the gradient estimate in the frequency domain, SISO systems are assumed here in a preliminary analysis. The above discussion is based on the increment equation (3.15) and the methods and conclusions can be easily extended to multi-input/multi-output (MIMO) and nonlinear systems when they can represent with such increment format as (3.15).

Since $\Phi_{G'}$ is defined as the partial derivative in the above discussion, in the MIMO case the gradient with respect to the control action in one control channel can be considered as the partial derivative with respect to control action for one channel.

There is no plant linearity limitation with the discussion so far in this subsection, the method and conclusions can be applied to nonlinear systems. While $\Phi_{G'}$ is a complex diagonal squared matrix in the linear case, it can be considered as complex squared matrix with elements in non-diagonal positions.

A detailed discussion about MIMO and nonlinear systems will be given in Chapter 5 and Chapter 6, respectively.

3.2.4 Extended analysis with periodic signals

In the previous subsections conclusions were drawn without any limitation about the disturbance and reference signals. However, in general, a finite time series signal has infinite spectrum in the frequency domain. While the acquired signals are always finite in the time domain in control engineering, their spectra will be infinite practically almost always in the frequency domain. This implies that the spectrum vectors $\boldsymbol{\phi}_y$ and $\boldsymbol{\phi}_r$, the matrix $\Phi_{T'}$ and $\frac{\partial \Phi_C(\boldsymbol{w}, \boldsymbol{\phi}_y, \boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}$ are all infinite-dimensional, which is impractical in real applications.

Fortunately, the above limitation can be avoided in the case that the signal has finite frequency spectrum for a periodic signal or when finite spectrum approximations are feasible.

Considering (3.28) in the general case, $\boldsymbol{\phi}_y^*$ can be considered as a weighting vector where the conjugate complex value of a single frequency spectrum, $\phi_y^*(\omega)$ is the weighting factor of $\frac{\partial \phi_y(\omega)}{\partial \boldsymbol{w}}$. It can be expressed as a sum

$$\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_i} = \frac{2}{N^2} \sum_{\omega \in (-\infty, \infty)} \phi_y^*(\omega) Q(i,i) \frac{\partial \phi_y(\omega)}{\partial \boldsymbol{w}_i} \tag{3.32}$$

where $\frac{\partial \phi_y(\omega)}{\partial \boldsymbol{w}_i} := \Phi_{T'}(\phi_y(\omega), \boldsymbol{\phi}_u) \frac{\partial \Phi_C(\boldsymbol{w}, \boldsymbol{\phi}_u, \boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}$.

In (3.32), note that if $\phi_y(\omega_i) = 0$ and $\frac{\partial \phi_y(\omega_i)}{\partial \boldsymbol{w}_i} < \infty$ then $\phi_y^*(\omega_i)\frac{\partial \phi_y(\omega_i)}{\partial \boldsymbol{w}_i} = 0$. In the matrix format of (3.28), if i-th element in $\boldsymbol{\phi}_y$ is zero and $||\Phi_{T'}\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}||_\infty < \infty$, then the elements in the i-th row in matrix $\Phi_{T'}\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}$ will not affect the result of $\nabla J(\boldsymbol{w})$.

To calculate the gradient of the performance criterion $\nabla J(\boldsymbol{w})$ for a stable system, it is only necessary to consider the system's frequency response with respect to the frequencies which are contained in the output spectra.

Assume that a periodic output $\boldsymbol{y}$ has a spectrum $\boldsymbol{\phi}_y$ with finite none-zero values. The none-zero value frequencies compose a n_Ω-size finite frequency set $\Omega := \{\omega_0, \ldots, \omega_{n_\Omega - 1}\}$,$n_\Omega < \infty$, which are denotated by $\boldsymbol{\phi}_y|_\Omega := \{\phi_y(\omega_0), \ldots, \phi_y(\omega_{n_\Omega - 1})\}$. And all the other elements in $\boldsymbol{\phi}_y$ are 0, i.e., $\phi_y(\omega) = 0, \forall \omega \neq \Omega$.

In the linear case the $[\Phi_{T'}\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}]$ for a finite frequency, the set Ω, can be represented as $\Phi_{T'}|_\Omega \frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}|_\Omega$, where $\Phi_{T'}|_\Omega$ and $\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}|_\Omega$ denote the frequency response with respect to the finite frequency set Ω.

As a consequence, in order to get $\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_i}$ in (3.28), only $\boldsymbol{\phi}_y|_\Omega$, $\Phi_{T'}|_\Omega$ and $\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}|_\Omega$ is required to to obtain $\nabla J(\boldsymbol{w})$. It is straightforward to rewrite (3.28) to the format with respect to the finite frequency set Ω as

$$\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_i} = \frac{2}{N}\boldsymbol{\phi}_y^*|_\Omega \Phi_{T'}|_\Omega \frac{\partial \Phi_C(\boldsymbol{\phi}_y, \boldsymbol{\phi}_r, \boldsymbol{w})}{\partial \boldsymbol{w}_i}|_\Omega \tag{3.33}$$

While the gradient estimate in the frequency domain has the limitation of finite spectrum which limits applicability in some control problems, using (3.33), in the case of periodic signals, the gradient estimate in the frequency domain can be much more effective than known methods in time domain.

Consider now a control problem with period length N for signals in the time domain and with finite frequency spectrum Ω in the frequency domain. While the problem in the time domain is to solve N sub-problems for $\frac{\partial y(t)}{\partial \boldsymbol{w}}, t = 0, \ldots, N-1$, the problem in the frequency domain is to solve n_Ω sub-problems of estimation of $\frac{\partial \phi_y|_{\omega_i}}{\partial \boldsymbol{w}}, i = 0, \ldots, n_\Omega - 1$. For instance, as shown in Fig. 3.4, a length N time series signal requires to process N data to compute the gradient in the time domain. When the signal contains only 3 frequencies, it requires to process only 3 spectral data to compute the gradient in the frequency domain.

Therefore if $n_\Omega << N$ the gradient estimate for control can be greatly simplified in view of (3.33). Furthermore, since $\boldsymbol{\phi}_y$ can be considered as weighting factors as shown in (3.32), if the spectrum of the signal $\boldsymbol{y}$ has some dominant frequencies Ω, i.e., $\boldsymbol{\phi}_y|_\Omega \approx \boldsymbol{\phi}_y$, then it is straightforward to rewrite (3.33) as

$$\frac{\partial J(\boldsymbol{w})}{\partial \boldsymbol{w}_i} \approx \frac{2}{N}\boldsymbol{\phi}_y^*|_\Omega \Phi_{T'}|_\Omega \frac{\partial \Phi_C(\boldsymbol{\phi}_y, \boldsymbol{\phi}_r, \boldsymbol{w})}{\partial \boldsymbol{w}_i}|_\Omega, \text{ if } \boldsymbol{\phi}_y|_\Omega \approx \boldsymbol{\phi}_y \tag{3.34}$$

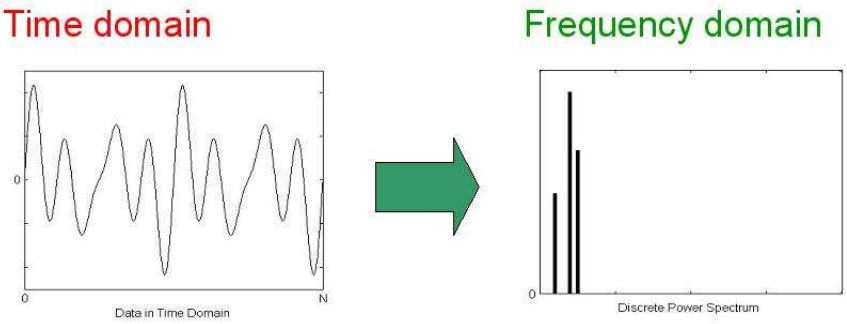

FIGURE 3.4: Proceeding in the time domain and frequency domain

Eqn. (3.33) represents a new approach to gradient estimates for control problems with periodic disturbances. Since periodic signals are very common in ANVC, the proposed gradient estimate method is especially suitable to solve ANVC problems. At the same time Eqn. (3.34) can also be used to deal with the disturbance containing some noise.

Remark 3.2.3. Although the primary application field is ANVC with periodic signals, the proposed gradient estimation techniques can be extended to some common control problems since the previous discussion is based on the general framework described with the Fig. 3.1 and the related mathematical model (3.5). Such a problem is the servo control problem illustrated by Fig. 3.2. The gradient estimate in the frequency domain for the common control problems is straightforward. Since most signals can be considered to have finite dominant frequencies, Eqn. (3.34) can be practical to solve the most of common control problems in applications.

3.3 Comparison with Other Gradient Estimate Techniques

As mentioned above, the gradient estimate in the frequency domain is based on a general idea of how to compute gradient in adaptive control problems. In this section, the proposed gradient estimation technique is compared with the gradient estimation approach taken in some other popular control methods in ANVC. This will illustrates FD-IT's similar representation in the frequency domain that helps to gain insight into other related IFT control methods.

In this section, the comparisons are based on the principle of gradient computation in control. The comparison about detail algorithms to implement control will be discussed in the next Chapter.

3.3.1 Reformation in LTI ANVC

Before making comparisons, the general gradient estimate in the frequency domain, i.e., (3.25), (3.26) and (3.28), is specially studied in the LTI system.

Considering the ANVC system shown in Fig. 3.1, the plant G and controllers H and F are all assumed to be LTI systems.

For LTI controllers we can write $\frac{\partial \Phi_H(\boldsymbol{\phi}_y, \boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}$ and $\frac{\partial \Phi_F(\boldsymbol{\phi}_y, \boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}$ as follows:

$$\Phi_{H'_w} = \frac{\partial \Phi_H(\boldsymbol{\phi}_y, \boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} = \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}\boldsymbol{\phi}_y \tag{3.35}$$

and

$$\Phi_{F'_w} = \frac{\partial \Phi_F(\boldsymbol{\phi}_r, \boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} = \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}\boldsymbol{\phi}_r \tag{3.36}$$

In the linear case there is $\Phi_{T'} = \Phi_T$ and $\Phi_{T'}$ in FD-IT is substituted as Φ_T in the following discussion in this section. It is straightforward to rewrite (3.25) and (3.26) as

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_H} = \Phi_T \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}\boldsymbol{\phi}_y \tag{3.37}$$

and

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_F} = \Phi_T \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}\boldsymbol{\phi}_r \tag{3.38}$$

Due to linearity Φ_T, $\frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}$ and $\frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}$ are all diagonal square matrixes and this leads to

$$\Phi_T \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} = \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}\Phi_T, \tag{3.39}$$

and

$$\Phi_T \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} = \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}\Phi_T \tag{3.40}$$

Therefore, (3.37) and (3.38) can be written as

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_H} = \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}\Phi_T\boldsymbol{\phi}_y \tag{3.41}$$

and

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_F} = \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}\Phi_T\boldsymbol{\phi}_r \tag{3.42}$$

Using (3.41) and (3.42), $\nabla J(\boldsymbol{w})$ in LTI ANVC can be expressed as

$$\frac{\partial J(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} = \frac{2}{N}\boldsymbol{\phi}_y^* \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}\Phi_T \boldsymbol{\phi}_y \tag{3.43}$$

and

$$\frac{\partial J(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} = \frac{2}{N}\boldsymbol{\phi}_y^* \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F}\Phi_T \boldsymbol{\phi}_r \tag{3.44}$$

3.3.2 Use of dynamical derivatives in the LTI case

The new gradient estimation method has been deduced completely in the frequency domain. In order to generalize the proposed method to common control problems, the incremental format equations as (3.15) and (3.18) are used because nonlinearities can be locally linearized with infinitesimal incremental equations.

Actually in the LTI case the proposed gradient estimates can be directly obtained through the derivation of the integrated system dynamics.

Considering a typical feedback and feed-forward hybrid control system as in Fig. 3.5, the plant G is a LTI system, the LTI feedback controller is $H(\boldsymbol{w}_H)$ and the feed-forward controller is $F(\boldsymbol{w}_F)$ that both have tunable parameters $\boldsymbol{w}_H$ and $\boldsymbol{w}_F$.

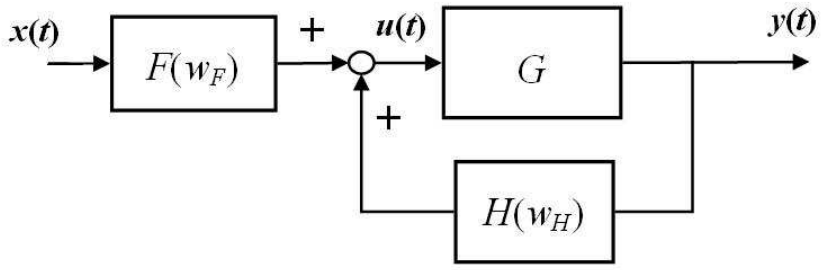

FIGURE 3.5: Diagram of feedback and feed-forward control system

As it is well known the whole dynamics can be expressed in the time domain as

$$\boldsymbol{y} = \frac{G(z^{-1})F(z^{-1}, \boldsymbol{w}_F)}{1 - G(z^{-1})H(z^{-1}, \boldsymbol{w}_H)}\boldsymbol{x} \tag{3.45}$$

which can be expressed in the frequency domain [24] as

$$\boldsymbol{\phi}_y = [I - \Phi_G \Phi_H(\boldsymbol{w}_H)]^{-1}\Phi_G \Phi_F(\boldsymbol{w}_F)\boldsymbol{\phi}_x \tag{3.46}$$

As mentioned in the above subsection, Φ_G, $\Phi_H(\boldsymbol{w}_H)$, $\Phi_F(\boldsymbol{w}_F)$ and $[I - \Phi_G\Phi_H(\boldsymbol{w}_H)]$ are all diagonal matrixes in the LTI case.

According to (3.46), it is straightforward to obtain

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_F} = [I - \Phi_G \Phi_H]^{-1} \Phi_G \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} \boldsymbol{\phi}_x = \Phi_T \frac{\partial \Phi_F(\boldsymbol{w}_F)}{\partial \boldsymbol{w}_F} \boldsymbol{\phi}_x \tag{3.47}$$

which is the same as (3.38).

According to (3.46), the derivative $\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_H}$ is first given as

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_H} = [I - \Phi_G \Phi_H]^{-2} \Phi_G \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} \Phi_G \Phi_F \boldsymbol{\phi}_x \tag{3.48}$$

In LTI systems $[I - \Phi_G \Phi_H]^{-1}$, Φ_G and $\frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}$ are all diagonal matrixes, and their positions are exchangeable in (3.48), which gives

$$\begin{aligned} & [I - \Phi_G \Phi_H]^{-2} \Phi_G \tfrac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} \Phi_G \Phi_F \boldsymbol{\phi}_x \\ = \; & \{[I - \Phi_G \Phi_H]^{-1} \Phi_G\} \{\tfrac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H}\} \{[I - \Phi_G \Phi_H]^{-1} \Phi_G \Phi_F \boldsymbol{\phi}_x\} \\ = \; & \Phi_T \tfrac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} \boldsymbol{\phi}_y \end{aligned} \tag{3.49}$$

Considering three braced items in (3.49), according to the denotation of the closed loop dynamics T (3.24) in LTI case and system output spectrum in (3.46), it is straightforward to write

$$\frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{w}_H} = \Phi_T \frac{\partial \Phi_H(\boldsymbol{w}_H)}{\partial \boldsymbol{w}_H} \boldsymbol{\phi}_y \tag{3.50}$$

which is same as (3.37).

Therefore in LTI case the new proposed gradient theory can be well explained by the direct deduction through the derivation of system dynamics.

3.3.3 Comparison with gradient estimate in the time domain

In this subsection, the computations of criterion gradient in two popular tuning methods in the time domain, including: FXLMS and TD-IFT, are compared with the proposed gradient estimate, which is now abbreviated as FD-GE for the rest of this discussion.

First we consider the original FXLMS method in which the performance criterion gradient can be expressed as

$$J(\boldsymbol{w}) = 2 \sum_{t=0}^{N-1} y(t) (\nabla F(\boldsymbol{w}) \boldsymbol{x}^f) \tag{3.51}$$

where $\boldsymbol{x}^f = G\boldsymbol{r}$.

Note that no feedback controller is used in the prototype of the FXLMS method, the closed-loop dynamics $T(G, H)$ is simplified to be secondary path dynamics G, which leads to $\Phi_T = \Phi_G$

in the frequency domain. Therefore, compared with (3.44), the terms in (3.51) have matching relationships as shown in Tab. 3.1:

FXLMS	FD-GE
$y(t)$	$\boldsymbol{\phi}_y^*$
∇F	$\frac{\partial \Phi_F(\boldsymbol{w}_F, \boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_F}$
$\boldsymbol{x}^f (= G\boldsymbol{r})$	$\Phi_T \boldsymbol{\phi}_r (= \Phi_G \boldsymbol{\phi}_r)$

TABLE 3.1: Comparison between gradient computations in FXLMS and in the frequency domain (FD-GE)

Considering the feedback tuning permutation of FXLMS and FELMS method, the computation of performance gradient is expressed as

$$J(\boldsymbol{w}) = 2 \sum_{t=0}^{N-1} y(t)(\nabla H(\boldsymbol{w}) \boldsymbol{e}^f) \tag{3.52}$$

where $\boldsymbol{e}^f = K\boldsymbol{r}$.

According to (2.20), K in FELMS is the closed-loop dynamics $T(G, H)$ in FD-GE. Therefore, compared with (3.43), the terms in (3.52) have matching relationship as shown in Tab. 3.2:

FELMS	FD-GE
$y(t)$	$\boldsymbol{\phi}_y^*$
∇H	$\frac{\partial \Phi_H(\boldsymbol{w}_H, \boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_H}$
$\boldsymbol{e}^f (= K\boldsymbol{r})$	$\hat{\Phi}_T \boldsymbol{\phi}_y (= \hat{\Phi}_T \boldsymbol{\phi}_y)$

TABLE 3.2: Comparison between gradient computations in feedback FELMS and in the frequency domain (FD-GE)

Secondly, considering TD-IFT as illustrated in Fig. 2.7 and Fig. 2.8, the computation of the performance criterion gradient is based on $\boldsymbol{z}_h$ and $\boldsymbol{z}_f$, which are the outputs of extra experiments by injecting the previously recorded output $\{y(-N), \ldots, y(-1)\}$ and repeated reference $\{r(-N), \ldots, r(-1)\}$, respectively. It is noted that z_h and z_f are both filtered by the closed-loop system, which can be presented as:

$$\boldsymbol{z}_h = \frac{G}{1 - GH} \boldsymbol{y}, \tag{3.53}$$

and

$$\boldsymbol{z}_f = \frac{G}{1 - GH} \boldsymbol{r}. \tag{3.54}$$

Therefore, compared with (3.43) and (3.44), the items in (2.25) and (2.26) have matching relationship as shown in Tab. 3.3.

TD-IFT	FD-GE
$y(t)$	$\boldsymbol{\phi}_y^*$
$\frac{\partial H(\boldsymbol{w}_H,\boldsymbol{y})}{\partial \boldsymbol{w}_H}$	$\frac{\partial \Phi_H(\boldsymbol{w}_H,\boldsymbol{\phi}_y)}{\partial \boldsymbol{w}_H}$
$\frac{\partial F(\boldsymbol{w}_F,\boldsymbol{r})}{\partial \boldsymbol{w}_F}$	$\frac{\partial \Phi_F(\boldsymbol{w}_F,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_F}$
$z_h(t)$	$\Phi_T \boldsymbol{\phi}_y$
$z_f(t)$	$\Phi_T \boldsymbol{\phi}_r$

TABLE 3.3: Comparison between gradient computations in TD-IFT and in the frequency domain (FD-GE)

3.3.4 Comparison with model-free frequency domain tuning methods

While there are only a few comprehensive tuning methods available in the literature for both feedback and feed-forward controllers for ANVC, the MF-FDT method can also be considered to be an iterative learning control (ILC) method in the frequency domain.

Therefore in MD-FDT the closed-loop dynamics is identical with the secondary path, i.e., $\Phi_T = \Phi_G$.

At the same time note that the tunable parameters in the model-free frequency domain tuning method are the ILC control actions $\boldsymbol{u}$ themselves, i.e. $\boldsymbol{w}_F = \boldsymbol{u}$ with

$$\frac{\partial H(\boldsymbol{u},\boldsymbol{y})}{\partial \boldsymbol{u}} \equiv \boldsymbol{I}. \tag{3.55}$$

Compared with (3.44), the items in (2.40) have matching relationship as shown in Tab. 3.3. Tab. 3.4.

MF-FDT	FD-GE
$[\boldsymbol{y}_i\, \boldsymbol{y}_r]_k^{\mathrm{T}}$	$\boldsymbol{\phi}_y^*$
$\boldsymbol{G}$	$\Phi_G (= \Phi_T)$
$\boldsymbol{I}$	$\frac{\partial H(\boldsymbol{u},\boldsymbol{y})}{\partial \boldsymbol{u}}$
$\boldsymbol{z}_k = \boldsymbol{G}[\boldsymbol{y}_i\, \boldsymbol{y}_r]_k^{\mathrm{T}}$	$\phi_y^* \Phi_T \frac{\partial H(\boldsymbol{u},\boldsymbol{y})}{\partial \boldsymbol{u}}$

TABLE 3.4: Comparison between gradient computations in the MF-FDT and in the frequency domain (FD-GE)

3.3.5 Chain rule in the gradient estimation procedure

According to the above comparisons made, the gradient estimate in the frequency domain can be used to explain the gradient computations in all the mentioned control methods. It gives a universal description about gradient estimate in the frequency domain.

Through the previous comparisons made between the each term in the equations for gradient computations, the computational procedures to for the performance gradient imply the "physical

essence" of the chain rule as shown in Fig. 3.6, i.e. an update of parameters causes a change in control action, the change of control action leads to change in output, and the change in output affects the final performance. As shown in Fig. 3.6, they have similar expressions in the time and frequency domain, i.e., $\frac{\partial C(\boldsymbol{w})}{\partial \boldsymbol{w}}$-$\frac{\partial \Phi_C(\boldsymbol{w})}{\partial \boldsymbol{w}}$, $\Delta \boldsymbol{u}$-$\Delta \boldsymbol{\phi}_u$, $\frac{G}{1-GH}$-$\frac{\Phi_G}{I-\Phi_G\Phi_H}$, $\Delta \boldsymbol{y}$-$\Delta \boldsymbol{\phi}_y$.

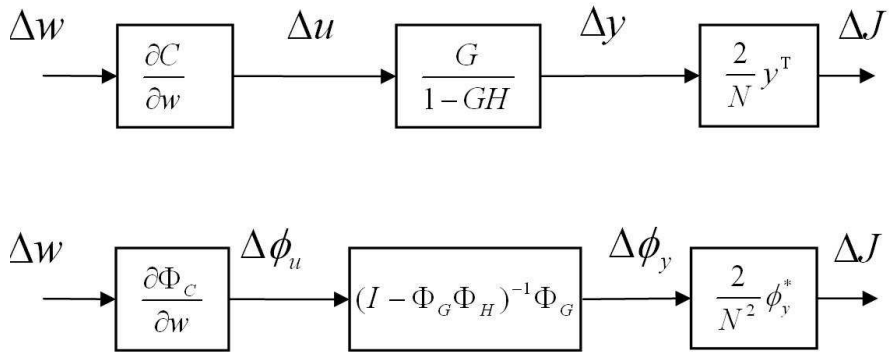

FIGURE 3.6: Chain rule of gradient estimate in the time and frequency domain

In the chain shown in Fig. 3.6, the most intricate and key link is the mapping from the control action $\boldsymbol{u}$ to the output $\boldsymbol{y}$, i.e., the closed-loop dynamics T.

Among the above mentioned adaptive control methods, TD-IFT and the proposed gradient estimate in the frequency domain gives a comprehensive presentation of the gradient estimation procedure which both explains the mapping of closed-loop dynamics and offers a control method that suits both feedback controllers and feed-forward controllers. While TD-IFT provides a way to perform gradient estimation in the time domain, FD-IFT can be considered as an equivalent expression in the frequency domain.

Despite the similarity of the gradient computations, the tuning algorithms and implementations of these adaptive control methods can be quite different. The difference between the new proposed method and existing methods is mainly due to different operations performed in the time and frequency domains.

3.4 Summary

This Chapter proposed a new gradient estimation theory completely performed in the frequency domain. The core is the gradient of the output spectrum $\boldsymbol{\phi}_y$ with respect to the controller parameters $\boldsymbol{w}$ in a feedback and feed-forward hybrid system. Although the presentation in this Chapter is mainly based on ANVC problem as in Fig.3.1 and on an average quadratic criterion as (3.29), the proposed gradient estimation method can also be used to attack common control problems and general performance criteria.

An advantage is that the proposed approach can greatly simplify gradient computation when the signal and system have much simpler presentation in the frequency domain than in the time

domain.

The gradient estimation method in the frequency domain has been discussed using equation (3.28) that has only one unknown item that is the closed-loop dynamics $\Phi_{T'}$. In the next chapter, based on (3.28), the details of the tuning algorithms are developed to obtain $\Phi_{T'}$ through control experiments.

Chapter 4

ITERATIVE TUNING IN THE FREQUENCY DOMAIN

Following the gradient estimation theory as presented in the previous chapter, this chapter will develop an online tuning method for ANVC problems with periodic signals.

This Chapter comprises three parts. First of all, the main task of gradient estimation in the frequency domain is to get the frequency response of the closed-loop dynamics, i.e. $\Phi_{T'}$. Some ideas to the online estimation of $\Phi_{T'}$ are briefly presented. One iterative method is studied in detail and the general algorithm for the iterative tuning in the frequency domain is developed.

Secondly, some issues with practical implementations are discussed, including the applications in different controller structures and the also the possible instability problem in the tuning process. At the end of this chapter the proposed online tuning algorithm is compared to some popular adaptive control methods in ANVC, i.e., FXLMS, IFT and MF-FDT.

4.1 Online Iterative Tuning In the Frequency Domain

In the previous chapter the computation of the performance criterion gradient is based on (3.28) or (3.33) for periodic disturbances, where the unknown terms $\Phi_{T'}$ or $\Phi_{T'}|_{\Omega}$ are the key to obtaining the gradient of the criterion.

In the k-th iteration, $\Phi_{T'}^k$ is the closed-loop dynamics, comprising secondary path dynamics $\Phi_{G'}^k$ and k-th feedback controller $\Phi_{H'}^k$. In LTI systems, $\Phi_{G'}^k$ is time invariant that is identical with $\Phi_{G'}$. In this case, if $\Phi_{G'}$ is known a priori through off-line identification, it is easy to compute $\Phi_{T'}^k$. The tuning process becomes straightforward by updating the controller parameters as (3.6) along the negative direction of the performance gradient obtained by (3.28) or (3.33).

4.1.1 Online tuning methods

However, in many ANVC applications the system dynamics G is not available to perform off-line identification, or can be slightly time varying in different operational states. An online gradient estimation technique is necessary and is more valuable in the application than a priori estimation.

According to (3.28) or (3.33) to perform online tuning it is necessary to estimate $\Phi_{T'}$ or $\Phi_{T'}|_{\Omega}$ in the case of periodic disturbances through realtime experiments.

Note that $\Phi_{T'}$ can be presented as an incremental equation as $\Delta\phi_y = \Phi_{T'}\Delta\phi_u$, the unknown dynamics $\Phi_{T'}$ can be estimated through the spectrum difference pair: output spectrum difference ($\Delta\phi_y$) and control action spectrum difference ($\Delta\phi_u$). There are a lot of methods to produce the spectrum difference pair to solve $\Phi_{G'}$.

As in IFT, one of the simplest ways is to perform one extra experiment that injects some additional control excitation $\Delta\boldsymbol{u}$ and acquires the difference in output $\Delta\boldsymbol{y}$ during the original and extra experiments. In implementation, the additional control excitation $\Delta\boldsymbol{u}$ is to be determined by design that takes into account the identical spectrum of the disturbance with output $\boldsymbol{y}$. The extra experiment is illustrated in Fig. 4.1.

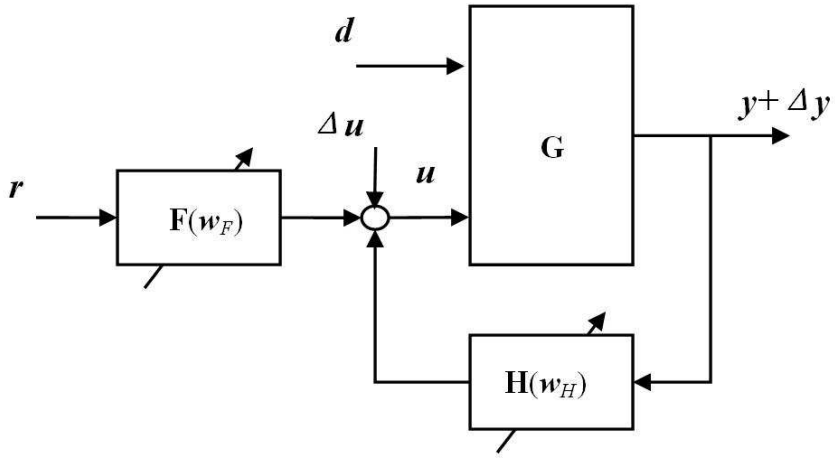

FIGURE 4.1: Extra experiment to estimate gradient in the frequency domain

However, the operation and structure of this method is complicated since there is an extra experiment and extra path to inject additional control actions. To avoid these drawbacks, a new adaptive control method is proposed that tunes the controllers without extra signals and corresponding injection path in each iteration.

The idea of iterative tuning in the frequency domain is simple and it is is also based on the spectrum difference pair $\{\Delta\phi_u, \Delta\phi_y\}$.

1. In k-th iteration, output spectrum ϕ_y^k and control action spectrum ϕ_u^k are obtainable.

2. Given gradient $\nabla J(\boldsymbol{w}^k)$ in the k-th iteration, the parameter vector for $(k+1)$-th iteration, i.e., $\boldsymbol{w}^{k+1}$, is updated as $\boldsymbol{w}^{k+1} = \boldsymbol{w}^k - \mu J(\boldsymbol{w}^k)$.

3. In $(k+1)$-th iteration, applying parameter vector $\boldsymbol{w}^{k+1}$, output spectrum $\boldsymbol{\phi}_y^{k+1}$ and control action spectrum $\boldsymbol{\phi}_u^{k+1}$ can be acquired through experiment.

4. After getting $\boldsymbol{\phi}_y^k$ and $\boldsymbol{\phi}_y^{k+1}$, and $\boldsymbol{\phi}_u^k$ and $\boldsymbol{\phi}_u^{k+1}$, the closed loop dynamics $\Phi_{T'}$ can be estimated through the spectrum difference pair, i.e., $(\Delta\boldsymbol{\phi}_u = \boldsymbol{\phi}_u^k - \boldsymbol{\phi}_u^{k+1})$ - $(\Delta\boldsymbol{\phi}_y = \boldsymbol{\phi}_y^k - \boldsymbol{\phi}_y^{k+1})$. Then $\Phi_{T'}^{k+1}$ and $\nabla J(\boldsymbol{w}^{k+1})$ are obtainable.

It should be noted that, in the step 4 of the above process, the spectrum difference pair, i.e., $(\Delta\boldsymbol{\phi}_u = \boldsymbol{\phi}_u^k - \boldsymbol{\phi}_u^{k+1})$ - $(\Delta\boldsymbol{\phi}_y = \boldsymbol{\phi}_y^k - \boldsymbol{\phi}_y^{k+1})$, can not be directly used in $\Delta\phi_y = \Phi_{T'}\Delta\boldsymbol{\phi}_u$ to estimate $\Phi_{T'}$ because the parameter of feedback controller has been changed from $H(\boldsymbol{w}^k)$ to $H(\boldsymbol{w}^{k+1})$ in k-th and $(k+1)$-th iterations.

There are two methods to estimate $\Phi_{T'}$ through the spectrum difference pair $\{\Delta\boldsymbol{\phi}_u, \Delta\boldsymbol{\phi}_y\}$: the direct estimate approach and indirect estimate approach. They are presented in the following two subsections.

Note that $\Phi_{T'}$ becomes a diagonal frequency response matrix for SISO LTI systems so that the single diagonal element $\Phi_{T'}(\omega)$ in $\Phi_{T'}$ for the frequency response about a single frequency ω can be estimated separately. In order to ease on the notation in the following subsections, the estimate of $\Phi_{T'}$ will be represented with the estimate of a single frequency response $\Phi_{T'}(\omega)$ and the extension to full $\Phi_{T'}$ is straightforward.

4.1.2 Direct estimate approach

The first method is the direct estimation which can be derived from the increment equation of $\Phi_{T'}$ (3.23) directly.

Considering a SISO LTI system G, the control action is denoted by $\boldsymbol{u}^i := \boldsymbol{u}_f^i + \boldsymbol{u}_h^i$ in the i-th iteration and $\boldsymbol{u}^j := \boldsymbol{u}_f^j + \boldsymbol{u}_h^j$ in the j-th iteration, the outputs in the two different iterations are denoted by $\boldsymbol{y}^i$ and $\boldsymbol{y}^j$, and the parameter vectors of the feedback controller are $\boldsymbol{w}_H^i$ and $\boldsymbol{w}_H^j$, respectively.

As mentioned above, the change in the feedback path, i.e., $\Delta\boldsymbol{u}_h = \boldsymbol{u}_h^i - \boldsymbol{u}_h^j$, comprises two parts:

- $\Delta\boldsymbol{\phi}_{uh}^y$ caused by the change of output spectrum $\Delta\boldsymbol{\phi}_y = \boldsymbol{\phi}_y^i - \boldsymbol{\phi}_y^j$,
- $\Delta\boldsymbol{\phi}_{uh}^w$ caused by the change of feedback parameters $\Delta\boldsymbol{w}_H = \boldsymbol{w}_H^i - \boldsymbol{w}_H^j$.

Note that the mapping of $\Phi_{T'}$ is from $\Delta\boldsymbol{\phi}_u^w$ to $\Delta\boldsymbol{\phi}_y$ in (3.23), $\Delta\boldsymbol{\phi}_{uh}^w$ can be obtained from following equation:

$$\Delta\boldsymbol{\phi}^{w}_{uh} = \boldsymbol{\phi}^{i}_{uh}(\omega) - \Phi_H(\boldsymbol{w}^j, \boldsymbol{\phi}^{i}_{y}(\omega)). \tag{4.1}$$

It is straightforward to write the difference equation between i-th and j-th iteration as:

$$\boldsymbol{\phi}^{i}_{y}(\omega) - \boldsymbol{\phi}^{j}_{y}(\omega) = \Phi^{i}_{T'}(\omega)[(\boldsymbol{\phi}^{i}_{uf}(\omega) - \boldsymbol{\phi}^{j}_{uf}(\omega)) + \boldsymbol{\phi}^{i}_{uh}(\omega) - \Phi_h(\boldsymbol{w}^j, \boldsymbol{\phi}^{i}_{y}(\omega))] \tag{4.2}$$

where the term $\Delta\boldsymbol{\phi}^{w}_{uf}$ in (3.23) is represent with $(\boldsymbol{\phi}^{i}_{uf}(\omega) - \boldsymbol{\phi}^{j}_{uf}(\omega))$.

Assuming $\boldsymbol{y}$ has finite dominant frequency Ω, $\Phi^{i}_{T'}|_\Omega$ is a diagonal matrix with diagonal element $\Phi_{T^i}(\omega), \omega \in \Omega$.

It is straightforward to obtain a direct estimate of $\Phi_{T^i}|_\Omega$ as:

$$\hat{\Phi}^{i}_{T'}|_\Omega = \{\mathrm{diag}[\boldsymbol{\phi}^{i}_{uf}|_\Omega - \boldsymbol{\phi}^{j}_{uf}|_\Omega + \boldsymbol{\phi}^{i}_{uh}|_\Omega - \Phi_H|_\Omega(\boldsymbol{w}^j, \boldsymbol{\phi}^{i}_{y}|_\Omega)]\}^{-1}\mathrm{diag}(\boldsymbol{\phi}^{i}_{y}|_\Omega - \boldsymbol{\phi}^{j}_{y}|_\Omega) \tag{4.3}$$

In some cases the control actions cannot be measured directly but the reference signal $\boldsymbol{r}$ is obtainable. The change of control action caused by the change of control parameters, i.e., $\Delta\boldsymbol{\phi}^{w}_{u}$, can be calculated through the following two equations:

$$\Delta\boldsymbol{\phi}^{w}_{uf} = \Phi_F(\boldsymbol{w}^{i}_{F}, \boldsymbol{\phi}_r) - \Phi_F(\boldsymbol{w}^{j}_{F}, \boldsymbol{\phi}_r); \tag{4.4}$$

and

$$\Delta\boldsymbol{\phi}^{w}_{uh} = \Phi_H(\boldsymbol{w}^{i}_{H}, \boldsymbol{\phi}^{i}_{y}) - \Phi_H(\boldsymbol{w}^{j}_{H}, \boldsymbol{\phi}^{i}_{y}); \tag{4.5}$$

Therefore, (4.3) can be rewritten as

$$\begin{aligned}\hat{\Phi}^{i}_{T'}|_\Omega \quad = \quad & \{\mathrm{diag}[\Phi_F(\boldsymbol{w}^{i}_{F}, \boldsymbol{\phi}_r) - \Phi_F(\boldsymbol{w}^{j}_{F}, \boldsymbol{\phi}_r) \\ & \Phi_H(\boldsymbol{w}^{i}_{H}, \boldsymbol{\phi}^{i}_{y}) - \Phi_H(\boldsymbol{w}^{j}_{H}, \boldsymbol{\phi}^{i}_{y})]\}^{-1}\mathrm{diag}(\boldsymbol{\phi}^{i}_{y}|_\Omega - \boldsymbol{\phi}^{j}_{y}|_\Omega)\end{aligned} \tag{4.6}$$

Although the derivation of (4.3) looks somewhat complicated, as shown in (4.3), the computation of $\Phi^{i}_{T'}|_\Omega$ is almost completely based on the measured data except one simple calculation, $\Phi_{H'}|_\Omega(\boldsymbol{w}^j, \boldsymbol{\phi}^{i}_{y}|_\Omega)$.

4.1.3 Indirect estimate approach

Another method is the indirect estimation approach, which computes $\Phi_{T'}$ after estimating $\Phi_{G'}$. It is easier and more flexible than the direct estimation approach.

Note that $\Phi_{T'}$ is the closed-loop dynamics comprising the unknown dynamics G and the feedback controller $H(\boldsymbol{w}_H)$. $\Phi_{T'}$ can be obtained when the designed feedback controller $\Phi_{H'}(\boldsymbol{w}_H)$ is known and an estimate of $\Phi_{G'}$ is available. The estimation of $\Phi_{G'}|_\Omega$ is much simpler than estimation of $\Phi_{T'}|_\Omega$.

$\hat{\Phi}_{G'}|_\Omega$, the realtime estimate of $\Phi_{G'}|_\Omega$, is given by

$$\hat{\Phi}^i_{G'}|_\Omega = [\mathrm{diag}(\boldsymbol{\phi}^i_u|_\Omega - \boldsymbol{\phi}^j_u|_\Omega)]^{-1}\mathrm{diag}(\boldsymbol{\phi}^i_y|_\Omega - \boldsymbol{\phi}^j_y|_\Omega) \tag{4.7}$$

Given $H(\boldsymbol{w}^i_H)$ produced in the i-th iteration and $\hat{\Phi}_{G'}|_\Omega$ estimated with (4.7), it is straightforward to obtain $\hat{\Phi}^i_{T'}|_\Omega$ at i-th iteration as

$$\hat{\Phi}^i_{T'}|_\Omega = (I - \hat{\Phi}_{G'}|_\Omega \Phi^i_{H'}|_\Omega)^{-1}\hat{\Phi}_{G'}|_\Omega \tag{4.8}$$

Considering the two different controller parameter vectors in the i-th and j-th iterations, i.e. $\{\boldsymbol{w}^i_f, \boldsymbol{w}^i_h\}$ and $\{\boldsymbol{w}^j_f, \boldsymbol{w}^j_h\}$, and if the control action $\boldsymbol{u}^i$ and $\boldsymbol{u}^j$ in i-th and j-th iteration are measurable, $\Phi_{T'}$ can be obtained from (4.8) after estimating $\Phi_{G'}$ using (4.7).

In some cases the control action $\boldsymbol{u}$ cannot be measured directly but the reference signal $\boldsymbol{r}$ is measurable. Then the control action caused by the change of parameters can be computed through the following two equations:

$$\Delta\boldsymbol{\phi}_{uf} = \Phi_F(\boldsymbol{w}^i_F, \boldsymbol{\phi}_r) - \Phi_F(\boldsymbol{w}^j_F, \boldsymbol{\phi}_r); \tag{4.9}$$

and

$$\Delta\boldsymbol{\phi}_{uh} = \Phi_H(\boldsymbol{\phi}^i_y, \boldsymbol{w}^i_H) - \Phi_H(\boldsymbol{\phi}^j_y, \boldsymbol{w}^j_H); \tag{4.10}$$

Therefore, (4.7) can be rewritten as

$$\begin{aligned}\hat{\Phi}_{G'}|_\Omega \quad = \quad & \{\mathrm{diag}[\Phi_H|_\Omega(\boldsymbol{w}^i, \boldsymbol{\phi}^i_y|_\Omega) - \Phi_H|_\Omega(\boldsymbol{w}^j, \boldsymbol{\phi}^j_y|_\Omega) \\ & + \Phi_F|_\Omega(\boldsymbol{w}^i, \boldsymbol{\phi}^i_r|_\Omega) - \Phi_F|_\Omega(\boldsymbol{w}^j, \boldsymbol{\phi}^j_r|_\Omega)]\}^{-1}\mathrm{diag}(\boldsymbol{\phi}^i_y|_\Omega - \boldsymbol{\phi}^j_y|_\Omega)\end{aligned} \tag{4.11}$$

After getting $\Phi_{G'}$ with (4.11), the estimate of $\Phi_{T'}$ is straightforward.

Compared with the direct estimation approach, an advantage of the indirect estimation method as described by (4.8) and (4.7) or (4.11) is obvious: in the LTI case $\Phi_{T'}$ can be updated without estimating $\Phi_{G'}$ in each iteration in the indirect estimation approach while $\Phi_{T'}$ have to be estimated in each iteration in the direct estimation approach.

4.1.4 General algorithm of iterative tuning in the frequency domain

According to the discussions in the above two subsections, the estimate of $\Phi_{T'}$ can be obtained from realtime experiments. In the proposed method the estimation is based on a sequence of experiments with different controllers which produce the blocks of data to obtain the differences in spectra. It is a kind of self-tuning control method since the tuning procedure employs blocks of data in the time domain. The proposed online tuning algorithm is called Iterative Tuning (IT) in the Frequency Domain (FD), abbreviated as FD-IT.

To summarize: under the assumption that the system is LTI and all signals contain a finite frequency set Ω only, the general algorithm of FD-IT is given as Algorithm 4.1.1.

Algrithm 4.1.1. **General Algorithm of FD-IT**

At the i-th iteration, given a prior estimate of the set Ω and designed controllers H^i and F^i, perform an experiment and record data with length of the common period of Ω

- I) Estimate Ω through frequency estimation techniques and $\hat{\Phi}_{T'}|_\Omega$ through the following methods:
  - Direct estimation approach
    - Let $j = i - 1$ solve $\hat{\Phi}_{T'}|_\Omega$ with (4.3),
  - Indirect estimation approach
    - a) Let $j = i - 1$, solve $\hat{\Phi}_{G'}|_\Omega$ with (4.7) or (4.11),
    - b) Solve $\hat{\Phi}_{T'}|_\Omega$ with (4.8),
- II) Obtain the derivative of J with (3.27) or (3.33);
- III) Update the controller parameters $\boldsymbol{w}_{i+1}$ with (3.6);
- IV) If not determined, let $i = i + 1$ and return to I)

Based on the above algorithm, given initial stable controllers $\{H(\boldsymbol{w}_H^0), F(\boldsymbol{w}_F^0)\}$, a typical tuning process of FD-IT with indirect estimate approach can be illustrated as in Fig. 4.2:

With regards to the above general algorithm it should be noted that controllers are only presented as $\frac{\partial C(\boldsymbol{w})}{\partial \boldsymbol{w}}$ in mathematical terms, which leaves us with the possibility of various choices to implement the algorithm in different control applications.

4.2 Implement FD-IT in Different Controller Structures

While the general algorithm was studied in the previous section, it can be applied for control in practice through various controller structures. The details of possible implementations in some typical controller structures are studied in this section.

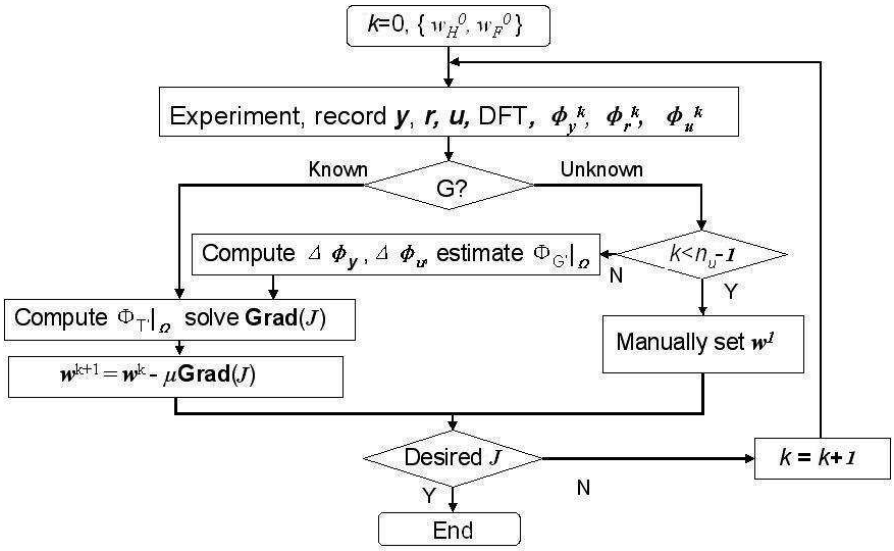

FIGURE 4.2: Block diagram of iterative tuning in the frequency domain with indirect estimate approach

4.2.1 Issues about phase error correction

Note that in (3.28) the $\Phi_{T'}$ is a partial derivative matrix with respect to the spectrum of the control action. This implies that ϕ_d is fixed for $\Phi_{T'}$. Generally, $\boldsymbol{d}$ is considered to be stationary so that the norm of ϕ_d is invariant. However, when the starting point of a acquiring a data sequence with length N changes, the phase ϕ_d can change as well.

Considering IFT, the injection of $\boldsymbol{y}$ and $\boldsymbol{r}$ are required at the same point in the common period to keep identical phases. In [92, 93, 142], it is stated that there is $\frac{\pi}{2}$ a phase error limitation in order to keep tuning convergent.

In the above discussions on FD-IT, the ϕ_{d^i} and ϕ_{d^j} between the i-th and j-th experiments were required to keep identical phases, which required the same starting points in a common period to acquire data.

Fortunately, in FD-IT the phase error can be corrected through reference spectra in implementations. Considering the linear ANVC system described as Fig. 3.1, it is straightforward to write the path from disturbance $\boldsymbol{d}$ to output $\boldsymbol{y}$ in the frequency domain as

$$\phi_y = \left(\frac{\Phi_{G_d}}{I - \Phi_{G_u}\Phi_H} + \frac{\Phi_{G_u}\Phi_F\Phi_S}{I - \Phi_{G_u}\Phi_H}\right)\phi_d \tag{4.12}$$

The reference signal, the feed-forward controller output and the feedback controller output can be written as

$$\begin{aligned} \boldsymbol{\phi}_r &= \Phi_S \boldsymbol{\phi}_d, \\ \boldsymbol{\phi}_{uf} &= \Phi_H \Phi_S \boldsymbol{\phi}_d, \\ \boldsymbol{\phi}_{uh} &= \Phi_H \left(\frac{\Phi_{G_d}}{I - \Phi_{G_u}\Phi_H} + \frac{\Phi_{G_u}\Phi_F\Phi_S}{I - \Phi_{G_u}\Phi_H}\right)\boldsymbol{\phi}_d \end{aligned} \tag{4.13}$$

Considering one experiment with disturbance $\boldsymbol{d}$, output $\boldsymbol{y}$, reference $\boldsymbol{r}$, feed-forward control action $\boldsymbol{u}_f$ and feedback control action $\boldsymbol{u}_h$, their spectra are $\boldsymbol{\phi}_d$, $\boldsymbol{\phi}_y$, $\boldsymbol{\phi}_r$, $\boldsymbol{\phi}_{uf}$ and $\boldsymbol{\phi}_{uh}$, respectively. If there is an offset of the starting point with time difference Δt. i.e., the offset disturbance $\tilde{d}(t) = d(t + \Delta t)$. The phase error can be described with one phase error matrix $\Phi_{\Delta t}$, i.e., $\tilde{\boldsymbol{\phi}}_d = \Phi_{\Delta t}\boldsymbol{\phi}_d$. From (4.13), there exist $\tilde{\boldsymbol{\phi}}_r = \Phi_{\Delta t}\boldsymbol{\phi}_r$, $\tilde{\boldsymbol{\phi}}_y = \Phi_{\Delta t}\boldsymbol{\phi}_y$, $\tilde{\boldsymbol{\phi}}_{uf} = \Phi_{\Delta t}\boldsymbol{\phi}_{uf}$ and $\tilde{\boldsymbol{\phi}}_{uh} = \Phi_{\Delta t}\boldsymbol{\phi}_{uh}$.

While $\boldsymbol{d}$ is unknown but $\boldsymbol{r}$ is measurable, if Φ_S is assumed as LTI system and invertible, $\Phi_{\Delta t}$ is invertible as well, i.e.,

$$\Phi_{\Delta t}^{-1} = \mathrm{diag}(\tilde{\boldsymbol{\phi}}_r)\mathrm{diag}(\boldsymbol{\phi}_r)^{-1} \tag{4.14}$$

It is straightforward to get the phase error in correct format as

$$\begin{aligned} \tilde{\boldsymbol{\phi}}_y &= \Phi_{\Delta t}^{-1}\boldsymbol{\phi}_y, \\ \tilde{\boldsymbol{\phi}}_{uf} &= \Phi_{\Delta t}^{-1}\boldsymbol{\phi}_{uf}, \\ \tilde{\boldsymbol{\phi}}_{uh} &= \Phi_{\Delta t}^{-1}\boldsymbol{\phi}_{uh} \end{aligned} \tag{4.15}$$

Considering the i-th and j-th iteration in FD-IT, there exist the phase errors leading to $\boldsymbol{\phi}_{r^i} = \Phi_{\Delta t}\boldsymbol{\phi}_{r^j}$. From (4.15), the phase error correction for i-th iteration with respect to the j-th iteration can be described as

$$\begin{aligned} \tilde{\boldsymbol{\phi}}_{y^i} &= \mathrm{diag}(\boldsymbol{\phi}_{r^j})\mathrm{diag}(\boldsymbol{\phi}_{r^i})^{-1}\boldsymbol{\phi}_{y^i}, \\ \tilde{\boldsymbol{\phi}}_{uf^i} &= \mathrm{diag}(\boldsymbol{\phi}_{r^j})\mathrm{diag}(\boldsymbol{\phi}_{r^i})^{-1}\boldsymbol{\phi}_{uf^i}, \\ \tilde{\boldsymbol{\phi}}_{uh^i} &= \mathrm{diag}(\boldsymbol{\phi}_{r^j})\mathrm{diag}(\boldsymbol{\phi}_{r^i})^{-1}\boldsymbol{\phi}_{uh^i}, \end{aligned} \tag{4.16}$$

For the direct estimation approach (4.3) can be rewritten as

$$\hat{\Phi}_{T^j} = \mathrm{diag}(\tilde{\boldsymbol{\phi}}_{y^i} - \boldsymbol{\phi}_{y^j})\mathrm{diag}[\tilde{\boldsymbol{\phi}}_{uf^i} - \boldsymbol{\phi}_{uf^j} + \frac{\partial \Phi_{H^j}}{\partial \boldsymbol{w}_H}\boldsymbol{\phi}_{y^j}(\boldsymbol{w}^i - \boldsymbol{w}^j)]^{-1} \tag{4.17}$$

For the indirect estimation approach (4.7) can be rewritten as

$$\hat{\Phi}_{G_u} = [\mathrm{diag}(\tilde{\boldsymbol{\phi}}_{u^i} - \boldsymbol{\phi}_{u^j})]^{-1}\mathrm{diag}(\tilde{\boldsymbol{\phi}}_{y^i} - \boldsymbol{\phi}_{y^j}) \tag{4.18}$$

With (4.16)-(4.17), the phase error can be corrected by using invariant reference spectrum in FD-IT.

4.2.2 Issues about variety of the controller

It is noted that there is no limitation on the structure and mathematical format of controllers in FD-IT except the existence of derivative matrix $\frac{\partial \Phi_C(\boldsymbol{\phi}_y, \boldsymbol{\phi}_r, \boldsymbol{w})}{\partial \boldsymbol{w}}|_\Omega$. The structure of implementation and the mathematical format are flexible for implementing FD-IT in applications.

Most common structures of controller, such as: FIR controller, IIR controller, neural-network (NN) controllers and fuzzy logic controllers, can be used in FT-IT, except some controllers with undifferentiable factors, such as, hard threshold functions in neural-networks with and hard membership function in fuzzy logics.

In Algorithm 4.1.1, the updating of controllers to provide controller tuning is represented in terms of $\frac{\partial \Phi_C(\boldsymbol{\phi}_y, \boldsymbol{\phi}_r, \boldsymbol{w})}{\partial \boldsymbol{w}}$. Given certain hardware implementation of controllers, the variety of mathematical formats can be chosen according to varying conditions of applications.

In most of the applications the polynomial format is the first choice for its simplicity in mathematics and in implementation. An IIR controller can be represented in polynomial format as

$$y(t) = b(1)x(t-1) + \ldots + b(p)x(t-p) - a(1)y(t-1) - \ldots - a(q)y(t-q) \tag{4.19}$$

which is a typical q-order AutoRegressive (AR) p-order Moving Average (MA) model with parameter vector $\boldsymbol{w} := \{a(1), \ldots, a(q), b(1), \ldots, b(p)\}$, the partial derivative in the frequency domain of the controller with $\boldsymbol{w}$ can be written as

$$\frac{\partial \phi_y(\boldsymbol{w})}{\partial a(i)}|_\omega = -\phi_y|_\omega \frac{e^{-ji\omega}}{1 + \sum_{k=1}^{q}[a(k)e^{-jk\omega}\phi_x(\omega)]}, i = 1, \ldots, q; \tag{4.20}$$

where $\Phi_C(\phi_x(\omega), \boldsymbol{w})|_\omega = \frac{\sum_{k=1}^{p}[b(k)e^{-jk\omega}\phi_x(\omega)]}{1+\sum_{k=1}^{q}[a(k)e^{-jk\omega}\phi_x(\omega)]}$, and

$$\frac{\partial \phi_y(\boldsymbol{w})}{\partial b(i)}|_\omega = \frac{e^{-ji\omega}}{1 + \sum_{k=1}^{q}[a(k)e^{-jk\omega}\phi_x(\omega)]}, i = 1, \ldots, p. \tag{4.21}$$

In some cases the Zeros-Poles-Gain (ZPK) format is more straightforward and suitable to make tuning take into consideration the stability problem concurrently.

The discrete transfer function of an IIR controller can be represent in the ZPK format as

$$h(z) = k\frac{\prod_{i=1}^{n_z}[z - r_z(i)]}{\prod_{i=1}^{n_p}[z - r_p(i)]} \tag{4.22}$$

where the zeros are $\{r_z(1), \ldots, r_z(n_z)\}$, poles are $\{r_p(1), \ldots, r_p(n_p)\}$, gain is k, and the parameter vector is $\boldsymbol{w} := \{r_z(1), \ldots, r_z(n_z), r_p(1), \ldots, r_p(n_p), k\}$.

It is straightforward to get the frequency domain's derivative function as follows:

$$\frac{\partial \phi_y(\boldsymbol{w})}{\partial r_z(i)}|_\omega = -\frac{\phi_y|_\omega \phi_x|_\omega}{e^{-ji\omega} - r_z(i)},\ i = 1, \ldots, n_z \tag{4.23}$$

$$\frac{\partial \phi_y(\boldsymbol{w})}{\partial r_p(i)}|_\omega = \frac{\phi_y|_\omega \phi_x|_\omega}{e^{-ji\omega} - r_p(i)},\ i = 1, \ldots, n_p \tag{4.24}$$

and

$$\frac{\partial \phi_y(\boldsymbol{w})}{\partial k}|_\omega = \frac{\phi_y|_\omega \phi_x|_\omega}{k} \tag{4.25}$$

While the polynomial format is simple to use, ZPK format is valuable when considering stability of IIR controllers. As state above, the mathematical format of a controller is only the representation of the controller in the tuning algorithm and independent from the numerical hardware implementation. It can be selected by algorithm designers according to different conditions.

The hybrid format can be used to combine advantages of the polynomial format and the ZPK format so that the numerator is in polynomial format and the denominator in poles format, i.e.,

$$y(t) = \frac{b(1)z + \ldots + b(p)z^p}{\prod_{i=1}^{n_p}[z - r_p(i)]}x(t) \tag{4.26}$$

It is straightforward to get then that

$$\frac{\partial \phi_y(\boldsymbol{w})}{\partial b(i)}|_\omega = \frac{e^{-ji\omega}\phi_x(\omega)}{\prod_{k=1}^{n_p}[1 - e^{-jk\omega}]}, i = 1, \ldots, p. \tag{4.27}$$

and

$$\frac{\partial \phi_y(\boldsymbol{w})}{\partial r_p(i)}|_\omega = \frac{\phi_y|_\omega \phi_x|_\omega}{e^{-ji\omega} - r_p(i)},\ i = 1, \ldots, n_p \tag{4.28}$$

Furthermore, it should be noted that the mathematical format can be changed during the tuning process, which will be discussed in the next section.

4.2.3 Frequency selective filtered based iterative tuning

In this subsection an important filter structure, namely the Frequency-Selective-Filter (FSF), is introduced to implement FD-IT. It is particularly suitable for control problems with periodic signals.

Fig. 4.3 illustrates the block diagram of a Frequency-Selective-Filter-based (FSF) controller system with m FSF channels. In this system, the FSF controller group can updated in realtime according to the signal's spectrum and maintained in both feedback and feed-forward paths.

Using the FSF-based controller system shown in Fig. 4.3, the implementation of FD-IT is straightforward.

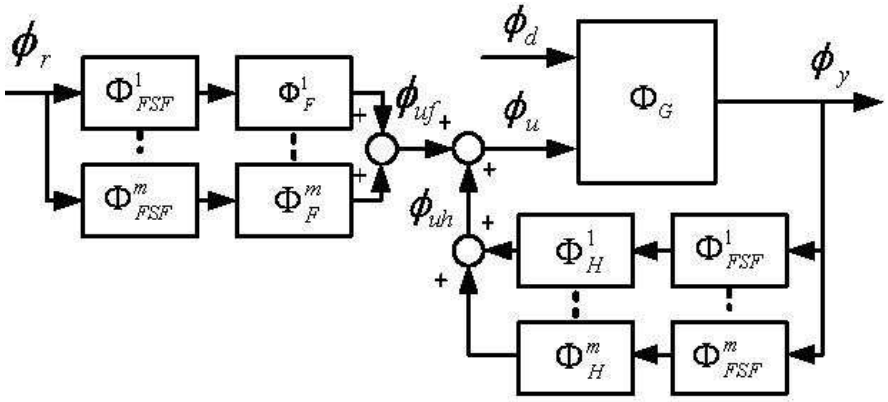

FIGURE 4.3: Block diagram in frequency domain of FSF-based controller system

Considering ideal Frequency-Selective-Filter with gain 1 at the pass frequency and gain 0 at other frequencies, $\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}|_\Omega$ in (3.33) is written as

$$\begin{bmatrix} \frac{\partial \Phi_C^{(0)}(\boldsymbol{\phi}_y(\omega_0),\boldsymbol{\phi}_r(\omega_0),\boldsymbol{w})}{\partial \boldsymbol{w}} & \mathbf{0} & 0 \\ \vdots & \ddots & \vdots \\ 0 & \mathbf{0} & \frac{\partial \Phi_C^{(n_\Omega-1)}(\boldsymbol{\phi}_y(\omega_{(n_\Omega-1)}),\boldsymbol{\phi}_r(\omega_{(n_\Omega-1)}),\boldsymbol{w})}{\partial \boldsymbol{w}} \end{bmatrix} \tag{4.29}$$

A real Frequency-Selective-Filter always has frequency leakage, which is not zero gain in the non-pass-frequency but often ignored when the gain is small enough. In practice, when the dominant frequencies are too close, the frequency leakage of FSF cannot be ignored. Considering the frequency leakage, $\frac{\partial \Phi_C(\boldsymbol{w},\boldsymbol{\phi}_y,\boldsymbol{\phi}_r)}{\partial \boldsymbol{w}_i}|_\Omega$ should be written as

$$\begin{bmatrix} \Phi^{(0)}_{FSF}(\boldsymbol{\phi}_y(\omega_0),\boldsymbol{\phi}_r(\omega_0),\boldsymbol{w})\frac{\partial \Phi_C^{(0)}(\boldsymbol{\phi}_y(\omega_0),\boldsymbol{\phi}_r(\omega_0),\boldsymbol{w})}{\partial \boldsymbol{w}} & \dots \\ \vdots & \ddots \\ \Phi^{(0)}_{FSF}(\boldsymbol{\phi}_y(\omega_{(n_\Omega-1)}),\boldsymbol{\phi}_r(\omega_{(n_\Omega-1)}),\boldsymbol{w})\frac{\partial \Phi_C^{(0)}(\boldsymbol{\phi}_y(\omega_{(n_\Omega-1)}),\boldsymbol{\phi}_r(\omega_{(n_\Omega-1)}),\boldsymbol{w})}{\partial \boldsymbol{w}} & \dots \end{bmatrix}$$
$$\begin{bmatrix} \dots & \Phi^{(n_\Omega-1)}_{FSF}(\boldsymbol{\phi}_y(\omega_0),\boldsymbol{\phi}_r(\omega_0),\boldsymbol{w})\frac{\partial \Phi_C^{(n_\Omega-1)}(\boldsymbol{\phi}_y(\omega_0),\boldsymbol{\phi}_r(\omega_0),\boldsymbol{w})}{\partial \boldsymbol{w}} \\ \ddots & \vdots \\ \dots & \Phi^{(n_\Omega-1)}_{FSF}(\boldsymbol{\phi}_y(\omega_{(n_\Omega-1)}),\boldsymbol{\phi}_r(\omega_{(n_\Omega-1)}),\boldsymbol{w})\frac{\partial \Phi_C^{(n_\Omega-1)}(\boldsymbol{\phi}_y(\omega_{(n_\Omega-1)}),\boldsymbol{\phi}_r(\omega_{(n_\Omega-1)}),\boldsymbol{w})}{\partial \boldsymbol{w}} \end{bmatrix} \tag{4.30}$$

4.3 Issues about Stability of FD-IT

Stability is one of the most important topics in control engineering. In previous discussions, the general algorithm of FD-IT has not considered the stability of the control system after applying

the updated controllers. In the following subsections, the stability of FD-IT is discussed to tune feed-forward and feedback controllers, respectively.

In FD-IT, while initial stability is guaranteed by prior selected initial controllers, the stability during tuning process is depended on choosing a proper small step size μ. In the following two subsections we will study how to determine the proper step size μ to update feed-forward controllers and feedback controllers.

4.3.1 Issues about stability of IIR controller

Firstly, the stability of feed-forward tuning is discussed when the tuned closed-loop dynamics T is assumed to be stable. The sufficient and necessary condition to make feed-forward controller F to stabilize system as Fig.3.1 is to locate all the poles of F in the unit circle. Obviously, FIR feed-forward controllers always stabilize a stable closed-loop dynamics since FIR filters' poles are all zeros. The remaining question is then how to tune IIR feed-forward controllers in FD-IT.

In order to satisfy the pole assignment rule in the discrete system that all the poles should be within the unit circle [78], the most straightforward method is to make trial error and test if poles of updated IIR controllers are within unit circle before applying controllers into real systems. The trial and test algorithm is

Algrithm 4.3.1. **Trial and Test for IIR feed-forward controller in FD-IT**

At the i-th iteration, initial conditions are satisfied as

1. the closed-loop dyanmics $T := \{G, H(\boldsymbol{w}^i)\}$ is stable,
2. $C(\boldsymbol{w}^i) := \{F(\boldsymbol{w}_F^i), H(\boldsymbol{w}_H^i)\}$ and $\nabla J(\boldsymbol{w}^i)$ is known,
3. a initial step size μ_0^i and limitation for $\mu^i \in (\mu_{min}^i, \mu_{max}^i)$ are manually set.

the trial and test method to update the IIR feed-forward controller is:

Let $k = 0$,

- I) Compute trial IIR $F(\boldsymbol{w}_k^{i+1})$ as $\boldsymbol{w}_k^{i+1} = \boldsymbol{w}^i + \mu_k^i \nabla J(\boldsymbol{w}_F^i)$,
- II) Compute poles of $F(\boldsymbol{w}_k^{i+1})$ and test if the poles are all within unit circle,
- III) If No, let $k = k+1$, change $\mu_k^i = \mu_{k-1}^i + \Delta$, where $\Delta \neq 0$ make $\mu_k^i \in (\mu_{min}^i, \mu_{max}^i)$, and goto I),
- IV) If Yes, $\boldsymbol{w}^{i+1} = \boldsymbol{w}_k^{i+1}$ and apply $C(\boldsymbol{w}^{i+1})$ to the next experiment.

Algorithm 4.3.1 can be used for any mathematical format of linear controllers, including polynomial format, gain-zeros-poles (ZPK) format and state-space format. The drawback is the high cost of trials and tests.

To avoid cost of trial and test, ZPK format as (4.22) can be adopted to represent IIR controllers in mathematics. Given $\nabla J(\boldsymbol{w})$ in ZPK format as (4.23), (4.24) and (4.25), the updating algorithm is quite straightforward since poles of updating the controller can be computed directly with

$$r_p^{i+1} = r_p^i - \mu \frac{\partial \phi_y(\boldsymbol{w})}{\partial r_p(i)} \tag{4.31}$$

The pure algorithm to tune ZPK format controllers can be considered an implementation of the Homotopy Continuation Method (HCM) in control engineering [45, 147], which searches optimal root solutions along their separate root pathway. Since the poles are limited to the unit circle, it is possible to find out the global optimal solution for the poles while fixing zeros. Like bifurcation points in the homotopy continuation method, the pure ZPK tuning algorithm may merge some pole values from different update tracks during tuning process. In this latter case the pure ZPK tuning can not divide them in the subsequent updates.

There are several ways to solve this problem. As in HCM, the update directions of bifurcation points are determined by introducing 2nd order derivatives, then the Hermitian matrix can be used to avoid merge of the zeros and poles. From the aspect of control implementation, since one controller can be expressed in different mathematical formats, the polynomial format can be used to substitute the ZPK format when bifurcation points appear. Alg. 4.3.2 gives the scheme of algorithm of such hybrid polynomial-ZPK tuning.

Algrithm 4.3.2. **Hybrid Polynomial-ZPK tuning in FD-IT**

At the i-th iteration, initial conditions are satisfied as

1. the closed-loop dyanmics $T := \{G, H(\boldsymbol{w}^i)\}$ is stable,
2. ZPK format $C(\boldsymbol{w}^i)$ and $\nabla J(\boldsymbol{w}^i)$ is known,

the hybrid tuning is

- I) If there is no duplicate poles or zeros, goto II), else goto III).
- II) Choose proper step size $\mu_z^i, \mu_p^i, \mu_k^i$, guarantee $||r_p^{i+1}||_2 < 1$ through (4.31), and update zeros r_z^{i+1}, poles r_p^{i+1} and gains r_k^{i+1} like (4.31). Goto IV).
- III) Change ZPK format of $C(\boldsymbol{w}^i)$ to polynomial format and update corresponding coefficients with Alg. 4.3.1. Change the polynomial format to IIR format and goto IV).
- IV) $i = i + 1$, goto I).

4.3.2 Issues about stability of closed-loop dynamics

After dealing with stability problem of IIR controllers for feed-forward control, another important issue of system stability is how to stabilize closed-loop dynamics $T := \{G, H\}$.

In order to discuss the stability problems of $T := \{G, H\}$, one of the possible approach is to use generalized stability margin $b_{G,H}$ [170]:

$$b_{G,H} := \begin{cases} ||T(G,H)||_\infty^{-1} & \text{for the stable closed-loop} \\ 0 & \text{for the unstable closed-loop} \end{cases} \tag{4.32}$$

where

$$T(G,H) := \begin{pmatrix} \frac{GH}{1-GH} & \frac{G}{1-GH} \\ \frac{H}{1-GH} & \frac{1}{1-GH} \end{pmatrix} \tag{4.33}$$

Given two LTI feedback controllers H_1 and H_2, if $T(G, H_1)$ is stable, the sufficient condition to make $T(G, H_2)$ stable is that the Vinnicombe distance $\delta_\nu(H_1, H_2)$ [170],

$$\begin{aligned} \delta_\nu(H_1,H_2) &:= ||\frac{H_1 - H_2}{\sqrt{1+H_2^* H_2}\sqrt{1+H_1^* H_1}}||_\infty \\ &= \max(\frac{H_1(j\omega) - H_2(j\omega)}{\sqrt{1+H_2^*(j\omega)H_2(j\omega)}\sqrt{1+H_1^*(j\omega)H_1(j\omega)}}, -\infty < \omega < \infty) \end{aligned} \tag{4.34}$$

should satisfy $\delta_\nu(H_1, H_2) < b_{G,H_1}$.

Therefore, while b_{G,H_1} is unknown but H_1 is known, $\delta_\nu(H_1, H_2)$ is a suitable criterion to describe the robust performance of the closed-loop dynamics $\{G, H_2\}$.

There are two methods to improve the stability of the control system.

Firstly, as direct iterative tuning via spectral analysis [65], one extra experiment can be added into each iteration with some additional zero-mean white-noise signal $\Delta\boldsymbol{u}$ injected into the closed-loop dynamics as in Fig. 4.1.

Assuming $\Delta\boldsymbol{u}$ having variance δ_u^2, the system has changed of output $\Delta\boldsymbol{y}$ and changed feedback bath $\Delta\boldsymbol{u}_h$, which is illustrated in Fig. 4.4.

In the extra stability experiment, there exist

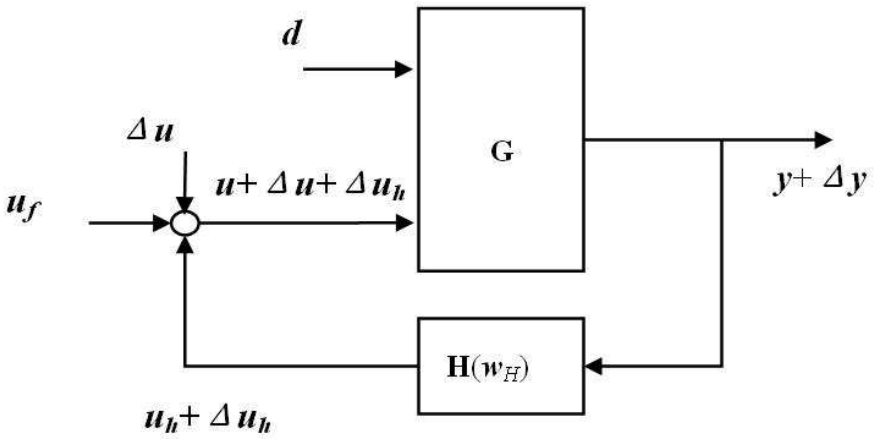

FIGURE 4.4: Extra stability experiment

$$\Delta \boldsymbol{u}_h = \frac{GH}{1-GH}\Delta \boldsymbol{u}, \tag{4.35}$$

$$\Delta \boldsymbol{y} = \frac{G}{1-GH}\Delta \boldsymbol{u}, \tag{4.36}$$

$$\Delta \boldsymbol{u}_h = \frac{H}{1-GH}\Delta \boldsymbol{y}, \tag{4.37}$$

$$\Delta \boldsymbol{u} + \Delta \boldsymbol{u}_h = \frac{1}{1-GH}\Delta \boldsymbol{u}. \tag{4.38}$$

According to the definition of norm of matrix [64], the generalized stability margin b_{G,H_i} in i-th iteration can be estimated with the following equation:

$$\hat{b}_{G,H_i} \doteq |\frac{\Delta \boldsymbol{\phi}_u}{\Delta \boldsymbol{\phi}_{u_h}}, \frac{\Delta \boldsymbol{\phi}_u}{\Delta \boldsymbol{\phi}_y}, \frac{\Delta \boldsymbol{\phi}_y}{\Delta \boldsymbol{\phi}_{u_h}}, \frac{\Delta \boldsymbol{\phi}_{u_h}}{\Delta \boldsymbol{\phi}_u + \Delta \boldsymbol{\phi}_{u_h}}|_\infty \tag{4.39}$$

After getting the estimated generalized stability margin $\hat{b}_{G,H_i}$ through extra experiment, the Vinnicombe distance $\delta_\nu(H_i, H_{(i+1)})$ can be obtained from (4.34), and stability of the updated controller $H(\boldsymbol{w}_{(i+1)})$ can be guaranteed by

$$\delta_\nu(H_i, H_{(i+1)}) < \hat{b}_{G,H_i} \tag{4.40}$$

However, there are some limitations with this method. At first, it requires one extra experiment in each iteration. At the same time, in practice, the ideal white-noise signal $\Delta \boldsymbol{u}$ is not available, and the exact spectrum $\{\Delta \boldsymbol{\phi}_y, \Delta \boldsymbol{\phi}_{u_h}\}$ is not obtainable either as it is only estimated from $\{\Delta \boldsymbol{y}, \Delta \boldsymbol{u}_h\}$.

The second method is a practical approach by introducing a synthetical cost function including stability performance.

Considering the initial condition that no feedback controller is applied, $\boldsymbol{w}_{H_0} = \mathbf{0}$ and $H(\mathbf{0}) \equiv 0$, the Vinnicombe distance between $H(\boldsymbol{w}_H)$ to $H(\mathbf{0})$ can be written with a simpler format as

$$\delta_\nu(\boldsymbol{w}_H) = ||\frac{H(\boldsymbol{w}_H)}{\sqrt{1 + H^*(\boldsymbol{w}_H)H(\boldsymbol{w}_H)}}||_\infty. \tag{4.41}$$

In order to consider stability properties, synthetic performance J_{syn} including output performance $J_P(\boldsymbol{w})$ as (3.29) and a robustness performance $J_R(\boldsymbol{w}_H)$ can be defined as

$$\begin{aligned} J_{syn}(\boldsymbol{w}_H) &= J_P(\boldsymbol{w}_H) + \lambda J_R(\boldsymbol{w}_H) \\ J_P: &= \frac{1}{N}\sum_{t=0}^{N-1} y^2(t) \\ J_R: &= \delta_\nu(\boldsymbol{w}_H) \end{aligned} \tag{4.42}$$

where λ is an appropriate weight factor preset by designer.

In this case, $H(\boldsymbol{w}_H)$ is known and J_R can be obtained. It should be noted that introduction of J_{syn} is only a melioration of the performance function that leads to sufficient generalized stability margin $b_{G,H}$, but there is not guarantee to stability of the closed-loop system.

4.4 Comparison with Other Adaptive Control Methods in ANVC

In this section, the tuning procedure of FD-IT is compared with the tuning procedures of some other typical adaptive control methods in ANVC. Through the comparison, FD-IT presents the advantage of simplicity in control structure and in convenience of operation.

4.4.1 Tuning procedure of FD-IT

At first the tuning procedure of FD-IT is studied in order to compare it with other methods.

As an iterative tuning control method, FD-IT is based on separate experiments which produce signals by iteration block. In ANVC problems, the length of the data block is always the common period of signals N. Controller updating happens after each iteration, hence there are often significant phases during tuning process.

In the case of a LTI plant and when the secondary path dynamics G is known a priori, the tuning procedure is straightforward. The relationship between the tuning iteration and the experiment is illustrated in Fig. 4.5.

When the secondary path dynamics G is prior unknown, as shown in Fig. 4.6, at the beginning of the tuning, one extra experiment with the controller parameters is manually performed. It is

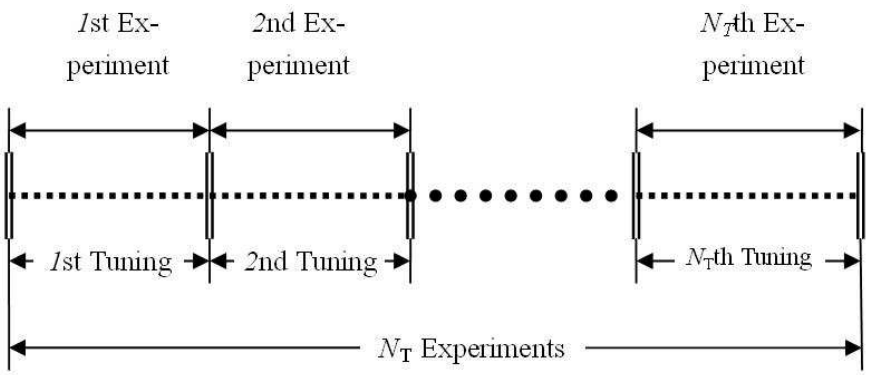

FIGURE 4.5: Tuning procedure of FD-IT with known G in LTI cases

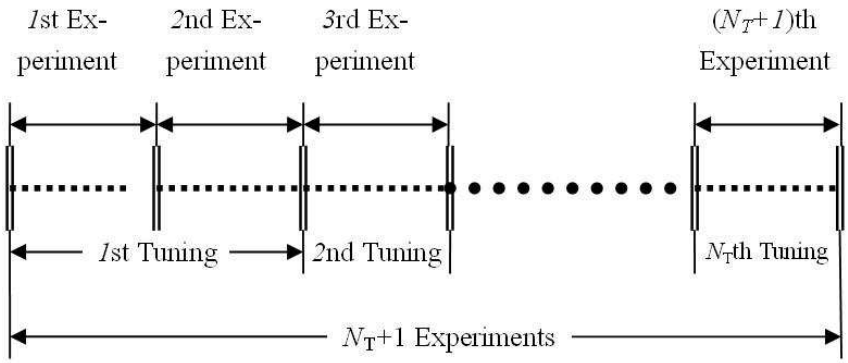

FIGURE 4.6: Tuning procedure of FD-IT with online estimate of G in LTI cases

then necessary to estimate G in LTI systems. After estimating G, the following procedure is the same as the procedure shown in Fig. 4.5.

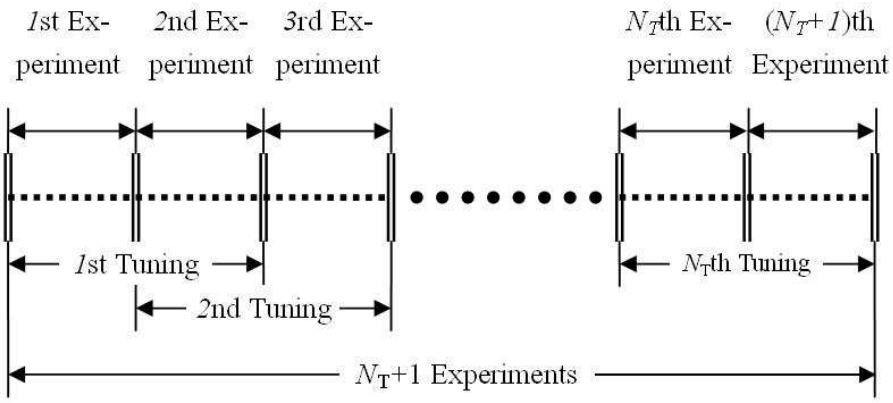

FIGURE 4.7: Tuning procedure of FD-IT in a slow LTV system

FD-IT can be used in slow LTV system as well, i.e. when the time variation of system dynamics is much slower to take place than the common period of the disturbance. In this case, $\hat{\Phi}^k_{G'}|_\omega$ is varying but can be estimated realtime through the data from k-th and $(k-1)$-th experiments. The relationship between tuning iteration and experiments can be illustrated in Fig. 4.7.

In all above procedures the controller structure is simple as in Fig. 3.1 without additional path to inject extra signals.

4.4.2 Comparison with FXLMS methods

As mentioned in Chapter 2, FXLMS and its permutations belong to adaptive control methods that tune controllers continuously. When the secondary path dynamics G is known, as shown in (2.8), the controller parameter vector $\boldsymbol{w}$ is tuned continuously in FXLMS. It requires neither extra experiments to operate nor additional injection paths in structures. Obviously, FXLMS method is one of the most simple and easy ways to control effectively.

When the secondary path dynamics G is a priori unknown, the update of controller parameters can also be based on Recursive-Least-Squares method as (2.16). Although the implementation of FXLMS_RLS is relatively simple as well, it is however not a gradient descent tuning method, but it may have the advantage of faster convergence of controller parameters and higher computational load.

There are three major factors to impact on the convergence speed of FXLMS with RLS method [50, 138]:

1. G is often IIR dynamics, which yields an infinite dimension vector $\Theta(t)$ in (2.13).
2. RLS method as (2.16) does not give steepest decent direction to update $\boldsymbol{w}$ in (2.8).
3. The speed of convergence is also determined by the adoption gain μ, which is priori unknown and quite varying in different control problems.

Therefore,the speed of convergence of FXLMS with RLS method may be very slow in practice.

Although FXLMS method is more convenient to use than FD-IT when the secondary path G is known, FD-IT offers steepest descent tuning with relatively simple operations in practice.

4.4.3 Comparison with other iterative tuning methods

The iterative feedback tuning in the time domain (IFT) and model-free frequency domain tuning (MF-FDT) are both iterative tuning control methods.

In IFT, there are two extra experiments, one for the feedback controller, another for the feedforward controller. The tuning procedure of IFT can be illustrated as in Fig. 4.8. In order to make N_T tuning iterations, $3N_T$ experiments are required.

Similar with IFT, one extra experiment is required to estimate the dynamics of the secondary path G in MF-FDT. Therefore, $2N_T$ experiments are required to make N_T tunings.

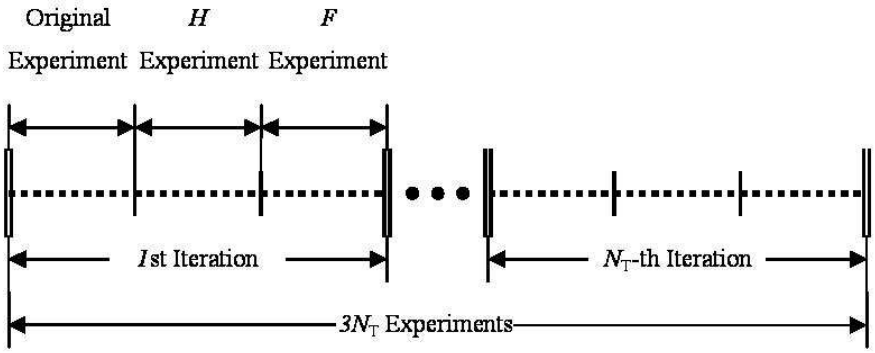

FIGURE 4.8: Tuning procedure of IFT

Compared with FD-IT, these two iterative tuning control methods require auxiliary path to inject additional signals and extra experiments. They are more complicated in control structure and implementation.

4.5 Summary

Based on the gradient estimation method proposed in Chapter 3, this chapter describes gradient-based iterative tuning methods in control implementation. The essential idea is presented to estimate $\Phi_{T'}$ through the spectrum difference pairs $\{\Delta\phi_u, \Delta\phi_y\}$ from the different control experiments.

Iterative Tuning in the Frequency Domain (FD-IT) is studied in detail, which estimates $\Phi_{T'}$ through a sequence of experiments during the tuning process. There is no requirement on extra experiments and additional signal injections except one extra experiments in the initial tuning iteration.

Some important issues about control implementation are discussed that include: implementation in different controller formats and the instability problem during tuning process.

Comparison with some other popular adaptive control method in ANVC is also presented that underlines the convenience and flexibility of FD-IT in implementations.

Chapter 5

Frequency Domain Iterative Tuning in MIMO Systems

In previous chapters all discussion has been based on SISO LTI systems in order to ease the presentation of the gradient theory and control algorithm. This Chapter discusses the issues about extension of FD-IT for SISO to Multiple Input Multiple Output (MIMO) LTI system.

This chapter contains three parts. In the first part a general framework is proposed for MIMO systems in ANVC problems. The definition of the extended FRF derivative matrix for MIMO systems is given in the second section. In the third section the FD-IT method is discussed in a general framework and the algorithms of FD-IT in the MIMO case are studied.

5.1 MIMO Linear ANVC

In practical applications, control systems are always complicated and affected by many factors that may necessitate multiple outputs and multiple control actions. For instance, in Fig. 3.1 the feedback and feed-forward control system can be considered as a SISO system when feedback and feed-forward control actions share the same secondary path G. However, this system should be considered as a MIMO system in practice because the secondary paths for feedback and feed-forward control actions are always different as the feedback and feed-forward control signals use different actuators. Therefore the extension of FD-IT to the MIMO case is very important for applications.

In this section a general framework is proposed to describe the general LTI ANVC problem including MIMO dynamics and some definitions are given.

A general ANVC system is illustrated in Fig. 5.1. As this is different from the SISO LTI system shown in Fig. 3.1, there are two major points to note:

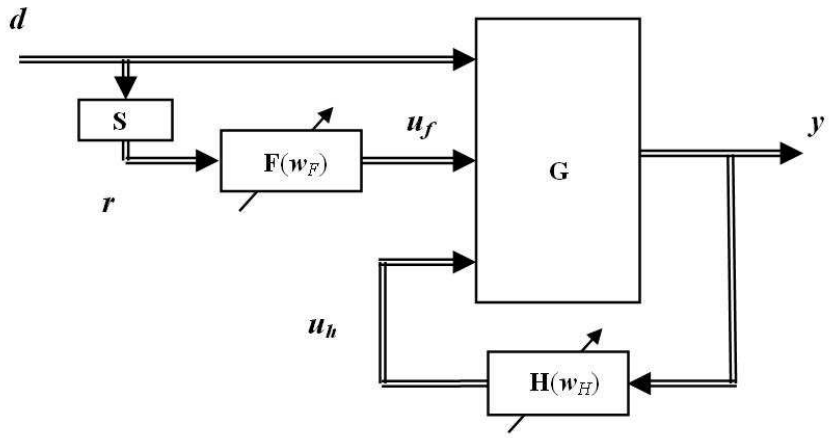

FIGURE 5.1: Block diagram of a MIMO ANVC system with feedback and feed-forward controllers.

- In Fig. 5.1, the double line connections between the modules denote transmission paths containing multiple signals.
- The feedback and feed-forward controllers can have independent secondary paths in Fig. 5.1.

The measured output that is affected by the disturbance $\boldsymbol{d} \in \mathbb{R}^{n_d}$ is represented by $\boldsymbol{y} \in \mathbb{R}^{n_y}$. G is the unknown plant dynamics with inputs $\boldsymbol{d}$ and $\boldsymbol{u} \in \mathbb{R}^{n_u}$, and produce $\boldsymbol{y}$. It can be described as

$$\boldsymbol{y} = G(\boldsymbol{d}, \boldsymbol{u}) \tag{5.1}$$

The control signals $\boldsymbol{u}$ from the feed-forward controller F and feedback controller H are denoted by $\boldsymbol{u}_f \in \mathbb{R}^{n_u}$ and $\boldsymbol{u}_h \in \mathbb{R}^{n_u}$, respectively. The tunable control system C comprises the parameterized feed-forward controller F and the feedback controller H:

$$\begin{aligned} C(\boldsymbol{w}, \boldsymbol{r}, \boldsymbol{y}): \quad F: \boldsymbol{u}_f &= F(\boldsymbol{w}_F, \boldsymbol{r}) \\ H: \boldsymbol{u}_h &= H(\boldsymbol{w}_H, \boldsymbol{y}) \\ \boldsymbol{u} &= (\boldsymbol{u}_f, \boldsymbol{u}_h) \end{aligned} \tag{5.2}$$

which can be tuned by adjusting their parameter vectors $\boldsymbol{w} := \{\boldsymbol{w}_F, \boldsymbol{w}_H\} \in \mathbb{R}^{n_w}$.

It is assumed that the disturbance detection ('reference') signal $\boldsymbol{r} \in \mathbb{R}^{n_r}$ is not correlated with the output $\boldsymbol{y}$, and $\boldsymbol{r}$ is often obtained through an unknown but time-invariant dynamics S from $\boldsymbol{d}$. While the output signal $\boldsymbol{y}(t)$ is measurable and recordable, the disturbance signal $\boldsymbol{d}$ cannot be measured directly.

When the system has steady output $\boldsymbol{y}$, the cost function of such an ANVC system is always defined as the average quadratic performance of a length N output sequence:

$$J(\boldsymbol{w}) := \frac{1}{N}\sum_{t=0}^{N-1} \boldsymbol{y}^{\mathrm{T}}(t) Q \boldsymbol{y}(t) \tag{5.3}$$

where Q is a priori defined weighting matrix.

Similarly as in the SISO LTI case, a general ANVC problem can be expressed as an optimization problem in formally as (5.4)

$$\begin{aligned} \min: \quad & J(\boldsymbol{w}) \text{ in (5.3)} \\ s.t. \quad & \begin{cases} \text{Eqn. (5.1)} \\ \text{Eqn. (5.2)} \end{cases} \end{aligned} \tag{5.4}$$

Comparing the above mathematical expression (5.4) to the optimization problem in SISO LTI 3.5, they are almost the same in mathematics and the solutions are always similar in theory.

In fact, based on the above general framework shown in Fig. 5.1, the deduction of a gradient estimate in the frequency domain for MIMO LTI case is almost same as the SISO LTI case, after making some slight modifications of the derivative matrix of FRF in MIMO systems.

5.2 Extended FRF derivative matrix in MIMO systems

In this section the dynamics of MIMO systems is expressed in the frequency domain and the extended FRF derivative matrix is introduced for linear MIMO systems.

Considering the MIMO linear system described by Fig. 5.1 there is an N-length output data set with the index of time $\mathcal{Y} := \{\boldsymbol{y}(0); \ldots; \boldsymbol{y}(N-1)\}$, where $\boldsymbol{y}(t) := \{y_1(t), \ldots, y_{ny}(t)\} \in \mathrm{R}^{n_y}$ is the output at time t. This can be rewritten with the index of the output channels as $\mathcal{Y} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{n_y}\}$ where $\boldsymbol{y}_i = \{y_i(0); \ldots; y_i(N-1)\}, i = 1, \ldots, n_y$ is the output in the i-th output channel.

As with the notation of discrete spectrum in SISO case, $\omega_m := \frac{2\pi}{N} m, m = 0, \ldots, N-1$ is defined as m-th discrete frequency for N-length data in MIMO system. Considering one single output in one channel in multiple output $\mathcal{Y}$, $\boldsymbol{\phi}_y^i := \{\phi_y^i(\omega_0); \ldots; \phi_y^i(\omega_{N-1})\} \in \mathbb{C}^N$ denotes the discrete spectrum of the i-th output $\boldsymbol{y}_i$.

Therefore, the discrete spectrum of $\mathcal{Y}$ is described as $\boldsymbol{\phi}_y := \{\boldsymbol{\phi}_y^1; \ldots; \boldsymbol{\phi}_y^{n_y}\} \in \mathbb{C}^{(n_y \times N)\times 1}$. There are similar notations used for $\boldsymbol{\phi}_d$, $\boldsymbol{\phi}_r$, $\boldsymbol{\phi}_{uf}$ and $\boldsymbol{\phi}_{uh}$.

In the frequency domain the plant G is described as function $\{\boldsymbol{\phi}_d, \boldsymbol{\phi}_u\} \in \mathbb{C}^{((n_d+n_u)\times N)\times 1} \mapsto \boldsymbol{\phi}_y \in \mathbb{C}^{(n_y \times N)\times 1}$:

$$\boldsymbol{\phi}_y = \Phi_G(\boldsymbol{\phi}_d, \boldsymbol{\phi}_u) = \Phi_G(\boldsymbol{\phi}_d, \boldsymbol{\phi}_u^1, \ldots, \phi_u^{n_u}) \tag{5.5}$$

and the controller system C is described as function $\{\boldsymbol{w}, \boldsymbol{\phi}_r, \boldsymbol{\phi}_y\} \in \mathbb{C}^{(n_w+((n_r,n_y)\times N))\times 1} \mapsto \boldsymbol{\phi}_u \in \mathbb{C}^{(n_u\times N)\times 1}$:

$$\begin{aligned} \Phi_C(\boldsymbol{w}, \boldsymbol{\phi}_r, \boldsymbol{\phi}_y): \quad \Phi_F: \boldsymbol{\phi}_{uf} &= \Phi_F(\boldsymbol{w}_F, \boldsymbol{\phi}_r) \\ \Phi_H: \boldsymbol{\phi}_{uh} &= \Phi_H(\boldsymbol{w}_H, \boldsymbol{\phi}_y) \\ \boldsymbol{\phi}_u &= \boldsymbol{\phi}_{uf} + \boldsymbol{\phi}_{uh} \end{aligned} \tag{5.6}$$

Considering LTI systems the extended frequency response functions (FRFs) Φ_G, Φ_H and Φ_F in MIMO cases are assumed as 1st order differentiable functions with respect to their input spectrums. It is also assumed that H and F are 1st order differentiable functions in the frequency domain with respect to their tunable parameters $\boldsymbol{w}$.

In order to describe local linearization in frequency domain with the linear equations, the matrix format of $\Phi_{G'}$, $\Phi_{H'}$ and $\Phi_{F'}$ is to be introduced before proceeding with further discussions on iterative tuning in the frequency domain.

Considering the ANVC system illustrated in Fig. 5.1, the local linearization of $\Phi_{G'}$ can be expressed as

$$\Delta\boldsymbol{\phi}_y = \begin{bmatrix} \Delta\boldsymbol{\phi}_y^1 \\ \vdots \\ \Delta\boldsymbol{\phi}_y^{n_y} \end{bmatrix} = \Phi_{G'}^{(d,u)} \begin{bmatrix} \Delta\boldsymbol{\phi}_d^1 \\ \vdots \\ \Delta\boldsymbol{\phi}_d^{n_d} \\ \Delta\boldsymbol{\phi}_u^1 \\ \vdots \\ \Delta\boldsymbol{\phi}_u^{n_u} \end{bmatrix} \tag{5.7}$$

where $\Phi_{G'}^{(d,u)} := \frac{\mathrm{d}\boldsymbol{\phi}_y}{\mathrm{d}\{\boldsymbol{\phi}_d, \boldsymbol{\phi}_u\}} \in \mathbb{C}^{[n_y\times N]\times[(n_d+n_u)\times N]}$ is the total derivative with respect to $\boldsymbol{\phi}_d$ and $\boldsymbol{\phi}_u$, and given by

$$\Phi_G^{(d,u)} = [\Phi_{G'}^d \quad \Phi_{G'}^u] = \begin{bmatrix} \Phi_G^{(d\mapsto y_1)} & \Phi_G^{(u\mapsto y_1)} \\ \vdots & \vdots \\ \Phi_G^{(d\mapsto y_{n_y})} & \Phi_G^{(u\mapsto y_{n_y})} \end{bmatrix} \tag{5.8}$$

In (5.8) the partial derivatives of G with respect to disturbance $\boldsymbol{\phi}_d$ and control action $\boldsymbol{\phi}_u$ are given by

$$\Phi_G^{(d\mapsto y_i)} := [\Phi_{G'}^{(d_1\mapsto y_i)} \ldots \Phi_{G'}^{(d_j\mapsto y_i)} \ldots \Phi_{G'}^{(d_{n_d}\mapsto y_i)}]^{\mathrm{T}} \in \mathbb{C}^{(n_d\times N)\times N}, j = 1, \ldots, n_d \tag{5.9}$$

$$\Phi_G^{(u\mapsto y_i)} := [\Phi_{G'}^{(u_1\mapsto y_i)} \ldots \Phi_{G'}^{(u_j\mapsto y_i)} \ldots \Phi_{G'}^{(u_{n_u}\mapsto y_i)}]^{\mathrm{T}} \in \mathbb{C}^{(n_u\times N)\times N}, j = 1, \ldots, n_u \quad i = 1, \ldots, n_y \tag{5.10}$$

Here, $\Phi_{G'}^{(d_j \mapsto y_i)} \in \mathbb{C}^{N\times N}$ is the partial derivative matrix of the i-th output spectrum $\boldsymbol{\phi}_y^i$ with respect to j-th disturbance spectrum $\boldsymbol{\phi}_d^j$, $\Phi_{G'}^{(u_j \mapsto y_i)} \in \mathbb{C}^{N\times N}$ is the partial derivative matrix of $\boldsymbol{\phi}_y^i$ with respect to j-th control action spectrum $\boldsymbol{\phi}_u^j$.

As LTI systems are now assumed, it is worthwhile pointing out that the frequency response is independent, and $\Phi_{G'}^{(d_j \mapsto y_i)}$ and $\Phi_{G'}^{(u_j \mapsto y_i)}$ are diagonal matrices.

Since $\boldsymbol{\phi}_d$ is always assumed time invariant during the tuning process and to ease the notation, $\Phi_{G'}^{u}$ is denoted by $\Phi_{G'}$, i.e.,

$$\Phi_{G'} := \frac{\partial \boldsymbol{\phi}_y}{\partial \boldsymbol{\phi}_u} = [\Phi_G^{(u \mapsto y_1)}, \ldots, \Phi_{G'}^{(u \mapsto y_{n_y})}]^{\mathrm{T}} \in \mathbb{C}^{(n_y \times N)\times(n_u \times N)} \tag{5.11}$$

It is straightforward to obtain that

$$\Phi_{H'} := \frac{\partial \boldsymbol{\phi}_{u_h}}{\partial \boldsymbol{\phi}_y} = [\Phi_{H'}^{(y \mapsto u_{h_1})}, \ldots, \Phi_{H'}^{(y \mapsto u_{h_{n_u}})}]^{\mathrm{T}} \in \mathbb{C}^{(n_u \times N)\times(n_y \times N)}, \tag{5.12}$$

$$\Phi_{H'}^{w} := \frac{\partial \boldsymbol{\phi}_{u_h}}{\partial \boldsymbol{w}_H} \in \mathbb{C}^{(n_u \times N)\times(n_{wh})} \tag{5.13}$$

and similarly

$$\Phi_{F'} := \frac{\partial \boldsymbol{\phi}_{u_f}}{\partial \boldsymbol{\phi}_r} = [\Phi_F^{(r, u_{f_1})}, \ldots, \Phi_F^{(r, u_{f_{n_u}})}]^{\mathrm{T}} \in \mathbb{C}^{(n_u \times N)\times(n_r \times N)}, \tag{5.14}$$

$$\Phi_{F'}^{w} := \frac{\partial \boldsymbol{\phi}_{uf}}{\partial \boldsymbol{w}_F} \in \mathbb{C}^{(n_u \times N)\times(n_{wf})}, \tag{5.15}$$

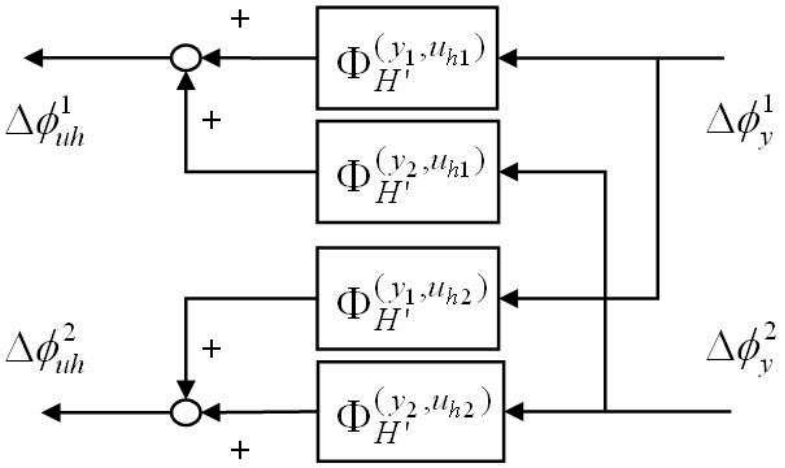

FIGURE 5.2: Block diagram of 2I2O feedback controller in the frequency domain

Fig. 5.2 gives a graphic illustration of one 2-Input/2-Output (2I2O) feedback controller in the frequency domain.

In an LTI system, using the above notations, the increment equation in the frequency domain of G with respect to the change in control action $\boldsymbol{u} = \boldsymbol{u}_f + \boldsymbol{u}_h$ can be written as

$$\Delta\boldsymbol{\phi}_y = \Phi_{G'}^{(u\mapsto y)}(\Delta\boldsymbol{\phi}_{uf} + \Delta\boldsymbol{\phi}_{uh}) \tag{5.16}$$

which is same as (3.15).

5.3 Iterative Tuning in the Frequency Domain for MIMO system

In Section 5.2 the MIMO dynamics is presented in the frequency domain in order to obtain the infinitesimal increment equation of the secondary dynamics $\Phi_{G'}^{(u\mapsto y)}$ as (5.16) that is used to deduct gradient estimate in the frequency domain for MIMO systems.

5.3.1 Gradient estimate in MIMO system

Note that (5.16) in MIMO systems has same increment equation as (3.15) in SISO systems. Considering a finite frequency set Ω it is straightforward to obtain similar results for gradient estimates in the MIMO case:

$$\frac{\partial\boldsymbol{\phi}_y}{\partial\boldsymbol{w}_H}|_\Omega = \Phi_{T'}^{(u\mapsto y)}|_\Omega \frac{\partial\Phi_H(\boldsymbol{\phi}_y|_\Omega, \boldsymbol{w}_H)}{\partial\boldsymbol{w}_H} \tag{5.17}$$

and

$$\frac{\partial\boldsymbol{\phi}_y}{\partial\boldsymbol{w}_F}|_\Omega = \Phi_{T'}^{(u\mapsto y)}|_\Omega \frac{\partial\Phi_F(\boldsymbol{\phi}_r|_\Omega, \boldsymbol{w}_F)}{\partial\boldsymbol{w}_F} \tag{5.18}$$

and

$$\frac{\partial J(\boldsymbol{w})}{\partial\boldsymbol{w}_i} = \frac{2}{N^2}\boldsymbol{\phi}_y^*\Phi_Q\Phi_{T'}^{(u\mapsto y)}|_\Omega \frac{\partial\Phi_C(\boldsymbol{\phi}_y|_\Omega, \boldsymbol{\phi}_r|_\Omega, \boldsymbol{w})}{\partial\boldsymbol{w}_i} \tag{5.19}$$

Although (5.17)-(5.19) for MIMO system are almost same as (3.25)-(3.28) in SISO, there are some differences with the dimensions of the terms in these equations.

- In SISO system, the dimensions of $\boldsymbol{\phi}_y$, $\boldsymbol{\phi}_r$ and $\Phi_{T'}$ are n_Ω, n_Ω and $n_\Omega \times n_\Omega$;
- In MIMO system, the dimensions of $\boldsymbol{\phi}_y$, $\boldsymbol{\phi}_r$ and $\Phi_{T'}$ are $n_y \times n_\Omega$, $n_r \times n_\Omega$ and $(n_y \times n_\Omega) \times (n_u \times n_\Omega)$.

5.3.2 Iterative tuning in the frequency domain in MIMO system

Similar to (3.28), in (5.19) the only unknown term is $\Phi_{T'}$, which is the key to the iterative tuning for MIMO systems too.

Both the direct estimation and the indirect estimation approach are applicable to MIMO systems. As stated in Chapter 4, since indirect estimation is more convenient in control implementations, only indirect estimation is discussed in detail in this subsection.

In the indirect estimation the key point is to estimate $\Phi_{G'}^{(u\mapsto y)}$. Note that $\Phi_{G'}^{(u\mapsto y)}$ has $n_y \times N$ rows and $n_u \times N$ columns, which has $n_y \times N + n_u \times N$ unknown variables to estimate in LTI system.

In the LTI case the frequency response of MIMO systems is independent at different frequencies. Therefore the indirect estimate of $\Phi_{G'}$ is discussed by studying the single frequency response $\Phi_{G'}^{(u\mapsto y)}(\omega) \in \mathbb{C}^{n_y \times n_u}$ in this subsection.

If the spectrum change of the i-th channel control action $u^i(t)$ and i-th channel output $y^i(t)$ at single frequency ω is denoted with $\Delta\phi_u^i(\omega)$ and $\Delta\phi_y^i(\omega)$, respectively, the increment equation of a single frequency response $\Phi_{G'}^{(u\mapsto y)}(\omega)$ caused by the change of control action spectrum $\Delta\boldsymbol{\phi}_u(\omega) := \{\Delta\phi_u^1(\omega), \ldots, \Delta\phi_u^{n_u}(\omega)\}^{\mathrm{T}} \in \mathbb{C}^{n_u \times 1}$ is given by an equation set as:

$$\Delta\boldsymbol{\phi}_y(\omega) = \Phi_{G'}^{(u\mapsto y)}(\omega)\Delta\boldsymbol{\phi}_u(\omega). \tag{5.20}$$

where $\Delta\boldsymbol{\phi}_y(\omega) := \{\Delta\phi_y^1(\omega), \ldots, \Delta\phi_y^{n_y}(\omega)\} \in \mathbb{C}^{n_y \times 1}$.

In (5.20), there are n_y equations, while $\Phi_{G'}^{(u\mapsto y)}(\omega)$ has $n_y \times n_u$ unknown variables to solve. Therefore, given n_u difference pairs $(\Delta\boldsymbol{\phi}_u(\omega) - \Delta\boldsymbol{\phi}_y(\omega))$, n_u equation sets (5.20) can be used to build up a large equation group which has $n_u \times n_y$ equations and can be used to estimate $\Phi_{G'}^{(u\mapsto y)}(\omega)$.

As discussed in Chapter 4 the n_u difference pairs $(\Delta\boldsymbol{\phi}_u(\omega) - \Delta\boldsymbol{\phi}_y(\omega))$ can be obtained through $n_u + 1$ experiments. For example, considering $n_u + 1$ sequential examples, from $k - n_u$-th to k-th experiment, the large equation group is given by

$$\begin{cases} \boldsymbol{\phi}_y^{(k)}(\omega) - \boldsymbol{\phi}_y^{(k-1)}(\omega) &= \hat{\Phi}_{G'}^{(u\mapsto y)}(\omega)[\boldsymbol{\phi}_u^{(k)}(\omega) - \boldsymbol{\phi}_u^{(k-1)}(\omega)] \\ \vdots & \vdots \quad\quad \vdots \\ \boldsymbol{\phi}_y^{(k)}(\omega) - \boldsymbol{\phi}_y^{(k-n_u)}(\omega) &= \hat{\Phi}_{G'}^{(u\mapsto y)}(\omega)[\boldsymbol{\phi}_u^{(k)}(\omega) - \boldsymbol{\phi}_u^{(k-n_u)}(\omega)] \end{cases} \tag{5.21}$$

where $\boldsymbol{\phi}_y^{(i)}(\omega)$ and $\boldsymbol{\phi}_u^{(i)}(\omega), i = k - n_u, \ldots, k$ denote the single frequency ω spectrum of output and control action in the i-th experiment, respectively.

If the $n_u \times n_y$ large equation group built up through $n_u + 1$ experiments is of full-rank, the $\Phi_{G'}^{(u\mapsto y)}(\omega)$ can be estimated and $\hat{\Phi}_{T'}^{(u\mapsto y)}(\omega) \in \mathbb{C}^{n_y \times n_u}$ is computed from

$$\hat{\Phi}_{T'}^{(u\mapsto y)}(\omega) = [I - \hat{\Phi}_{G'}^{(u\mapsto y)}(\omega)\Phi_{H'}^{(y\mapsto u)}(\omega)]^{-1}\hat{\Phi}_{G'}^{(u\mapsto y)}(\omega) \tag{5.22}$$

where I is the n_y-dimension unit matrix and $\Phi_{H'}^{(y\mapsto u)}(\omega) \in \mathbb{C}^{n_u \times n_y}$.

In the SISO case (5.22) was possible to simplify as a single equation

$$\hat{\Phi}_{T'}^{(u\mapsto y)}(\omega) = \frac{\hat{\Phi}_{G'}^{(u\mapsto y)}(\omega)}{1 - \hat{\Phi}_{G'}^{(u\mapsto y)}(\omega)\Phi_{H'}^{(y\mapsto u)}(\omega)}. \tag{5.23}$$

To summarize, under the assumption of a finite frequency set Ω for the disturbance and assuming an LTI system, we have the following tuning strategy in for MIMO system:

Algrithm 5.3.1. At the k-iteration,

1. Estimate $\boldsymbol{\phi}_r^k$, $\boldsymbol{\phi}_u^k$ and $\boldsymbol{\phi}_y^k$ in k-th experiment;
2. Build up a full rank equation group as (5.21) with n_u spectrum difference pairs $\{\Delta\boldsymbol{\phi}_u|_\Omega$, $\Delta\boldsymbol{\phi}_y|_\Omega\}$ through (n_u+1) sequential experiments, and estimate $\Phi_{G'}^{(u\mapsto y)}|_\Omega$ by solving the equation group;
3. Calculate $\Phi_{T'}^{(u\mapsto y)}|_\Omega$ with (5.22);
4. Obtain the derivative of J with (5.19);
5. Update the controller parameter vector $\boldsymbol{w}$ using

$$\boldsymbol{w}^{i+1} = \boldsymbol{w}^i - \mu\nabla J(\boldsymbol{w}^i) \tag{5.24}$$

where μ is a suitable step size to update the controller.

5.3.3 Implementation of FD-IT in MIMO systems

As mentioned above, at least n_u different equation sets as in (5.20) are required to obtain $\Phi_{G'}^{(u\mapsto y)}(\omega)$, which means n_u pairs of difference data $\{\Delta\boldsymbol{u}, \Delta\boldsymbol{y}\}$. In an implementation to estimate $\nabla J(\boldsymbol{w})$ and make one gradient-based tuning update, $1+n_u$ experiments are required to yield n_u pairs of $\{\Delta\boldsymbol{u}, \Delta\boldsymbol{y}\}$ to estimate $\Phi_{G'}^{(u\mapsto y)}|_\Omega$.

Remark 5.3.2. In LTI systems, $\Phi_{G'}^{(u\mapsto y)}|_\Omega$ is considered unchanged and can be estimated off-line or at the beginning of tuning. If $\Phi_{G'}^{(u\mapsto y)}|_\Omega$ is priori unknown, $\Phi_{G'}^{(u\mapsto y)}|_\Omega$ can be estimated in realtime through the first n_u+1 experiments. After then, $\Phi_{T'}^{(u\mapsto y)}|_\Omega(\Phi_{G'}^{(u\mapsto y)}|_\Omega, \Phi_H|_\Omega(\boldsymbol{w}_H^i))$ can be computed with the change of H^i without estimating $\Phi_{G'}^{(u\mapsto y)}|_\Omega$ again.

As shown in Fig. 5.3, to perform N_T times gradient-based tuning, plus n_u initial extra experiments to estimate $\Phi_{G'}^{(u\mapsto y)}|_\Omega$, N_T+n_u times experiments are necessary to be performed.

It is worthwhile to note the realtime estimate of $\Phi_{G'}^{(u\mapsto y)}|_\Omega$, which is useful for tuning Linear Time Variant (LTV) systems. FD-IT can be used to tune slow LTV system as well in which the G changes so slowly that they can be considered as invariant relative to the interval of length $1+n_u$ for one tuning iteration.

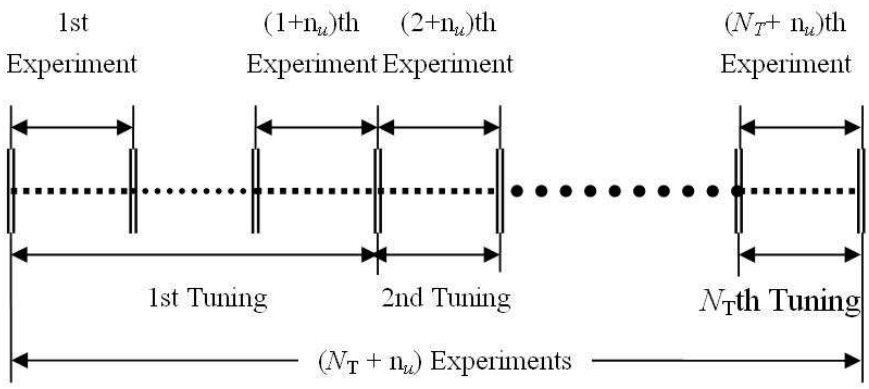

FIGURE 5.3: Tuning procedure of FD-IT for LTI MIMO systems

Remark 5.3.3. At k-th tuning iteration the current secondary dynamics $\Phi_{G'}^{k(u\mapsto y)}|_{\Omega}$ can be estimated with the $1+n_u$ difference data pairs $\{\Delta \boldsymbol{u}, \Delta \boldsymbol{y}\}$ from the $(k-n_u)$-th experiment to the k-th experiment. As shown in Fig. 5.4, in case of slow LTV systems, FD-IT can also perform N_T gradient-based tuning iterations within $N_T + n_u$ experiments.

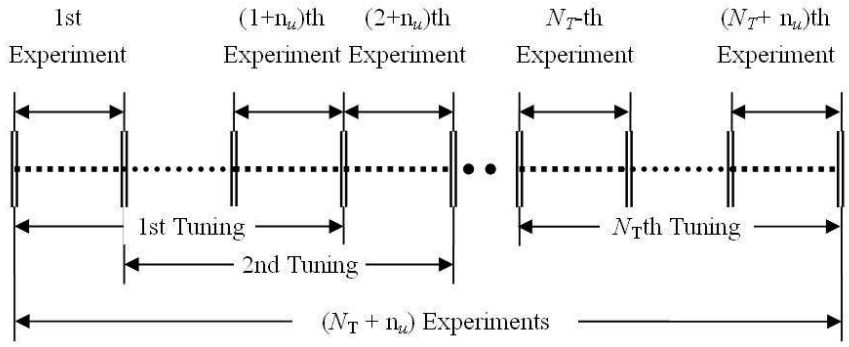

FIGURE 5.4: Tuning procedure of FD-IT for slow LTV MIMO systems

As mentioned above, the obvious advantage of FD-IT over TD-IFT is that FD-IT requires much less experiments to tune than TD-IFT. As stated in [52, 53, 58], to solve the gradient with respect to all possible parameters in full block controllers, $n_u \times n_y$ gradient experiments for the feedback controller H and $n_u \times n_r$ gradient experiments for the feed-forward controller F are required, which needs $(1 + n_u \times n_y + n_u \times n_r)$ experiments in order to compute all gradients for one step tuning. If the blocks of the controller are not independently parameterized, a more elaborate procedure is required [53]. Therefore, in order to perform N_T tuning updates, FD-IT can finished within only $N_T + n_u$ periods as shown in Fig. 5.3 and Fig. 5.4, while TD-IFT requires $N_T \times (1 + n_u \times n_y + n_u \times n_r)$ periods of experiments as shown in Fig. 5.5.

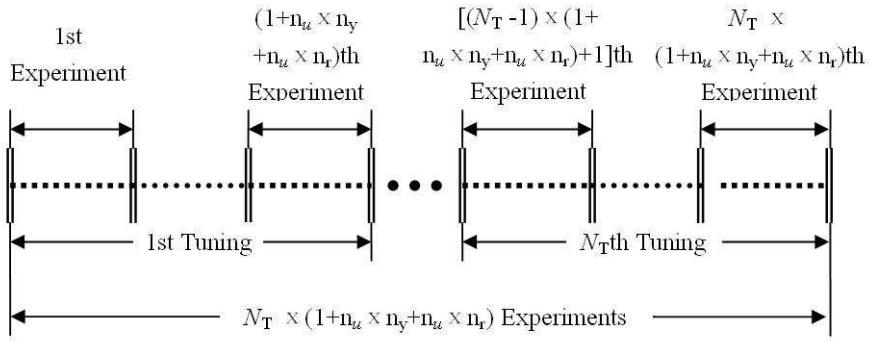

FIGURE 5.5: Tuning procedure of iterative feedback tuning in MIMO system.

5.4 Summary

This chapter extended FD-IT from SISO system to MIMO system.

The derivative matrix of the extended FRF is introduced for MIMO systems. With the derivative matrix the local linearity can be presented as increment equations as (5.16) in MIMO system. Based on (5.16), the deduction procedure is almost same as the procedure for SISO systems, the difference lies with the dimension of the increment equations.

In linear MIMO FD-IT, only one experiment is required to make tuning except that n_u extra experiments that is same as the number of input channels are required at the beginning of the tuning.

Chapter 6

Frequency Domain Iterative Tuning in Nonlinear Systems

This chapter presents an extension of the principles of FD-IT to nonlinear systems. The topic is divided into four parts. In the first part, general ANVC problem including nonlinear dynamics is set up. Next the Generalized Frequency Response Function (GFRF) and its derivative matrix is introduced. In the third section the gradient estimate for nonlinear system is studied in the frequency domain. Finally implementation issues of the FD-IT methodology for nonlinear plants is discussed.

6.1 SISO Nonlinear ANVC problem

In many applications the actual control system is always nonlinear while LTI dynamics is often used by engineers to approximate real systems. In some cases the LTI assumption is accurate enough to describe the essence of the dynamics. However, in a lot of cases nonlinearity is not ignorable and the LTI assumption can lead to significant under performance.

As nonlinear systems are much more common than LTI systems in practical control engineering, the extension of FD-IT to nonlinear systems can be valuable in practice. As mentioned in Chapter 3 the gradient estimate in the frequency domain is still valid for nonlinear plants. To simplify the descriptions of FD-IT for nonlinear systems, first SISO systems are assumed here, the extension to nonlinear MIMO case is straightforward and can be done along similar lines as it was in Chapter 5 for LTI systems.

The general framework illustrated in Fig. 3.1 is still valid to describe nonlinear systems, including most definitions and equations. The two important sets of differences between linear and nonlinear dynamics are as follows:

I Different expressions in the time domain and frequency domain

- In linear systems the dynamics can be written as convolution of Markov weights and the input sequence so that $\boldsymbol{y} = G[\boldsymbol{d}, \boldsymbol{u}]^{\mathrm{T}}$, where function G is not related to the input $\boldsymbol{d}$ and $\boldsymbol{u}$. Similarly, in the frequency domain the FRF of linear dynamics can be written as multiplication between FRF matrix and spectrum vector such as $\boldsymbol{\phi}_y = \Phi_{G_d}\boldsymbol{\phi}_d + \Phi_{G_u}\boldsymbol{\phi}_u$, and the derivative matrix in the frequency domain is identical with the FRF matrix.
- In nonlinear systems the nonlinear dynamics is always represented by a general function such as $\boldsymbol{y} = G(\boldsymbol{d}, \boldsymbol{u})$. In the frequency domain the nonlinear dynamics has more complicated representation of its frequency response, and its derivative matrix in the frequency domain is not necessarily the same as its frequency response matrix.

II Different output considerations

- In in linear systems the disturbance $\boldsymbol{d}$ is periodic, then the steady output $\boldsymbol{y}$ contains the same frequencies as $\boldsymbol{d}$. Therefore the frequency set in output $\boldsymbol{y}$ is invariant while tuning of the controllers is taking place. In the cost function (3.3), the common period N can be determined from the original output and it is kept unchanged during the whole tuning process.
- In a nonlinear system, when the disturbance $\boldsymbol{d}$ is periodic, then the frequencies contained in the steady output $\boldsymbol{y}$ are not necessarily the same as in the disturbance $\boldsymbol{d}$. In general the frequencies contained in output $\boldsymbol{y}$ are varying while the controller is tuned. Therefore, in the cost function (3.3) the common period N may change and should be determined by each tuned output.

Consequently, there will be a slightly modification in the performance criterion in the case of nonlinear ANVC. When the system has steady output $\boldsymbol{y}^k$ in the k-th experiment and it has common period N^k, the cost function of a system as in Fig 3.1 can be defined as the average quadratic performance of a length N^k output sequence:

$$J(\boldsymbol{w}) := \frac{1}{N^k}\sum_{t=0}^{N^k-1} \boldsymbol{y}^{\mathrm{T}}(t)Q\boldsymbol{y}(t) \tag{6.1}$$

where Q is an a priori defined weighting matrix.

The nonlinear ANVC problem can be expressed as an optimization problem as

$$\begin{aligned} \min: \quad & J(\boldsymbol{w}) \text{ in (6.1)}, \\ s.t. \quad & \begin{cases} \text{Eqn. (3.1)} \\ \text{Eqn. (3.2)} \end{cases} \end{aligned} \tag{6.2}$$

Comparing the above mathematical expression (6.2) to the optimization problem in SISO LTI (3.5), they have similar presentation and hence the gradient based method is one of most effective solution for (6.2) as well.

In fact, based on the above general framework shown in Fig. 3.1, the derivation of a gradient estimate in the frequency domain for nonlinear systems is almost the same as that for linear system except some slight modifications of the derivative matrix of FRF in the nonlinear case.

6.2 Derivative Matrix of Generalized FRF in nonlinear systems

In this section the generalized frequency response function (GFRF) [7, 8] is introduced for nonlinear dynamics, which is based on Volterra/Wiener series function [89, 119, 125], and the derivative matrix of GFRF is proposed for iterative tuning in the frequency domain for nonlinear system.

6.2.1 Generalized Frequency Response Function for the Nonlinear System

In this subsection the Generalized Frequency Response Function (GFRF) is introduced to describe nonlinear dynamics and the matrix format of the GFRF is presented.

GFRF is based on Volterra series functional representation from mathematics. The Volterra series functional [149] was first discussed in various publications by V. Volterra since 1887 and finally published in Spanish in 1927. Volterra's representation for nonlinear systems appeared in the work of N. Wiener in the early 1940s as he studied the response of a nonlinear system to white noise inputs. Since the 1950s, several works [4, 11] contained important subsequent work on the Volterra functional representation applied to nonlinear systems.

First the Volterra functional representation of nonlinear discrete time SISO system G is considered. Given an N-length input sequence $\boldsymbol{u} := \{u(0), \ldots, u(N-1)\}$, the output of G is a sequence $\boldsymbol{y} := \{y(0), \ldots, y(N-1)\}$. It can be described with an n-degree Volterra/Wiener series function as [125]:

$$y(t) = h_0 + \sum_{k=1}^{n} \{ \sum_{i_1=0}^{N-1} \cdots \sum_{i_k=0}^{N-1} h_k(i_1, \ldots, i_k) \prod_{j=0}^{k} u(t - i_j) \} \tag{6.3}$$

where the k-th Volterra kernel $h_k(i_1, \ldots, i_k)$ is real and equals to zeros if any argument is negative.

In [125], the sufficient but not necessary condition to guarantee the convergence of (6.3) is stated as:

$$\sum_{i_1=0}^{N-1} \cdots \sum_{i_k=0}^{N-1} |h_k(i_1, \ldots, i_k)| < \infty \tag{6.4}$$

The steady response to sinusoidal inputs in nonlinear systems can be studied through the Volterra/Wiener approach. Generally, unlike LTI system, in nonlinear systems there is no one-to-one mapping in the frequency domain. The presentation of nonlinear response in the frequency domain is often called Generalized Frequency Response Function (GFRF) in most of papers.

The n-th order GFRF is the n-fold Fourier transform of the n-th order Volterra kernels:

$$H_n(\omega_1, \ldots, \omega_n) = \sum_{i_1=0}^{N-1} \cdots \sum_{i_n=0}^{N-1} \{h_n(i_1, \ldots, i_n) e^{-j(\omega_1 + \ldots + \omega_n)}\} \tag{6.5}$$

In the probing method [5] the input signal is given by

$$u(t) = \sum_{m=1}^{n_\omega} A_m e^{j\omega_m t}, \tag{6.6}$$

and this is injected to probe the plant to reveal information in the frequency domain.

As stated in [151], the output of the system described by (6.3) can be expressed as

$$\phi_y(t) = \sum_{k=1}^{n} \sum_{I} \sum_{II} \{H_k(j\omega_1, \ldots, j\omega_k)(\prod_{i=1}^{n_k} A_i) e^{j(\omega_1 + \ldots + \omega_k)t}\} \tag{6.7}$$

where $\sum_I$ denotes the summation of all combinations of taking k frequencies from n_ω frequencies at a time, and $\sum_{II}$ denotes the summation of all the permutations of $\omega_1, \ldots, \omega_k$.

Note that in (6.7) the output $\boldsymbol{y}$ can also be expressed as the combination of several frequencies. If the kernel functions of GFRF $H_{\{\cdot\}}(\cdot)$ are assumed to be 1st order differentiable, then it is straightforward to derive the matrix format of 1st order derivative functions of GFRF using (6.7).

For example, considering a nonlinear system described by a 2nd order Volterra function as (6.3), the input signal is

$$u(t) = A_1 e^{j(\omega_1 t + \alpha_1)} + A_2 e^{j(\omega_2 t + \alpha_2)}. \tag{6.8}$$

It is then straightforward to write the output $y(t)$ as

$$\begin{aligned} y(t) \quad = \quad & H_0 + H_1(\omega_1) A_1 e^{j(\omega_1 t + \alpha_1)} + H_1(\omega_2) A_2 e^{j(\omega_2 t + \alpha_2)} \\ & + H_2(\omega_1, \omega_1) A_1^2 e^{j2(\omega_1 t + \alpha_1)} + H_2(\omega_2, \omega_2) A_2^2 e^{j2(\omega_2 t + \alpha_2)} \\ & + 2H_2(\omega_1, \omega_2) A_1 A_2 e^{j((\omega_1 + \omega_2)t + \alpha_1 + \alpha_2)} \end{aligned} \tag{6.9}$$

Unlike an LTI system that would only have two frequencies of ω_1 and ω_2, the nonlinear output $\boldsymbol{y}$ includes frequencies 0 (offset item), ω_1, ω_2, $2\omega_1$, $2\omega_2$ and $\omega_1+\omega_2$.

Considering the infinitesimal increment of the $\boldsymbol{\phi}_u$ as

$$u(t) = A_1 e^{j(\omega_1 t+\alpha_1)} + \Delta A_1 e^{j(\omega_1 t+\Delta\alpha_1)} + A_2 e^{j(\omega_2 t+\alpha_2)} + \Delta A_2 e^{j(\omega_2 t+\Delta\alpha_2)}, \tag{6.10}$$

the increment of $y(t)$ can be locally linearized with 1st order derivatives as

$$\begin{aligned} \Delta y(t) \quad = \quad & (H_1(\omega_1)e^{j\omega_1 t} + 2H_2(\omega_1,\omega_1)A_1 e^{j2\omega_1 t} \\ & +2H_2(\omega_1,\omega_2)A_2 e^{j(\omega_1+\omega_2)t})\Delta A_1 e^{j(\omega_1 t+\Delta\alpha_1)} \\ & +(H_1(\omega_2)e^{j\omega_2 t} + 2H_2(\omega_2,\omega_2)A_2 e^{j2\omega_2 t} \\ & +2H_2(\omega_1,\omega_2)A_1 e^{j(\omega_1+\omega_2)t})\Delta A_2 e^{j(\omega_2 t+\Delta\alpha_2)} \end{aligned} \tag{6.11}$$

6.2.2 Derivative matrix of GFRF

In the above subsection the 1st order derivative function of GFRF in the time domain is illustrated with a 2-frequency signal input, which is presented as a local linearization format as (6.11). In this subsection the matrix format of the derivative function of GFRF for nonlinear systems is studied in the frequency domain.

To ease the presentation let us start from the previous 2nd order Volterra function example described by (6.8)and (6.9). Considering single frequency spectrum $\boldsymbol{\phi}_y(\omega_k) := A_k e^{j\alpha_k}$ in the frequency domain, the amplitude is A_k and phase angle is α_k. Therefore, in the previous example, the input $u(t)$ can be expressed in the frequency domain with non-zero spectrum as $\boldsymbol{\phi}_u := \{\phi_u(\omega_1), \phi_u(\omega_2)\}$, and $y(t)$ can be expressed as $\boldsymbol{\phi}_y := \{\phi_y(0), \phi_y(\omega_1), \phi_y(\omega_2), \phi_y(2\omega_1), \phi_y(2\omega_2), \phi_y(\omega_1+\omega_2)\}^{\mathrm{T}}$.

Denoting the finite frequency sets $\Omega_{\Delta u} := \{\omega_1, \omega_2\}$ and $\Omega_{\Delta y} := \{0, \omega_1, \omega_2, 2\omega_1, 2\omega_2, \omega_1+\omega_2\}$, $\Delta\boldsymbol{\phi}_u|_{\Delta u} := \{\Delta\phi_u(\omega_1), \Delta\phi_u(\omega_2)\}^{\mathrm{T}}$ and $\Delta\boldsymbol{\phi}_y|\Delta y := \{\Delta\phi_y(\omega_1), \Delta\phi_y(\omega_2), \Delta\phi_y(2\omega_1), \Delta\phi_y(2\omega_2), \Delta\phi_y(\omega_1+\omega_2)\}^{\mathrm{T}}$, Eqn. (6.11) can be expressed as the matrix format in the frequency domain as

$$\Delta\boldsymbol{\phi}_y|_{\Delta y} = \Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_{\Delta y}\}}\Delta\boldsymbol{\phi}_u|_{\Delta u}, \tag{6.12}$$

where $\Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_{\Delta y}\}}$ denotes the matrix of 1st order partial derivative GFRF with respect to the spectrum of $\boldsymbol{u}$, i.e.,

$$\Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_{\Delta y}\}} = \begin{bmatrix} H_1(\omega_1)e^{j\omega_1 t} & 0 \\ 0 & H_1(\omega_2)e^{j\omega_2 t} \\ 2H_2(\omega_1,\omega_1)A_1e^{j2\omega_1 t} & 0 \\ 0 & 2H_2(\omega_2,\omega_2)A_2e^{j2\omega_1 t} \\ 2H_2(\omega_1,\omega_2)A_2e^{j(\omega_1+\omega_2)t} & 2H_2(\omega_1,\omega_2)A_1e^{j(\omega_1+\omega_2)t} \end{bmatrix} \tag{6.13}$$

According to (6.7) the extension of the above derivation from the 2-frequency input case to the general multi-tone input case is straightforward and the matrix $\Phi_{G'}$ in the multi-tone input case is of similar format as (6.13).

Consider the nonlinear dynamics G for which the spectrum of input $\boldsymbol{u}$ and output $\boldsymbol{y}$ are $\boldsymbol{\phi}_u = \{\phi_u(\omega_{u1}),\ldots,\phi_u(\omega_{un_{\Omega_u}})\}$ and $\boldsymbol{\phi}_y = \{\phi_y(\omega_{y1}),\ldots,\phi_y(\omega_{yn_{\Omega_y}})\}$, are defined at the frequency sets of input and output: $\Omega_u := \{\omega_{u1},\ldots,\omega_{un_{\Omega_u}}\}$ and $\Omega_y := \{\omega_{y1},\ldots,\omega_{yn_{\Omega_y}}\}$. Similarly, given a small increment of $\boldsymbol{\phi}_u$ in $\Omega_{\Delta u}$, i.e., $\Delta\boldsymbol{\phi}_u|_{\Omega_{\Delta u}}$, the local linearized increment of $\boldsymbol{\phi}_y$ in $\Omega_{\Delta y}$, i.e., $\Delta\boldsymbol{\phi}_y|_{\Omega_{\Delta y}}$, can be expressed in a format as (6.12), where the derivative matrix of GFRF, $\Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_{\Delta y}\}} \in \mathbb{C}^{yn_{\Omega_y}\times un_{\Omega_u}}$, is given as

$$\Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_{\Delta y}\}} := \begin{bmatrix} \Phi_{G'}(\omega_{u1},\omega_{y1}) & \cdots & \Phi_{G'}(\omega_{un_{\Omega_u}},\omega_{y1}) \\ \vdots & \ddots & \vdots \\ \Phi_{G'}(\omega_{u1},\omega_{yn_{\Omega_{\Delta y}}}) & \cdots & \Phi_{G'}(\omega_{un_{\Delta u}},\omega_{yn_{\Delta y}}) \end{bmatrix} \tag{6.14}$$

where $\Phi_{G'}(\omega_{uk},\omega_{yl}), uk = u1,\ldots,un_{\Omega_{\Delta u}}, yl = y1,\ldots,yn_{\Omega_{\Delta y}}$ is the local linearized coefficient of the mapping from $\Delta\boldsymbol{\phi}_u(\omega_{uk})$ to $\Delta\boldsymbol{\phi}_y(\omega_{yl})$.

Compared with the diagonal FRF matrix for linear systems, the derivative matrix of GFRF for nonlinear systems is different in four points:

1. While the derivative matrix of FRF, $\Phi_{G'}$, is always identical with the FRF matrix, Φ_G, in a linear system, $\Phi_{G'}$ is always not same as Φ_G in a nonlinear system.

2. While $\Phi_{G'}$ is invariant with respect to the input spectrum $\boldsymbol{\phi}_u$ in a linear system, $\Phi_{G'}$ in (6.14) is a local linearized coefficient matrix and variant with respect to the input spectrum $\boldsymbol{\phi}_u$.

3. While $\Phi_{G'}$ is always a square matrix in a linear system for $\boldsymbol{\phi}_y$ has identical frequencies with $\boldsymbol{\phi}_u$, $\Phi_{G'}$ is generally a non-square matrix in a nonlinear system for $\Omega_{\Delta u}$ is always different from $\Omega_{\Delta y}$.

4. While $\Phi_{G'}$ is a diagonal matrix in a linear system, there exist non-zero off-diagonal items in (6.14), which is caused by the cross-coupling among the different frequencies in the input signal and leads to the different frequency set in output from the input.

Using (6.14), the local linearization of G at $\{\boldsymbol{\phi}_{uh}, \boldsymbol{\phi}_{uf}, \boldsymbol{\phi}_y\}$ can be written with the infinitesimal increment equation as (3.15):

$$\Delta\boldsymbol{\phi}_y|_{\Omega_{\Delta y}} \approx \Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_{\Delta y}\}}(\Delta\boldsymbol{\phi}_{uh}|_{\Omega_{\Delta u}} + \Delta\boldsymbol{\phi}_{uf}|_{\Omega_{\Delta u}}). \tag{6.15}$$

6.3 Iterative Tuning in the Frequency Domain in Nonlinear System

In section 6.2, nonlinear dynamics is presented with GFRF in the frequency domain in order to deliver the increment equation of local linearized secondary dynamics $\Phi_{G'}^{(u\mapsto y)}$ as (6.15). In this section, iterative tuning in the frequency domain is studied for nonlinear system based on the increment equation.

6.3.1 Gradient estimate in nonlinear system

Note that the increment equation (6.15) for nonlinear systems has the same format as (3.15) for LTI systems. Using (6.15), local linearization can be performed in nonlinear systems using infinitesimal increments analysis.

As illustrated in Fig. 3.3, the derivation of the gradient estimate in the frequency domain for nonlinear systems is similar to that in Chapter 3. Considering the full spectrum it is straightforward to get similar results on the gradient estimate in the nonlinear case as the results in linear case:

$$\frac{\partial\boldsymbol{\phi}_y}{\partial\boldsymbol{w}_H} = \Phi_{T'}\frac{\partial\Phi_H(\boldsymbol{\phi}_y, \boldsymbol{w}_H)}{\partial\boldsymbol{w}_H} \tag{6.16}$$

and

$$\frac{\partial\boldsymbol{\phi}_y}{\partial\boldsymbol{w}_F} = \Phi_{T'}\frac{\partial\Phi_F(\boldsymbol{\phi}_r, \boldsymbol{w}_F)}{\partial\boldsymbol{w}_F} \tag{6.17}$$

and

$$\frac{\partial J(\boldsymbol{w})}{\partial\boldsymbol{w}_i} = \frac{2}{N^2}\boldsymbol{\phi}_y^*\Phi_Q\Phi_{T'}\frac{\partial\Phi_C(\boldsymbol{\phi}_y, \boldsymbol{\phi}_r, \boldsymbol{w})}{\partial\boldsymbol{w}_i} \tag{6.18}$$

However, when considering finite frequency set signals, i.e. a discrete spectrum, there is an important difference for nonlinear systems relative to LTI systems. Given $\boldsymbol{\phi}_r$,$\boldsymbol{\phi}_u$ and $\boldsymbol{\phi}_y$ with finite frequency sets Ω_r, Ω_u and Ω_y respectively, (6.18) can be rewritten as

$$\frac{\partial J(\boldsymbol{w})}{\partial\boldsymbol{w}_i} = \frac{2}{N^2}\boldsymbol{\phi}_y^*|_{\Omega_y}\Phi_Q\Phi_{T'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}\frac{\partial\Phi_C(\boldsymbol{\phi}_y|_{\Omega_y}, \boldsymbol{\phi}_r|_{\Omega_r}, \boldsymbol{w})}{\partial\boldsymbol{w}_i} \tag{6.19}$$

where $\Omega_{\Delta u}$ denotes the frequency set of the change of the control action spectrum $\Delta\boldsymbol{\phi}_u$ caused by the change of control parameters $\Delta\boldsymbol{w}$ at the $\{\boldsymbol{\phi}_y|_{\Omega_y}, \boldsymbol{\phi}_r|_{\Omega_r}, \boldsymbol{w}\}$.

Compared with (3.33), (6.19) has three important difference.

1. In a nonlinear system, the partial closed dynamics $\Phi_{T'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ is presented as

$$\Phi_{T'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}} = (I - \Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}\Phi_{H'}|_{\{\Omega_y\mapsto\Omega_{\Delta u}\}})^{-1}\Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}} \tag{6.20}$$

   which is derived from

$$\begin{aligned}\Delta\boldsymbol{\phi}_y|_{\Omega_y} \;=\; & \Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}[\frac{\partial\Phi_F(\boldsymbol{\phi}_r|_{\Omega_r},\boldsymbol{w}_F)}{\partial\boldsymbol{w}_F}\Delta\boldsymbol{w}_F \\ & +\frac{\partial\Phi_H(\boldsymbol{\phi}_y,\boldsymbol{w}_H)}{\partial\boldsymbol{w}_H}\Delta\boldsymbol{w}_H + \frac{\partial\Phi_H(\boldsymbol{\phi}_y|_{\Omega_y},\boldsymbol{w}_H)}{\partial\boldsymbol{\phi}_y}|_{\{\Omega_y\mapsto\Omega_{\Delta u}\}}\Delta\boldsymbol{\phi}_y|_{\Omega_y}];\end{aligned} \tag{6.21}$$

   In a linear system, the partial closed dynamics is $\Phi_T|_{\Omega_y} = (I - \Phi_G|_{\Omega_y}\Phi_H|_{\Omega_y})^{-1}\Phi_G|_{\Omega_y}$.

2. In (6.19) for a nonlinear system, Ω_r, Ω_u and $\Omega_{\Delta u}$ are often different from Ω_y; In (3.33) for LTI system, Ω_r, Ω_u and $\Omega_{\Delta u}$ are always identical with Ω_y and all substituted with Ω in (3.33).

3. In (6.19) for a nonlinear system, $\Phi_{T'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ is a $(n_{\Omega_y}\times n_{\Omega_{\Delta u}})$-dimension matrix; In (3.33) for LTI system, $\Phi_{T'}|_{\Omega_y}$ is a n_{Ω_y}-dimension diagonal square matrix.

6.3.2 Iterative tuning in the frequency domain in nonlinear system

In (6.19) controller C is known by the designer, $\frac{\partial\Phi_C(\boldsymbol{\phi}_y|_{\Omega_y},\boldsymbol{\phi}_r|_{\Omega_r},\boldsymbol{w})}{\partial\boldsymbol{w}_i}$ and $\Delta\phi_u$ caused by $\Delta\boldsymbol{w}$ is computable given $\{\phi_y|_{\Omega_y},\boldsymbol{\phi}_r|_{\Omega_r},\boldsymbol{w}\}$. Like (3.28), the only unknown item is $\Phi_{T'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$, which is the key in the iterative tuning in the frequency domain in nonlinear system as well.

In the indirect estimate approach the problem of estimating $\Phi_{T'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ turns out to be estimating $\Phi_{G'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ through experiments and then computing $\Phi_{T'}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ with (6.20).

Like the analysis of FD-IT for MIMO systems in the previous chapter, given one spectrum difference pair $\{\Delta\boldsymbol{\phi}_u|_{\Omega_{\Delta u}},\Delta\boldsymbol{\phi}_y|_{\Omega_y}\}$, one infinitesimal increment equation set is given by

$$\Delta\boldsymbol{\phi}_y|_{\Omega_y} \approx \Phi_{G'}^{u\mapsto y}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}\Delta\boldsymbol{\phi}_u|_{\Omega_{\Delta u}} \tag{6.22}$$

which has n_{Ω_y} equations.

Since $\Phi_{G'}^{u\mapsto y}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ has $n_{\Omega_y}\times n_{\Omega_{\Delta u}}$ unknown variables, $n_{\Omega_{\Delta u}}$ such equation sets as (6.22) are required to build up a large equation group to solve $\Phi_{G'}^{u\mapsto y}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$.

As discussed in the previous section, the $n_{\Omega_{\Delta u}}$ difference pairs $\{\Delta\boldsymbol{\phi}_u,\Delta\boldsymbol{\phi}_y\}$ can be obtained through $(n_{\Omega_{\Delta u}}+1)$ experiments. For example, considering $(n_{\Omega_{\Delta u}}+1)$ sequential examples, from $k-n_{\Omega_{\Delta u}}$-th to k-th experiment, if the change of control action is properly small, the large equation group is given as:

$$\begin{cases} \boldsymbol{\phi}_y^{(k)}|_{\Omega_y} - \boldsymbol{\phi}_y^{(k-1)}(\omega)|_{\Omega_y} & = & \hat{\Phi}_{G'}^{(u\mapsto y)}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}[\boldsymbol{\phi}_u^{(k)}|_{\Omega_{\Delta u}} - \boldsymbol{\phi}_u^{(k-1)}|_{\Omega_{\Delta u}}] \\ \vdots & \vdots & \vdots \\ \boldsymbol{\phi}_y^{(k)}|_{\Omega_y} - \boldsymbol{\phi}_y^{(k-n_{\Omega_{\Delta u}})}(\omega)|_{\Omega_y} & = & \hat{\Phi}_{G'}^{(u\mapsto y)}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}[\boldsymbol{\phi}_u^{(k)}|_{\Omega_{\Delta u}} - \boldsymbol{\phi}_u^{(k-n_{\Omega_{\Delta u}})}|_{\Omega_{\Delta u}}] \end{cases} \tag{6.23}$$

which has $(n_{\Omega_{\Delta u}} \times n_{\Omega_y})$ equations to solve $\hat{\Phi}_{G'}^{(u\mapsto y)}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$.

If the $(n_{\Omega_{\Delta u}} \times n_{\Omega_y})$ large equation group built up through $n_u + 1$ experiments is full-rank, $\Phi_{G'}^{(u\mapsto y)}(\omega)$ can be estimated and $\hat{\Phi}_{T'}^{(u\mapsto y)}(\omega) \in \mathbb{C}^{n_y \times n_u}$ is computable with (6.20).

To summarize, the tuning strategy in the frequency domain for a nonlinear system is given as:

Algrithm 6.3.1. At the kth iteration the Ω_r^k, Ω_u^k, Ω_y^k and $\Omega_{\Delta u}^k$ can be vary during the tuning process.

1. Estimate $\boldsymbol{\phi}_r^k$, $\boldsymbol{\phi}_u^k$ and $\boldsymbol{\phi}_y^k$, compute $\Omega_{\Delta u}^k$ through $\frac{\partial \Phi_C(\boldsymbol{\phi}_y^k, \boldsymbol{\phi}_r^k, \boldsymbol{w}^k)}{\partial \boldsymbol{w}}$;
2. Build up a full rank equation group as (6.23) with $n_{\Omega_{\Delta u}}^k$ spectrum difference pairs $\{\Delta\boldsymbol{\phi}_u, \Delta\boldsymbol{\phi}_y\}$ through $(n_{\Omega_{\Delta u}^k} + 1)$ sequential experiments, and estimate $\Phi_{G'}^{(u\mapsto y)}|_{\{\Omega_{\Delta u}^k\mapsto\Omega_y^k\}}$ by solving the equation group;
3. Calculate $\Phi_{T'}^{(u\mapsto y)}|_{\{\Omega_{\Delta u}^k\mapsto\Omega_y^k\}}$ with (6.20);
4. Solve the derivative of J with (6.19);
5. Update the controller parameter $\boldsymbol{w}$ with

$$\boldsymbol{w}^{i+1} = \boldsymbol{w}^i - \mu\nabla J(\boldsymbol{w}^i) \tag{6.24}$$

where μ is a proper step size to update the controller.

6.4 Implementation issues of FD-IT in nonlinear system

First of all at least $n_{\Omega_{\Delta u}}$ different equation sets as (6.15) are required to obtain $\Phi_{G'}^{(u\mapsto y)}|_{\Omega_{\Delta u}}$, which means that $n_{\Omega_{\Delta u}}$ pairs of difference data $\{\Delta\boldsymbol{u}, \Delta\boldsymbol{y}\}$ are required. In the implementation, in order to estimate $\nabla J(\boldsymbol{w})$ and make one gradient-based tuning, $1 + n_{\Omega_{\Delta u}}$ experiments are required to yield $n_{\Omega_{\Delta u}}$ pairs of $\{\Delta\boldsymbol{\phi}_u, \Delta\boldsymbol{\phi}_y\}$ to estimate $\Phi_{G'}^{(u\mapsto y)}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$.

In nonlinear systems $\Phi_{G'}^{k(u\mapsto y)}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ generally varies during the tuning process. Hence, in order to estimate the performance gradient $\nabla J(\boldsymbol{w})$, the unknown secondary path $\Phi_{G'}^{(u\mapsto y)}|_{\{\Omega_{\Delta u}\mapsto\Omega_y\}}$ has to be online estimated in every tuning iteration. At the same time it should be pointed out that the number of required extra experiments, i.e., $n_{\Omega_{\Delta u}^k}$, in each tuning iterations is always different. A typical tuning procedure of FD-IT for nonlinear systems is illustrated in Fig. 6.1.

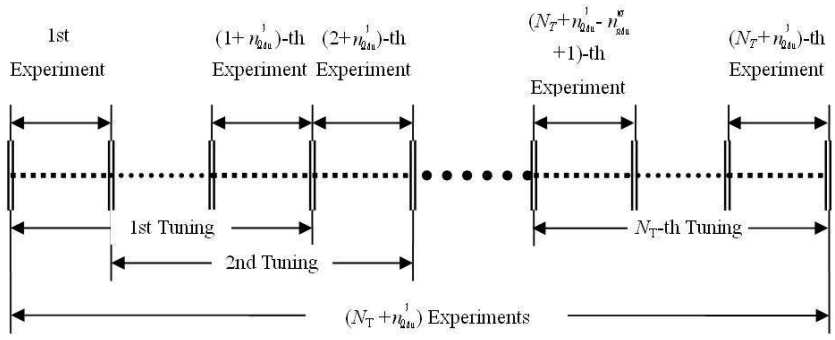

FIGURE 6.1: Tuning procedure of iterative tuning in the frequency domain in nonlinear system

As shown in Fig. 6.1, the initial tuning iteration requires $(n_{\Omega^1_{\Delta u}} + 1)$ experiments manually performed (or automated), where the manually set control parameters, i.e., $\boldsymbol{w}^1, \ldots, \boldsymbol{w}^{n_{\Omega^1_{\Delta u}}}$, should stabilize the ANVC system, and the change among these parameters should be properly small to satisfy the local linearization condition. After then, the subsequent k-th tuning iteration can use its $n_{\Omega^k_{\Delta u}}$ sequential experiments as extra experiments to estimate $\nabla J(\boldsymbol{w}^k)$. In the tuning procedure shown in Fig. 6.1, it is assumed that the subsequent $\{n_{\Omega^k_{\Delta u}}, k > 1\}$ is not greater than the number of its previous experiments and no extra manual experiment is required after initial tuning iterations. In this case, in order to perform N_T gradient-based tuning, $(N_T + n_{\Omega^0_{\Delta u}})$ experiments are required.

Secondly, $\Omega_{\Delta u}$ is often very small due to the infinitesimal increment condition. If Δu is directly computed through $\Delta \boldsymbol{u} = \boldsymbol{u}^{k+1} - \boldsymbol{u}^k$, it is very sensitive with to noise in sensors and actuators. In practice, when $H(\boldsymbol{w}_H)$, $F(\boldsymbol{w}_F)$ and $\boldsymbol{r}$, $\boldsymbol{y}$ are known, $\boldsymbol{\phi}_{\Delta u}$ can be solved with a more robust method as

$$\boldsymbol{\phi}_{\Delta u} = \Phi_H(\boldsymbol{w}_H^{k+1}, \boldsymbol{\phi}_y^{k+1}) - \Phi_H(\boldsymbol{w}_H^k, \boldsymbol{\phi}_y^k) + \Phi_F(\boldsymbol{w}_F^{k+1}, \boldsymbol{\phi}_r^{k+1}) - \Phi_F(\boldsymbol{w}_F^k, \boldsymbol{\phi}_r^k) \tag{6.25}$$

where $\boldsymbol{\phi}_{\Delta u}$ is determined by the main frequencies of output Ω_y and reference Ω_r.

Specifically, if $H(\boldsymbol{w}_H)$ and $F(\boldsymbol{w}_F)$ are linear controllers, $\Omega_{\Delta u} = \Omega_y \cup \Omega_r$.

Finally, it is worthwhile to point out that there is a trade-off in the step size μ between performance and robustness. μ must be sufficiently small so that the high order terms in Taylor expansions can be neglected to perform local linearization in (6.21). On the other hand, too small μ yields a poor signal to noise ratio (SNR). When the nonlinearity is not significant then larger μ can be used. When the system's SNR is high and the nonlinearity is notable, some small μ should be adopted. Sometimes, some extra manual experiments to change control actions $\boldsymbol{u}^k$ slightly are required to perform local linearization when the nonlinearity is notable.

6.5 Summary

This chapter extended the gradient estimation method and iterative tuning in the frequency domain from the linear case to the nonlinear case.

The derivative matrix format of Generalized FRF is introduced for nonlinear systems. With this generalized derivative matrix the local linearization can be presented by infinitesimal increment equations as (6.15) in nonlinear systems. Based on (6.15), the derivation procedure is almost the same as the derivation procedure in the linear case and some differences lie in the dimensions of the increment equations.

To implement FD-IT for a nonlinear system some more extra experiments are required at the beginning of tuning. $\Omega_{\Delta u}$ can be computed with (6.25). The trade-off in the choice of the step size is clarified.

Chapter 7

Simulation Work

In this chapter SIMULINK-based simulations are presented to demonstrate FD-IT in ANVC problems. The simulation work consists of three parts. Firstly, FD-IT in SISO LTI system is illustrated with a simulated duct system; Secondly, FD-IT is illustrated in a simulated 2-Input-2-Output system; Finally, FD-IT for a nonlinear system is illustrated with saturation nonlinearity and dead-zone nonlinearity.

7.1 Simulation Work for SISO LTI System

This section illustrates the usefulness of the FD-IT in a SISO LTI system. FIR-FD-IT and FSF-FD-IT are tested and compared. The robustness of FD-IT against errors in common period is also illustrated through simulation experiments.

7.1.1 Simulation platform

The SIMULINK block diagram of ANVC for a SISO LTI system is given in Fig. 7.1, which is used in [138] to test IFT in ANVC. It comprises a feedback controller H and a feed-forward controller F.

The digital control system sampling rate is 4 kHz. The discrete-time plant model is an 8th-order transfer function with an additional delay of 14 sampling periods:

$$\begin{aligned} G(z) \quad = \quad & z^{14}\frac{0.417z - 0.3989z^2 + 0.023z^3 + 0.0232z^4}{1 - 1.6675z + 0.8358z^2 - 0.0182z^3 - 0.0165z^4} \\ & \frac{-0.0201z^5 + 0.0178z^6 - 0.3608z^7}{-0.0133z5 - 0.0076z^6 - 0.0003z^7} \\ & \frac{+0.5728z^8}{-0.0525z^8}\boldsymbol{u} + \boldsymbol{d} \end{aligned} \tag{7.1}$$

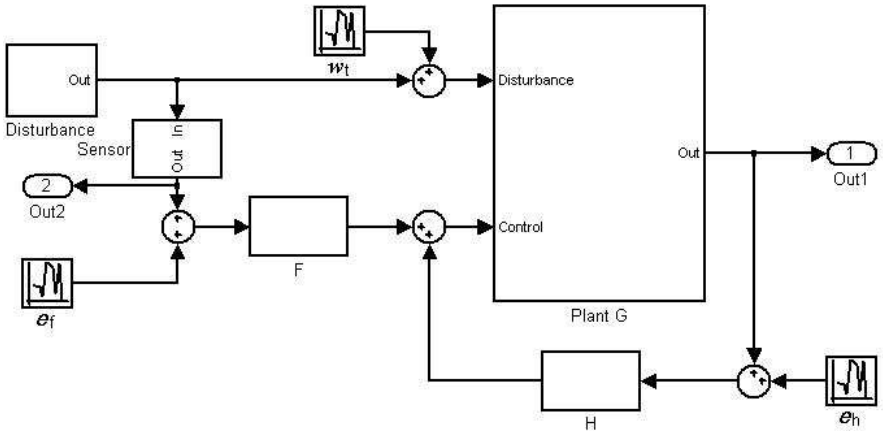

FIGURE 7.1: Block diagram of ANVC for a SISO LTI system

The disturbance signal is a mix of three sine-waves with frequencies of 100Hz, 160Hz and 250Hz and a white noise signal w_t with variance 0.01, leading to:

$$d(t) = \frac{1}{3}(\sin(2\pi 100t) + \sin(2\pi 160t) + \sin(2\pi 250t)) + w_t \tag{7.2}$$

as showed in Fig. 7.2.

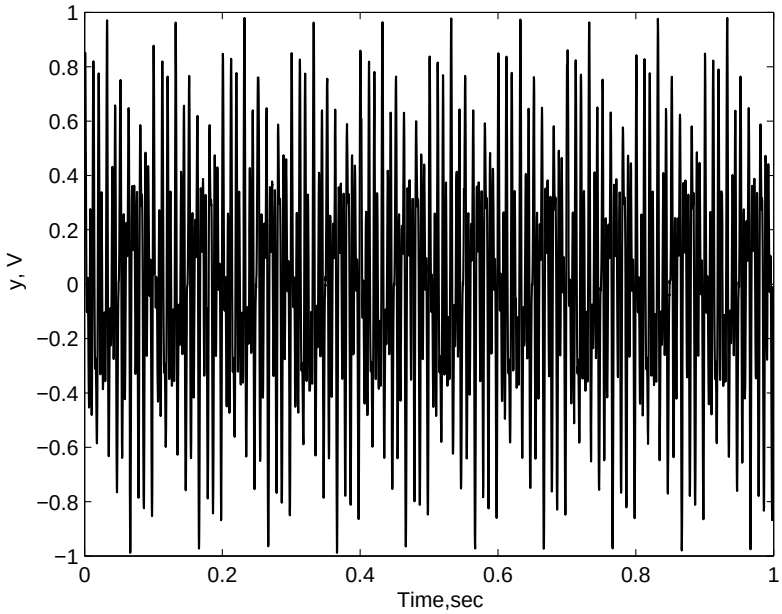

FIGURE 7.2: Initial output without control in SISO ANVC

The reference signal $r(t)$ is obtained from $d(t)$ by:

$$r(t) = 0.4d(t-10) + e_f \tag{7.3}$$

where e_f is a sensor white noise with variance 0.001.

There is sensor white noise e_h with variance 0.001 as well. According to off-line or online frequency estimate, the minimal common period of disturbance is 400 samples. In the following SISO simulation tests the common period is set as 800, i.e., $N = 800$.

In ANVC, in order to satisfy the stability demand and high-level of cancellation, a large-order finite impulse response (FIR) controller is generally required [138, 142, 165]. Two type of controllers are tested in the simulation platform (Fig. 7.1). One is the Finite Impulse Response (FIR) controllers, another is the Frequency-Selective-Filtered (FSF) controllers, which is abbreviated as FIR-FD-IT and FSF-FD-IT, respectively.

In the following SISO simulations all the initial controllers are set to zero. As shown in Fig.7.2, the initial performance criterion without control is 0.1668.

7.1.2 Simulation result of FIR-FD-IT

The first SISO simulation tests FIR controllers in FD-IT. The feedback controller is realized with a 10-th order FIR structure and the feed-forward controller with a 60-th order FIR structure. The step size for the feedback controller H is $\mu_h = 0.02$ and the step size for feed-forward controller F is $\mu_f = 0.2$.

After 47 tuning iterations, the performance is 0.00015 with 30dB cancellation. The final output $\boldsymbol{y}$ after tuning are illustrated in Fig. 7.3.

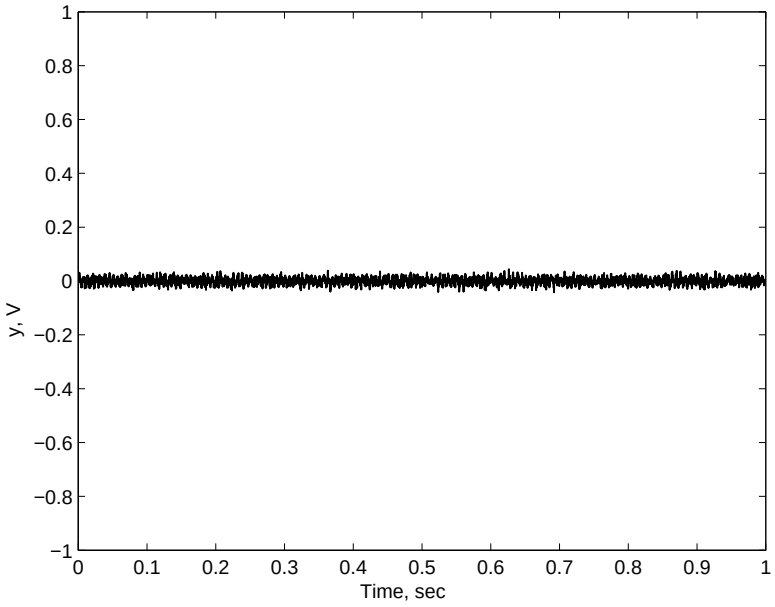

FIGURE 7.3: Final output after tuning in SISO FIR-FD-IT

Fig. 7.5 illustrates performance updating during tuning process. In Fig. 7.5, the decrease of the performance criterion is stable after the 2nd experiment when the gradient based tuning works and the final cancelation is satisfactory.

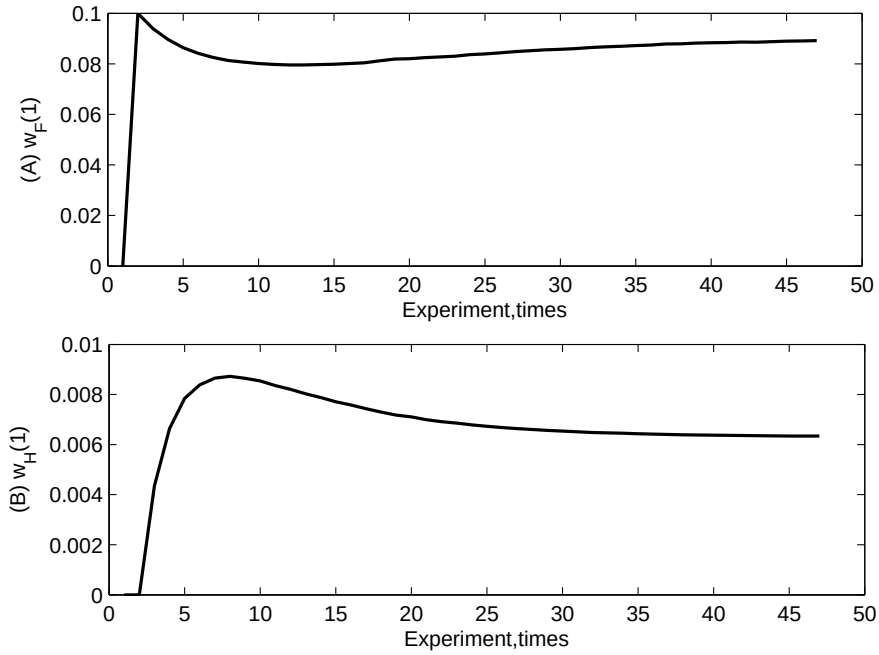

FIGURE 7.4: Controller parameter's updating during tuning process in SISO FIR-FD-IT, (A) updating of the coefficient of z in FIR in feed forward path, (B)updating of the coefficient of z in FIR in feedback path

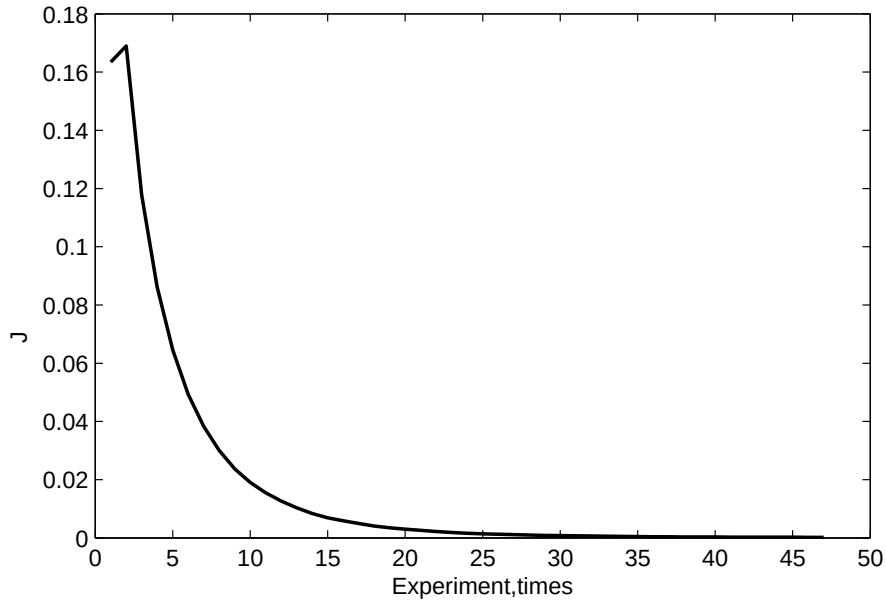

FIGURE 7.5: Performance updating during tuning process in FIR-FD-IT

Fig. 7.4 illustrates the updating of control parameter $w_F(1)$ and $w_H(1)$'s (i.e., the coefficient of z in FIR filter of F and H) during tuning process. The second experiment is a manual extra

experiment when the feed-forward controller F is manually set as $\boldsymbol{u}_f = 0.1\boldsymbol{r}$.

In the simulation tests of FIR-FD-IT, from around $\mu_f = 0.4$, oscillating phenomena begin to happen, and from around $\mu_h = 0.025$, instability begins during the tuning process.

7.1.3 Simulation result of FSF-FD-IT

The second SISO simulation tests FSF controllers in FD-IT. Fig.7.6 illustrates FSF feed-forward controller group for 100Hz, 160Hz and 250Hz, which is used in module F in Fig.7.1. The cascaded tunable modules 'F***' are 11-th FIR controllers. The feedback controller module H has similar structure except that the cascaded tunable modules 'H***' are 3-rd FIR controllers. In this simulation test the step size for feedback controller H is $\mu_h = 0.5$ and the step size for feed-forward controller F is $\mu_f = 2.5$.

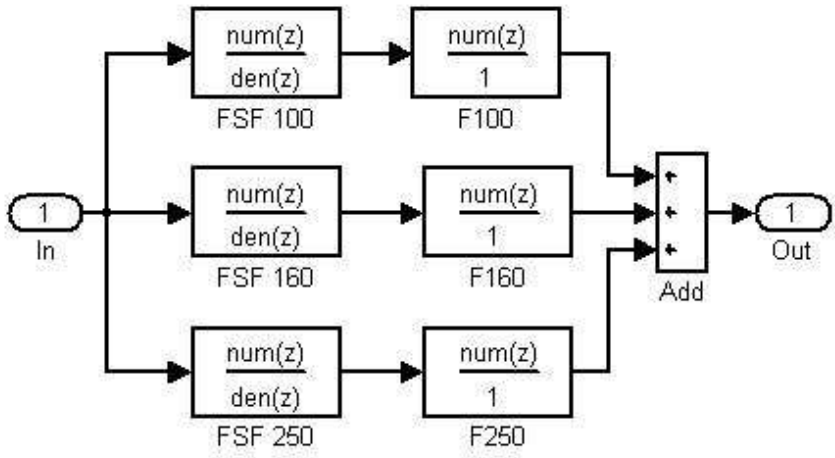

FIGURE 7.6: FSF feed-forward controller group

After 40 tuning iterations, the performance is 0.00011 with 32dB cancelation. The final output $\boldsymbol{y}$ after tuning are illustrated in Fig. 7.7 .

Fig. 7.8 illustrates the updating of control parameter $w_F(1)$ and $w_H(1)$'s (i.e., the coefficient of z in three FIR filters of three FSF channels in F and H) during tuning process. The second experiment is a manual extra experiment when the feed-forward controller F is manually set as $\boldsymbol{u}_f = 0.05\boldsymbol{r}$.

Fig. 7.9 illustrates the performance updating during tuning process where the decrease of the performance criterion is stable after the 2nd experiment when the gradient based tuning works and the final cancellation is satisfactory.

In the simulation tests of FSF-FD-IT, from around $\mu_f = 5.0$, oscillating phenomena begin to happen, and from around $\mu_h = 0.9$, instability begins to happen during the tuning process.

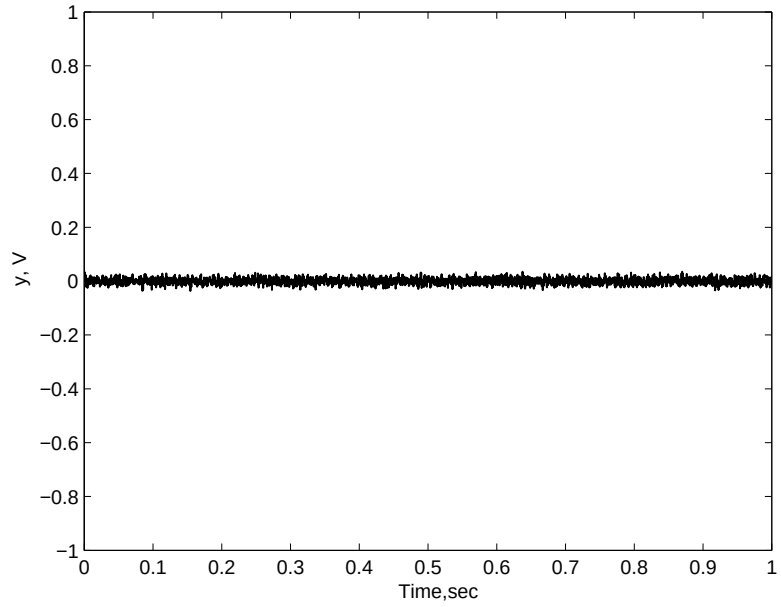

FIGURE 7.7: Final output after tuning in FSF-FD-IT

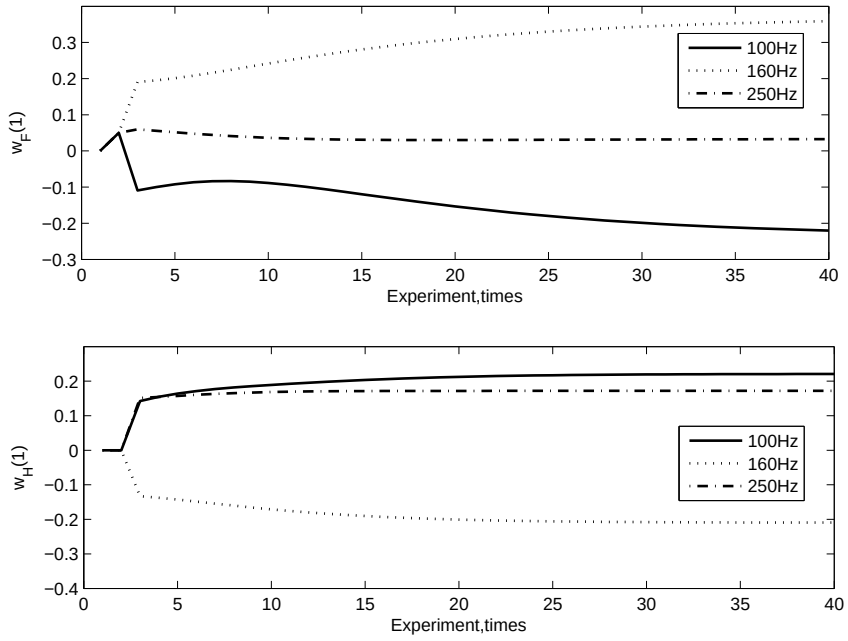

FIGURE 7.8: Controller parameter's updating during tuning process in SISO FSF-FD-IT, (A) updating of the coefficient of z in three FIR for 100Hz, 160Hz and 250Hz frequencies FSF channels in feed forward path, (B)updating of the coefficient of z in FIR for 100Hz, 160Hz and 250Hz frequencies FSF channels in feedback path

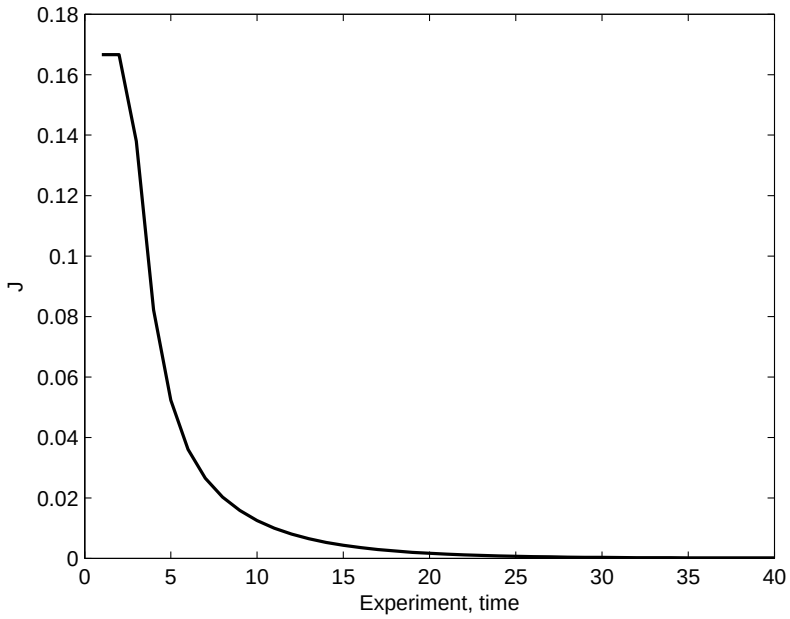

FIGURE 7.9: Performance updating during tuning process in FSF-FD-IT

7.1.4 Comparison of simulation results

In this subsection, the comparison of results of FSF-FD-IT, FIR-FD-IT and TD-IFT [138] simulation is performed to illustrate the simplicity and effectiveness of FSF-FD-IT.

Firstly, the simulation work of FD-IT is compared to that of TD-IFT. In Chapter 6 of [138] a detailed simulation work is presented to test FIR based TD-IFT and FSF based TD-IFT, which is based on the simulated duct system as 7.1. Since they are both gradient based tuning methods, the final tuning performance is almost same. In FIR based TD-IFT, the final cancelation is around 30dB, and in FSF based TD-IFT, the final cancelation is 32dB, which is almost same tuning results as FIR-FD-IT and FSF-FD-IT, respectively.

However, there are two important advantages of FD-IT over TD-IFT.

- According to Fig. 7.1, the diagram of control system for FD-IT is very simple which only requires some measurement path to acquire experimental data. On the other hand to implement TD-IFT the auxiliary path to inject extra signals is necessary.
- As shown in Fig. 7.5 and Fig. 7.9, only one extra experiment is required, which is the non-gradient-based tuning in the second experiment. To implement FIR based FD-IT, there are three stages in one iteration.

Through the simulation tests in the above two subsections, comparing FSF-FD-IT to FIR-FD-IT, it is clear that FSF-FD-IT has advantages over FIR-FD-IT in two respects:

1. FSF-FD-IT is more stable in tuning than FIR-FD-IT. While FIR-FD-IT gives stable tuning when $\mu_h < 0.025$ and $\mu_f < 0.4$, FSF-FD-IT gives stable tuning when $\mu_h < 0.9$ and $\mu_f < 5.0$. This is caused by the introduction of Frequency-Selective-Filter group. While FIR controller is of "all-pass" filter in character, the stability of FIR-FD-IT has to be considered in the complete spectrum. On the other hand, through introducing FSF filters, the stability of FSF-FD-IT is mainly determined by the the closed dynamics in the finite frequencies Ω_y.

2. FSF-FD-IT is more efficient in tuning than FIR-FD-IT. While FIR-FD-IT has 60-th FIR in F and 10-th FIR in H, and requires 47 iterations to achieve 30dB cancelation, FSF-FD-IT has 11-th tunable FIR in F and 3rd tunable FIR in H , and spends 40 iterations to achieve 32dB cancellation.

   The reason is straightforward. In FIR-FD-IT, given n_F-th order FIR controller F and n_H-th order FIR controller H, the tuning is to perform minimization optimization in the n_F-dimensional space for F and n_H-dimensional space for H. In FSF-FD-IT, FSFs can split the whole solution space into independent n_Ω sub-spaces since the FRF is independent in the LTI system. Each sub-space is n_F-dimensional for the FIR tunable controllers in F and n_H-dimensional for the FIR tunable controllers in H. The tuning is performed by multiple minimization sub-optimization in these sub-spaces independently. While $\{n_F, n_H\}$ in FSF-FD-IT is always much smaller than $\{n_F, n_H\}$ in FIR-FD-IT, the minimization optimization is efficient in lower dimensional space, i.e., the tuning is more efficient in FSF-FD-IT.

7.1.5 Robustness against error in the common period

Iterative tuning in the frequency domain is a self-tuning approach completely based on a frequency domain approach. Hence its robustness against errors in the fundamental period of the system is one of the most important factors affecting its robustness.

An important spectrum error is the error in the common period N. FD-IT shows some robustness against the error of estimation of the common period N too.

Given that the actual common period of the disturbance is $N = 800$, some estimation error N_e of the estimated common period are generated so that $\hat{N} = N + Ne$. A series of simulation experiments based on FSF-FD-IT are performed while $\hat{N}$ is changing. Fig. 7.10 gives the change of final performance after 20 iterations when the estimated common period $\hat{N}$ changes from 795 to 805:

Fig. 7.11 illustrates a typical tuning performance updating curve when $\hat{N} = 795$, which gives 13dB cancellation after 20 iterations.

According to above simulation result, in SISO system, FSF-FD-IT presents good robustness against errors in the common period N.

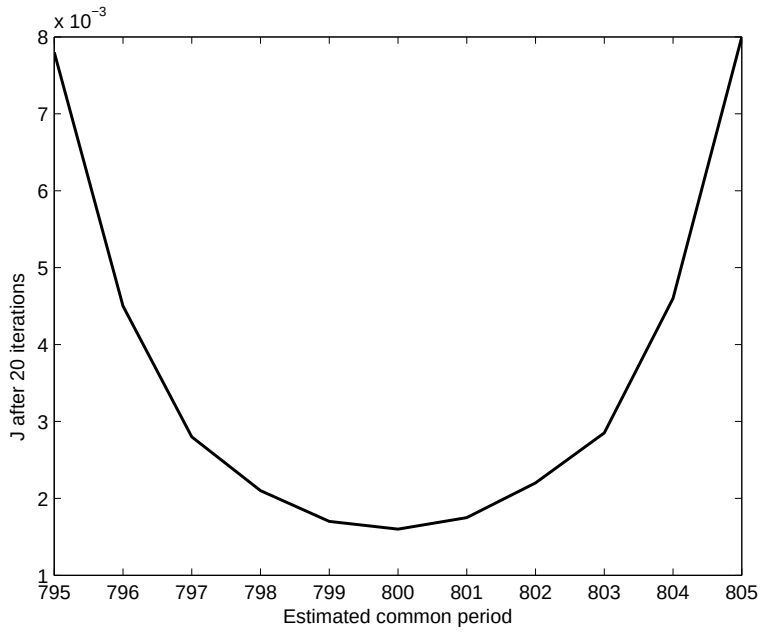

FIGURE 7.10: Change of tuning result with respect to $\hat{N}$

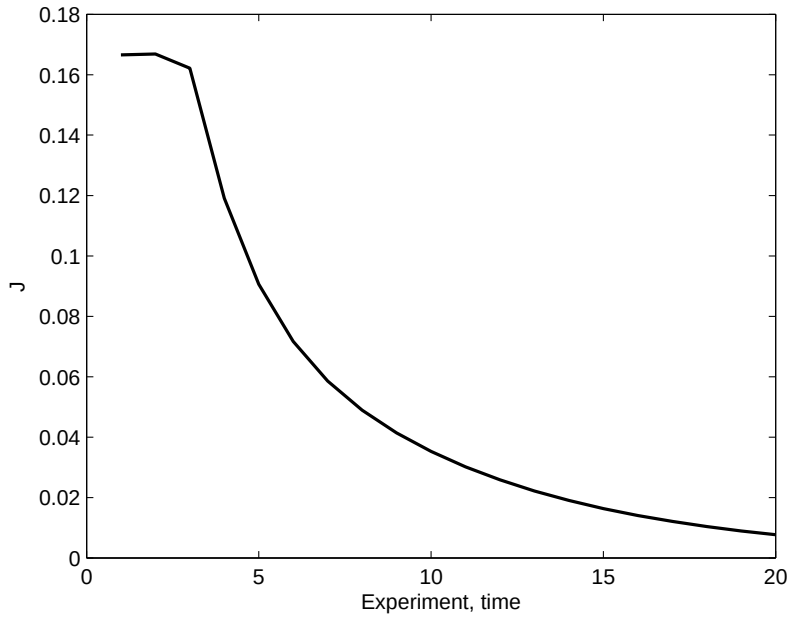

FIGURE 7.11: Performance updating when $\hat{N} = 795$

7.2 Simulations on MIMO Systems

This section illustrates the effectiveness of FD-IT in MIMO system as tested in simulation using MATLAB. FIR based and FSF-based control structures are tested and compared. The robustness against the error of the common period N is also discussed on a simulation example.

7.2.1 Simulation platform

The block diagram of a SIMULINK-based simulation is given in Fig. 7.12. It is a 2-input and 2-output LTI system. $y1$, $y2$, $r1$ and $r2$ denote the data acquired for output and reference signals. Module $Ny1$, $Ny2$, $Nr1$ and $Nr2$ denote the senor noise in the output and reference paths. They are assumed as white noise with standard deviation 0.001.

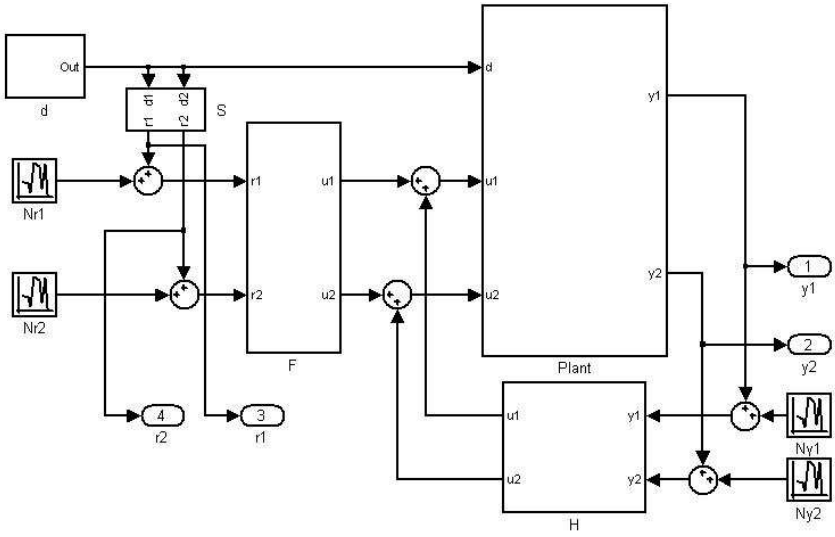

FIGURE 7.12: Block diagram of MIMO ANVC in SIMULINK

Figure 7.13 illustrates the block diagram of the unknown plant G.

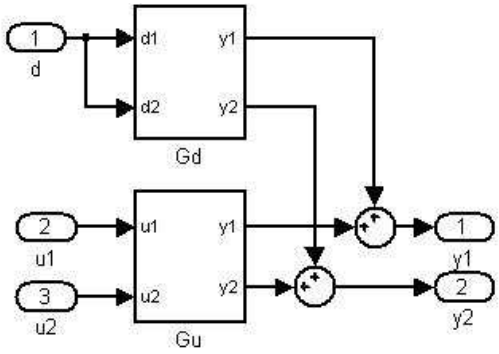

FIGURE 7.13: Block diagram of MIMO Plant G

In [94], a 2-Input and 2-Output linear dynamics is used to simulate the secondary path G_u to illustrate the usefulness of Model-Free Frequency-Domain-Tuning (MF-FDT), which is given as

$$G_u(z) = \begin{bmatrix} \frac{0.1z^8-0.3z^9}{1+0.2z-0.2z^2} & \frac{0.01z^6-0.03z^7}{1+0.02z+0.01z^2} \\ \frac{-0.02z^7-0.02z^8}{1+0.01z-0.01z^2} & \frac{-0.2z^8-0.3z^9}{1+0.1z-0.2z^2} \end{bmatrix} \tag{7.4}$$

To compare the tuning results between MF-FDT and FD-IT, it is also used in the platform illustrated in Fig. 7.13. The disturbance path G_d are given by and

$$G_d(q) = \begin{bmatrix} \frac{0.85z^4}{1+0.4z} & 0 \\ 0 & \frac{0.95z^6}{1-0.2z^2} \end{bmatrix} \tag{7.5}$$

The sampling frequency is $4k$Hz. The disturbance signal $\boldsymbol{d}$ is a mix of three sine-waves with frequencies of 50Hz, 80Hz and 100Hz and a white noise signal w_t with standard deviation 0.01 leads to:

$$d(t) = \frac{1}{3}[\sin(100\pi t) + \sin(80\pi(t-0.091)) + \sin(50\pi t)] + w_t \tag{7.6}$$

In the above system the uncontrolled initial output is shown in Fig. 7.14.

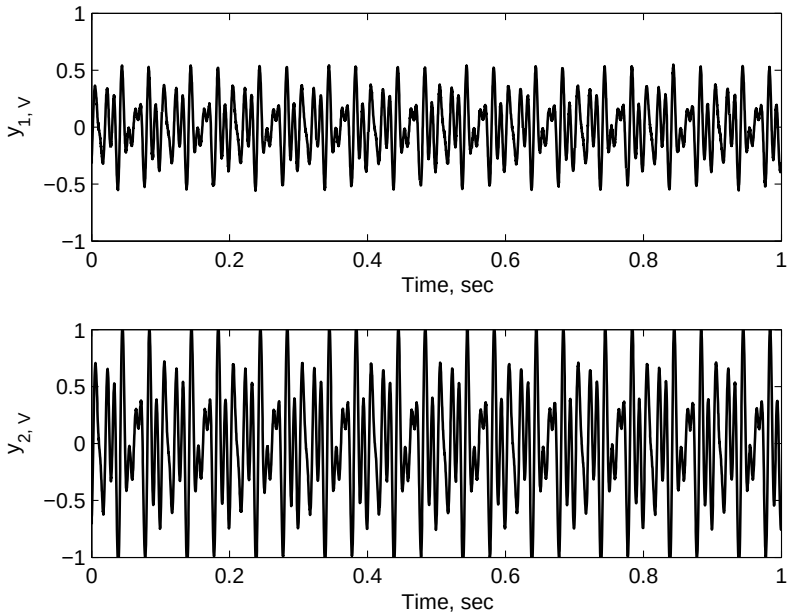

FIGURE 7.14: Initial output without control in MIMO ANVC

The reference signal $r(t)$ is obtained from $d(t)$ by S:

$$S(q) = \begin{bmatrix} \frac{0.8z^8}{1+0.8z} & 0 \\ 0 & \frac{0.5z^{10}}{1+0.9z} \end{bmatrix} \tag{7.7}$$

7.2.2 Simulation for FIR-FD-IT and FSF-FD-IT in MIMO system

In this subsection, two formats of the controller structure are tested with the above simulated platform. One is an FIR controller structure (FIR-FD-IT), another is an FSF based controllere structure (FSF-FD-IT).

FIR-FD-IT has the 10-th order feedback controller and the 40-th order feed-forward controller. The step size (adaptation gain) for feed-forward controller tuning is $\mu_f = 0.6$ and step size for feedback controller tuning is $\mu_h = 0.15$.

In FSF-FD-IT, 1st-order Butterworth bandpass filters are online designed according to the online estimated spectrum of $\boldsymbol{y}$, and the tunable modules use 2nd FIR format filters. The bandwidths of the FSF were given by the disturbance frequency ± 10 percent which also eliminates the unwanted white noise in the tuning. The step size (adaptation gain) for feed-forward controller tuning is $\mu_f = 10.0$ and step size for feedback controller tuning is $\mu_h = 2.5$.

When applying FIR-FD-IT, after 50 iterations, the final performance is $J = 0.0371$ with 8.2dB cancellation. The final output with control is shown in Fig. 7.15.

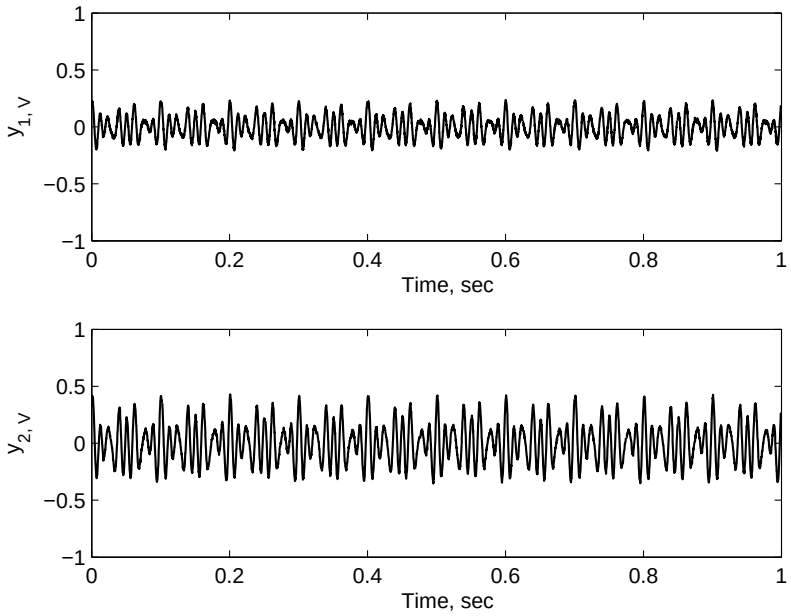

FIGURE 7.15: Final output of FIR-FD-IT in MIMO ANVC

During tuning process, the 2nd and 3rd iteration are manual updates, where all feed forward controller H is set as zeros except that sub-path of $F11$ and $F21$ is set as 0.1, respectively. The

updating of controller parameters $w_F(1)$ and $w_H(1)$ (i.e., the coefficient of z in FIR filters in F and H) along tuning process is illustrated as Fig. 7.16.

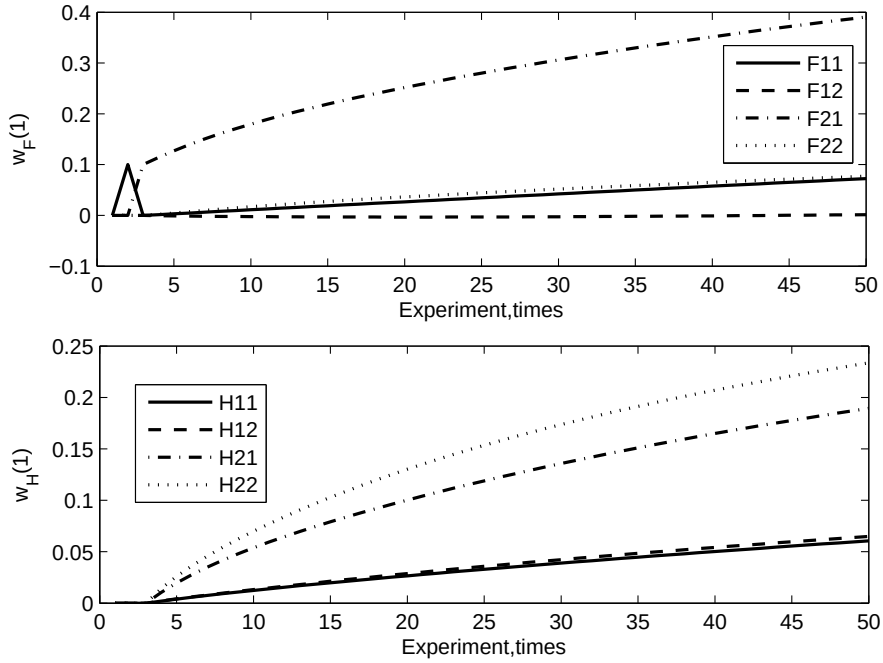

FIGURE 7.16: Controller parameter's updating during tuning process in MIMO FIR-FD-IT, (A) updating of the coefficient of z of FIR in four sub-pathes in the feed forward path, (B)updating of the coefficient of z of FIR in four sub-pathes in the feedback path.

When applying FSF-FD-IT, after 50 iterations, the final performance is $J = 0.0014$ with 22.4dB cancelation. The final output with control is shown in Fig. 7.18.

When updating controller parameters in MIMO FSF-FD-IT, ZPK format is adopted to produce more accurate dynamics, i.e., the parallelized FSF filters are intergraded as one ZPK format filter. Fig. 7.19 illustrates the updating of the integrated gains in four sub-pathes in F and H during tuning process. The 2nd and 3rd experiment is a manual extra experiment when the feed-forward controller F is manually set as gain 0.2.

Fig. 7.20 shows the performance updating of FSF-FD-IT along tuning process. Compared to MF-FDT in [94] which has around 15dB cancelation, FSF-FD-IT produce almost same control result.

On the other hand, as shown in the performance updating curves in Fig. 7.17 and Fig. 7.20, FD-IT in the above two simulations has 2 manual-set experiments, i.e., the 2nd and 3rd experiments, all other iterations are gradient-based tuning. Compared to MF-FDT in [94] which has 3 stages in each iteration and requires an additional path to inject extra signals, FD-IT is much more convenient to implement in practice.

Compared with FIR-FD-IT, FSF-FD-IT in MIMO systems can supply much better tuning performance than FIR-FD-IT in MIMO as well. The reason is straightforward according to the

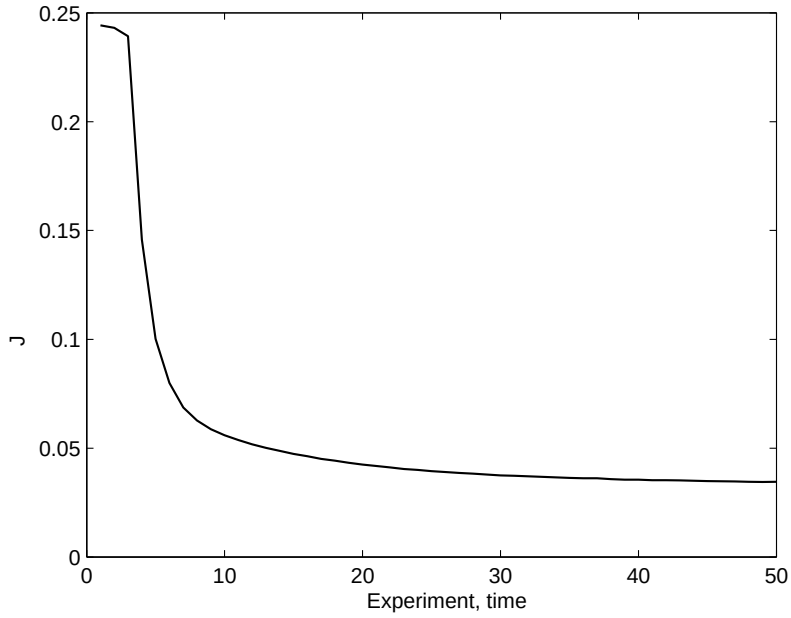

FIGURE 7.17: Performance update in FIR-FD-IT in MIMO ANVC

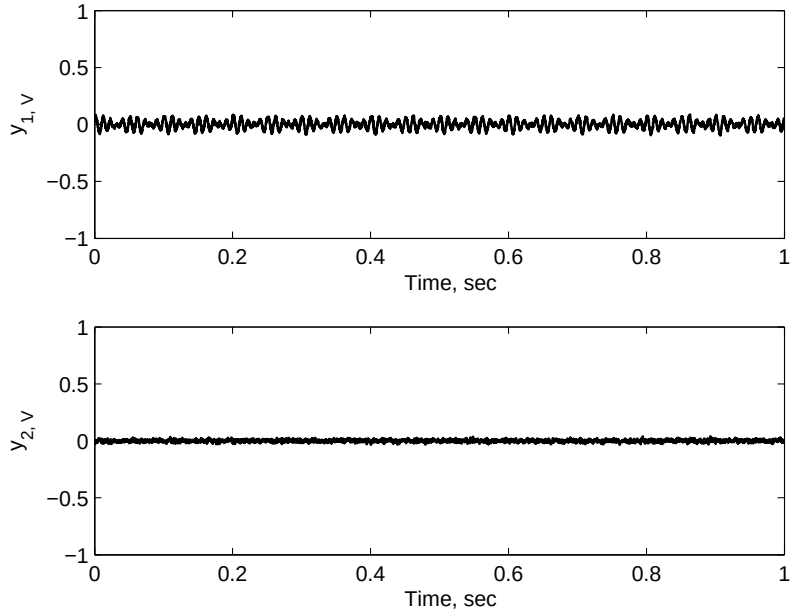

FIGURE 7.18: Final output of FSF-FD-IT in MIMO ANVC

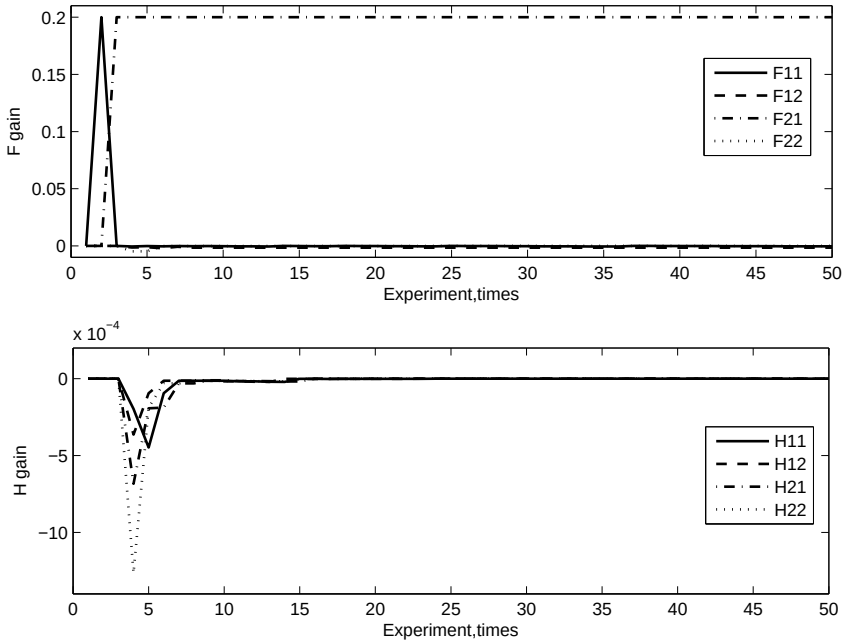

FIGURE 7.19: Controller parameter's updating during tuning process in MIMO FSF-FD-IT, (A) updating of the integrated gains in four sub-pathes in the feed forward path, (B)updating of the integrated gains in four sub-pathes in the feedback path.

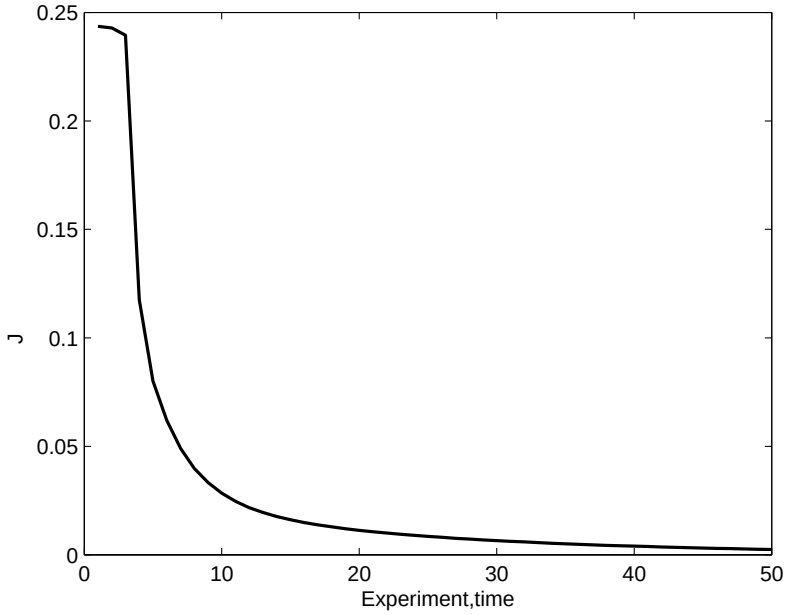

FIGURE 7.20: Performance update in FSF-FD-IT in MIMO ANVC

explanation for SISO case in the pervious section.

7.2.3 Simulation for the robustness against the error of N

Simulation is performed to test the robustness of FD-IT against the error in the common period N in MIMO system. To test the robustness against errors of N, a series of simulations based on FSF-FD-IT are performed with varying N.

Some estimation error N_e of the estimated common period N usually occurs that will be denoted by $\hat{N} := N + Ne$. A test with a series of simulation experiments are performed while the exact common period is $N = 800$ and $\hat{N}$ is changed from 795 to 805. There are 4 experiments for each $\hat{N}$. In these simulation experiments, $\mu_f = 4.0$ and $\mu_h = 1.0$.

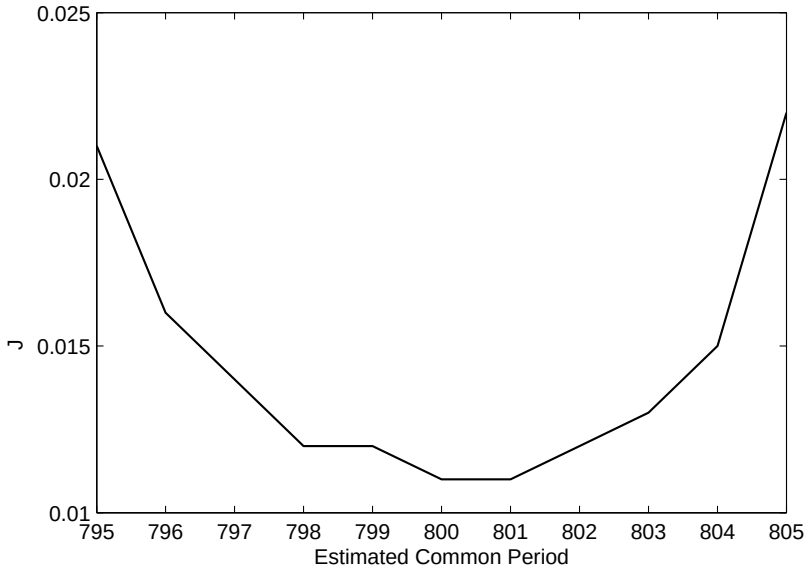

FIGURE 7.21: Change of final performance with respect to $\hat{N}$, while the true common period $N = 800$

Fig. 7.21 gives the change of the average final performance after 40 iterations when the estimated common period $\hat{N}$ changes from 795 to 805:

According to the simulation results, FSF-FD-IT is robust against errors in the estimate of common period around the true N.

7.3 Simulation Work for a Nonlinear System

In this section, simulated nonlinear platform is used to test the applicability of FD-IT for nonlinear ANVC problems.

7.3.1 Simulation platform

In [138], a linear simulated model was used as a linear ANVC example, which is also used for previous SISO LTI simulation tests. As shown in Fig. 7.22, in order to test the practical application and compare the different performance of FSF-FD-IT in the linear and nonlinear cases, the same linear simulated duct plant is selected and pre-cascaded with a nonlinear module to simulate nonlinearity of actuators.

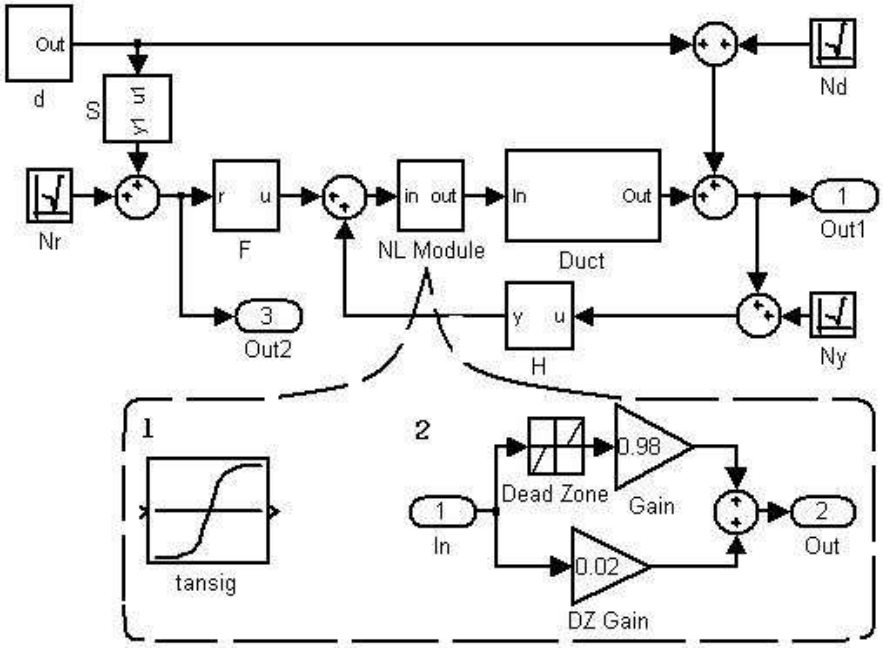

FIGURE 7.22: Block diagram for simulation

The sampling rate of the digital control systems is 4 kHz. As shown in Fig. 7.22, the secondary path comprises two parts. The first part is a LTI module which is the same as the module used in SISO simulation example as in Eqn. 7.1; another part is a pre-filtered nonlinear module which presents the nonlinearity of actuators.

Two common nonlinearities, i.e. saturation and dead zone, in the actuator path are simulated. In the first simulation, a 'tansig' module marked '1' in Fig. 7.22 is pre-cascaded before G which described the saturation. In the second simulation, the dead zone nonlinearity is simulated by the block marked '2' in Fig. 7.22.

The disturbance signal is a mix of three sine-waves with frequencies of 100Hz, 160Hz and a white noise signal N_d with variance 0.01, leading to:

$$d(t) = 0.7[\sin(2\pi 100(t + 0.07)) + \sin(2\pi 160t)] + N_d, \tag{7.8}$$

which has spectrum shown in Fig. 7.23.

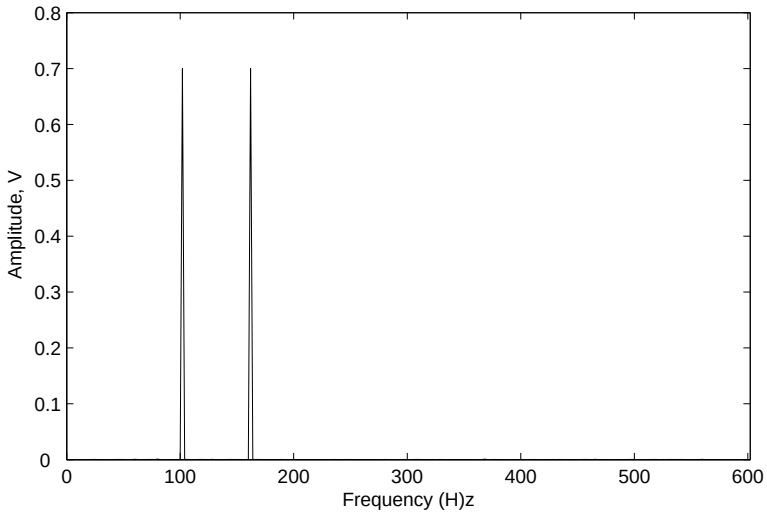

FIGURE 7.23: Spectrum of the initial output in nonlinear ANVC

The reference signal $r(t)$ is obtained from $d(t)$ by:

$$r(t) = 0.4d(t-10) + N_r \tag{7.9}$$

where N_r is a white noise with variance 0.001. There exists white noise N_y with variance 0.001 in the path of measurement of $\boldsymbol{y}$.

In nonlinear systems, considering the change of frequency set Ω, the common period is always set to some large value. In the following simulation work, the common period is defined as $N = 2000$.

7.3.2 Simulation result

As shown in Fig. 4.3, the tunable modules are a 6th order FIR filters in the feed-forward controller and are 3rd order FIR filters in the the feedback controller. All the initial controllers are set to zeros. The initial performance criterion without control is 0.49. In the 2nd to 4-th iteration, all feed-forward controllers are manually set and all the feedback controllers keep zeros.

In the first simulation with 'saturation' nonlinearity, the step size (adaptation gain) for the feed-forward controller tuning is $\mu_f = 0.25$ and the step size for the feedback controller tuning is $\mu_h = 0.025$.

Fig.7.24 shows the updating of performance in a typical tuning example. After 50 tuning iterations, the 'final' performance is 0.0196 with cancelation of 14.0dB. Fig.7.25 is the partial power spectrum of the final output, where the main frequencies are 160Hz, 420Hz, 40Hz, 360Hz and 220Hz in turn.

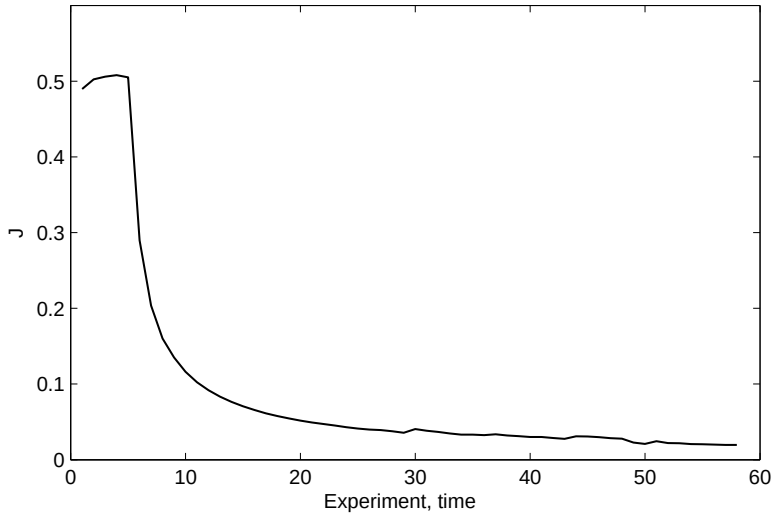

FIGURE 7.24: Performance updating in NL-FD-IT in saturation nonlinearity

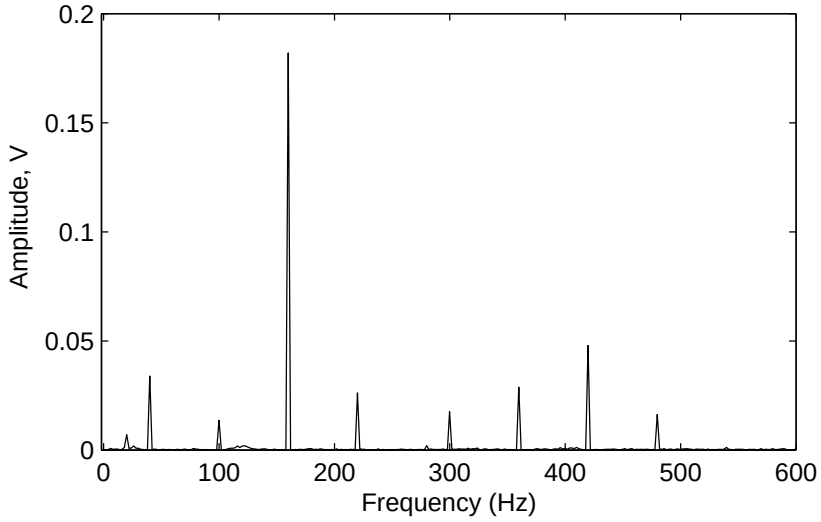

FIGURE 7.25: Spectrum of the final output after 58 tuning steps for saturation nonlinearity

In the second simulation, with regard the stability sensitivity of 'dead zone' nonlinearity to the feedback controller, only the feed-forward controller is tuned. The step size for feed-forward controller tuning is $\mu_f = 0.4$ and $\mu_h = 0.04$ for the feedback controller. Fig. 7.26 shows the updating of performance in a typical tuning example. After 50 tuning iterations, the final performance is 0.0115 with cancelation of 16.3dB. Fig.7.27 is the partial spectrum of the final output, where the main frequencies are 420Hz, 100Hz, 160Hz, 220Hz and 280Hz in turn.

As shown in Fig. 7.26 and Fig. 7.27, the nonlinearities in these two simulation examples are significant by the end of tuning and FSF-FD-IT give satisfying tuning performance under the nonlinear conditions.

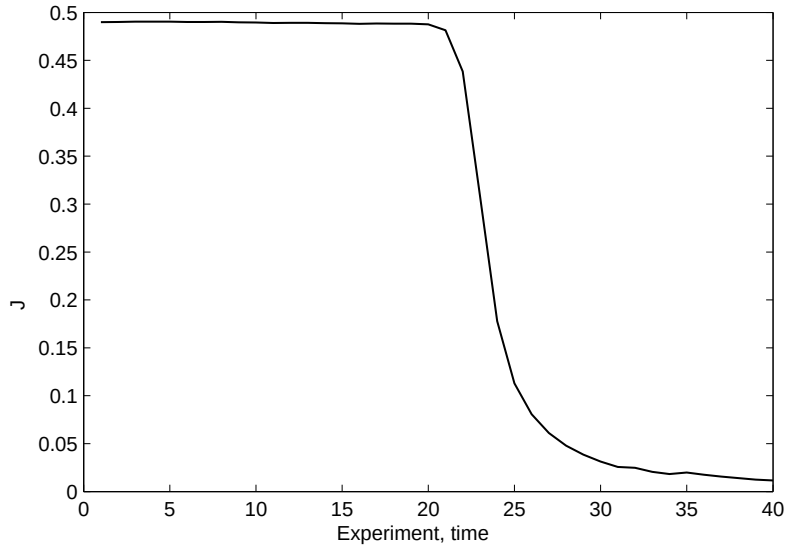

FIGURE 7.26: Performance updating in NL-FD-IT in dead-zone nonlinearity

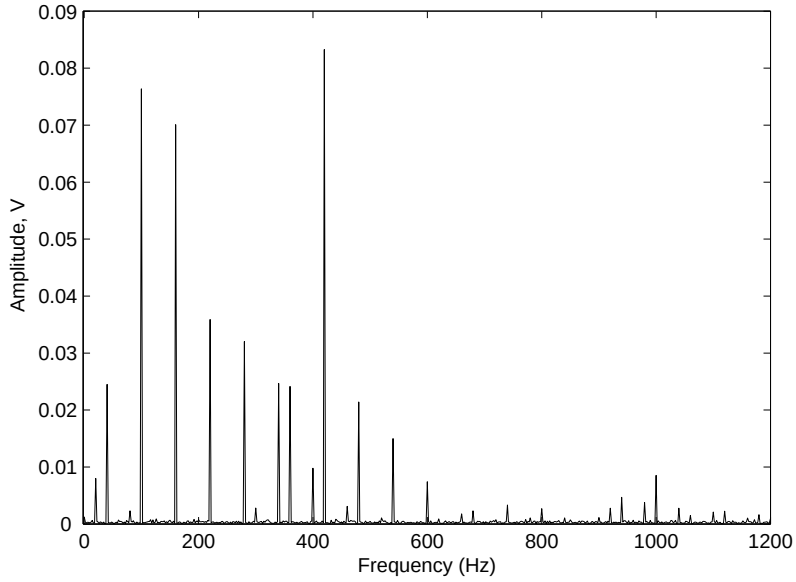

FIGURE 7.27: Spectrum of the final output after 40 tuning steps for dead-zone nonlinearity

7.4 Summary

This Chapter tested the proposed Iterative Tuning in the Frequency Domain (FD-IT) through simulation examples.

In the simulation work for FD-IT in SISO and MIMO LTI systems, two control structures, i.e., FIR-FD-IT and FSF-FD-IT, are used to illustrate the flexibility of FD-IT with different type of controllers. The results of FIR-FD-IT and FSF-FD-IT are compared and the reasons for different performances are discussed to see which FSF-FD-IT presents better tuning performance for ANVC control. Robustness against some errors in the frequency domain is illustrated through simulation examples as well in order to see which FD-IT presented satisfactory robustness against the error of common period N.

Two typical nonlinearities in actuators, saturation and dead-zone, are simulated to test the usefulness of FD-IT in the given nonlinear system. The simulation result illustrate that FD-IT is applicable to nonlinear ANVC engineering problems.

Chapter 8

Experimental Work

This Chapter presents laboratory tests of the proposed FD-IT method for ANVC. FD-IT were implemented on a hardware setups to demonstrate its effectiveness, an air duct system.

There are three parts in this Chapter. First the hardware setup is described including some nonlinear specification of actuators. The software setup is followed in the second part. Finally, the experimental results are shown before the summary.

8.1 Hardware Implementation

8.1.1 Platform of duct system

As shown in Fig. 8.1, an air duct system is used to test FD-IT for ANC in a real semi-closed environment.

FIGURE 8.1: Duct system

The duct is made of aluminium plates. It consists of two main parts. As shown in Fig.8.2, the speaker (EuroTec 520CO 4Ω, 10w), i.e., Spk0, fixed in the left part, is a disturbance speaker. In the middle of the whole duct, the speaker (EuroTec 520CO), i.e., Spk1, is fixed in the right part as a control speaker. The reference microphone (Mic0) is 10cm right from the disturbance

speaker (Spk0). The error microphone (Mic1) is fixed to the right end of the duct and it acquires the error signal as system output $\boldsymbol{y}$.

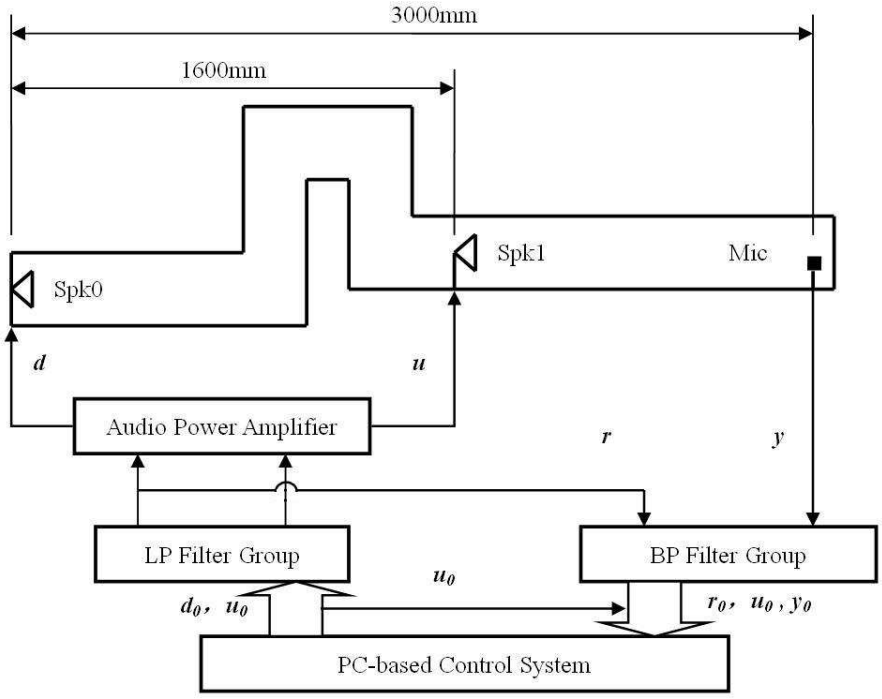

FIGURE 8.2: Schematic diagram of the duct system used.

The disturbance speaker and control speaker are driven by an audio power amplifier (SONY TA-FE570), which amplifies the stimulus source signal from analogue output of a PC after low-pass filtering. The error signal collected by Mic1 is amplified by an audio power amplifier (Philips TD1015) with 6V power from a regulated power supply (MANSON EP-920), and the reference signal is amplified by another TD1015 with 3V power from another EP23. They both feed into the analogue input of the PC.

The disturbance source signal into the power amplifier (TA-FE570) is derived from an analogue output of a PC-based control system, and filtered by a low-pass RC filter with cut off frequency of 4000Hz. The disturbance source signal and control signals are both produced by C programs running on a DSP board in the PC-based control system.

The amplified error signal from TDA1015 is filtered through a band-pass RC filter with pass-band from 65Hz to 4000Hz, and then fed into the analogue input of the PC-based control system. The experimental data are processed by the DSP board in the PC.

In order to test the feed-forward tuning part of FD-IT, the reference signal $\boldsymbol{r}$ is derived from the analogue output of thde PC, and filtered by a low-pass RC filter with 4000Hz cut off.

8.1.2 PC-based control system

The experimental platform is controlled by the a PC-based control system that comprise three parts:

1. A/D and D/A interface: Blue Wave System PC/16IO8 multichannel I/O board provides a 16 channel 12 bit Analog to Digital Converter (ADC), an 8 channel 12 bit Digital to Analog Converter (DAC) and 4 digital input lines and 4 digital output lines. It acquires analogue measurement data and converts them to digital values for the DSP control unit. It also converts digital control signals from the DSP control unit to analogue disturbance and control signals used in experimental platform.

2. DSP control unit: Blue Wave System PCI/C44S-60-1 is a DSP board based on Texas Instruments TMS320C4x range of Digital Signal Processors (DSPs). Independent of the PCI bus, it has a parallel data transfer interface, the Blue Wave DSPLINK2 interface, to communicate with PC/16IO8 board to receive experiment data and send disturbance source and control signals. It has a full 32 bit PCI interface that allows rapid communication between the host PC and other PCI bus peripherals to send tuning data and results. It acts as disturbance source generator and real-time controllers $H(\boldsymbol{w}_H)$ and $F(\boldsymbol{w}_F)$, in the PC-based control system. In these two experiment, the sampling rates are both set at 8000Hz by the C program running in DSP board.

3. PC host unit: Compatible PC system has AMD Althlon 1G Hz CPU and 384M RAM, and the operating system is Windows 2000. The PC host acts as manager of the whole PC-based control system, it perform initialization, receives tuning data, implements FD-IT algorithm, sends tuning result, manages the tuning process.

8.1.3 FRF of actuator path

As discussed in Chapter 7, nonlinearities exist in real dynamics, such as: sensors, actuators. People often choose some 'linear' dynamic range to simplify the control problems. However, the real actuator path, including, audio amplifier, speaker, sometime shows some unignorable nonlinear characteristics.

In this experimental ANC system, 200Hz, 400Hz, 500Hz and 800Hz harmonic signals are produced from the DSP card to stimulate Spk1. Before the ANC experiments a series of test experiments can be performed to investigate the frequency response of the whole actuator path with respect to those frequencies.

For instance, the frequency response of 500Hz is investigated through a series test experiments. In these experiments, a 500Hz single harmonic stimulus signal is produced by the DSP card with varying amplitude K, i.e., $y_0(t) = K\sin(2\pi \times 500)$ to stimulate the control speaker Spk1.

TABLE 8.1: Frequency Response at 500Hz

Amplitude of stimulus input K	0.1	0.2	0.3	0.4	0.5
Amplitude of output (V)	0.06	0.19	0.32	0.45	0.54
Phase of output (°)	-145	-146	-147	-147	-147
Amplitude of stimulus input K	0.6	0.7	0.8	0.9	1.0
Amplitude of output (V)	0.57	0.59	0.61	0.61	0.62
Phase of output (°)	-148	-148	-149	-149	-149

The output signal is measured by the sensor Mic. For each different K, 10 experiments are done to find the mean value of the output.

The result of frequency response for 500Hz is shown in Table. 8.1, and the amplitude response is illustrated as Fig. 8.3.

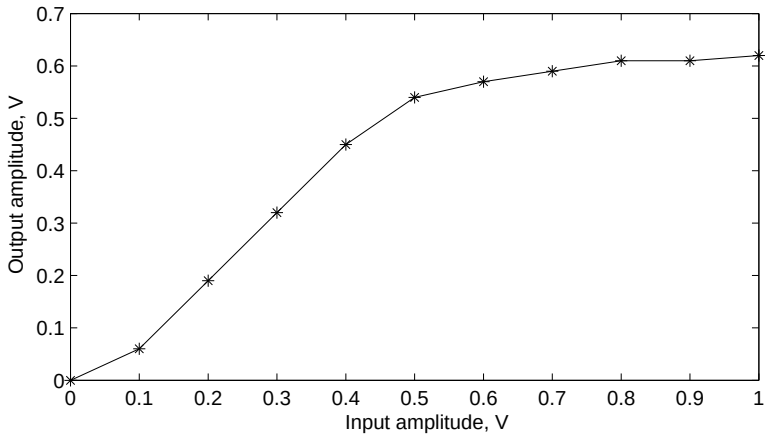

FIGURE 8.3: FRF of actuator path at 500Hz

According to above FRF test experiments the FRF at 500Hz presents three different sections:

1. When $K < 0.1$, FRF presents small gain that looks like 'dead zone' nonlinearity;
2. When $0.1 < K < 0.5$, FSF presents higher gain that looks like linear dynamics;
3. When $K > 0.5$, FRF presents smaller gain that looks like 'saturation' nonlinearity.

8.2 Software Implementation

With regard to the limitations of the DSPs used, the software implementation is divided into two parts: real time control on the DSP board and iterative tuning on the PC host. These two parts run simultaneously and require proper communication and cooperation.

This kind of adaptive control system requires several independent routines to cooperate to realize the control task. It can be implemented through agent-based programming techniques, which makes the control system smarter and more flexible.

8.2.1 Agent-based programming

An Agent is a system that is situated in some environment and that is capable of autonomous action in this environment in order to meet its design objectives. With a deeper analysis of possible agent behaviours, researcher developed some models of agents: logic-based agents [82], reactive agents [46], belief-desire-intention (**BDI**) agents [114, 115], and layered architectures agent [98].

As one of the originators of the concept of Agent [12], agent based control has been applied in various environments requiring autonomous control, such as: robot [37], unmanned vehicle [30], traffic control [105], warehouse management [68].

There are some basic characteristics of ”agents” [60]:

- **Autonomous**: they react themselves to observations of their environment without requiring explicit commands by making their own decisions,
- **Proactive**: they recognize and react to changes in the environment which present opportunities,
- **Embedded**: their actions take into consideration the real-time constraints imposed by the environment,
- **Heterogeneous**: many different kinds of agents can work together in the same system, and be added or removed without interrupting it.

This notation is quite similar to the concept of adaptive control systems, that is a system requiring sensing, taking actions, and is designed to satisfy some (performance) objective.

The use of the concept of ‘Agent’ can increase productivity of organizations both in cooperative and competitive environments. Therefore, the application of Agent concepts in adaptive control systems should focus on the management of adaptive process.

Similar to the idea proposed in [140, 141], an adaptive control system can be organized as an agent-based system as shown in Fig. 8.4:

- **ACa**: Adaptive control algorithm implementation.
- **CMa**: Supervisor or organizer of the whole adaptive control system.
- **Ea**: Executive agent to acquire data and produce control signals.

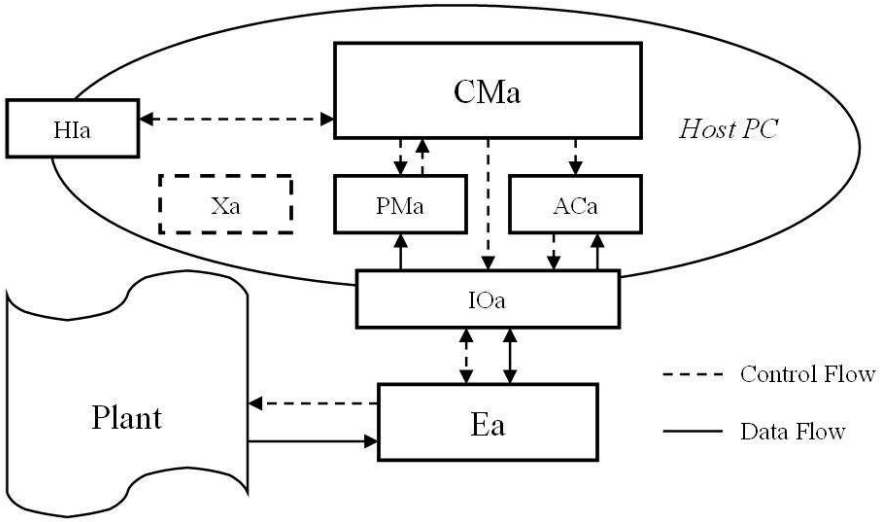

FIGURE 8.4: Agent-based adaptive control system

- **HIa**: Interface to human beings.
- **IOa**: I/O interface between host PC and actuator/sensor components.
- **PMa**: Modelling agent for adaptive control if necessary.
- **Xa**: Removable and flexible agent offering some more functions.

8.2.2 A simple Agent-based structure: MATLAB-C-DSP mixed programming

MATLAB is a very powerful software for control engineering. However, there is no direct interface between MATLAB and PCI/C44S-60-1 DSP board. Therefore, MATLAB-C-DSP mixed program is developed to realized a simple agent-based structure for FD-IT as shown in Fig. 8.5.

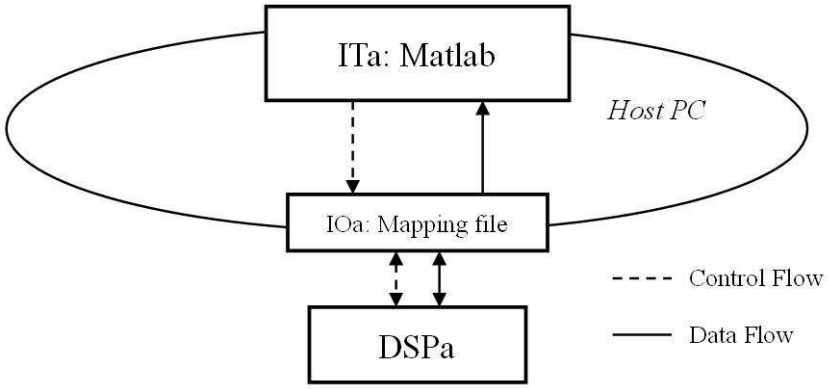

FIGURE 8.5: Schematic diagram of FD-IT software implementation by agents.

- **ITa**: It is the union of all the single agents in host PC. There are a lot of integrated functions and rich resource of control routines available online for MATLAB. It is very conve-

nient to perform control tuning and improve tuning algorithm in MATLAB. In FD-IT, the program in MATLAB is in charge of the management of iterative tuning processes, including: initialization of work, manually setting controllers in extra experiments, receiving tuning data, implementing FD-IT tuning, sending tuning result, adaptively determining step size μ, stopping tuning process.

- **IOa**: It is a communication agent between DSP and MATLAB. Since the PCI/C44S-60-1 board is not supported by MATLAB, there exists an interprocess communication problem between the two sub-communication-process: DSP-Operation System (OS), and OS-MATLAB. Windows 2000 has supplied several techniques for interprocess communications [120], such as: WMD hardware driver, Clipboard, DDE, COM. File mapping [117] enables a process to treat the contents of a memory file as if they were a block of memory in the process's address space, which is widely used in a lot of OSs. In this agent-based programming, mapping file techniques is adopted to make the program more mobile because it is used in some embedded system such as Win CE [9], Linux [113].

  In PC side, Blue wave system provides some C-based library functions to read and write data between memory in the host PC and shared memory in PCI/C44S-60-1 board. A service routine is programmed to read tuning data from DSP shared memory to mapping a file and write tuning parameters from mapping file to DSP shared memory.

  At the same time, MATLAB-R12 provides an interfaces to external routines written in C languages, i.e. C MEX-files, that are dynamically linked subroutines that the MATLAB interpreter can automatically load and execute. Using C MEX-files integrated mapping file I/O functions, MATLAB can share the mapping file accessed by the previous service routine.

- **DSPa** It is the executive agent that is in charge of acquiring experimental data and producing control actions through the real-time controller on the DSP board. In DSP side, Blue Wave System (now part of Motorola) also provides C-language-based and assembly-language-based methods to program the routine in DSP, which is written off-line on the PC side and loaded into the DSP during the initialization of the whole control system.

8.3 Experimental Results

In ANC in the duct system, the level of performance during tuning process demonstrates the effectiveness of FD-IT in an actual ANVC system. Given the nonlinearity in the actuator path, linear FD-IT and nonlinear FD-IT are both tested in the duct system. Two types of source disturbances are produced on the DSP board, which are given by

$$\boldsymbol{d}_{00} = 0.33 \times [sin(2\pi \times 200t) + sin(2\pi \times 400t) + sin(2\pi \times 500t)] \tag{8.1}$$

and

$$\boldsymbol{d}_{10} = 0.33 \times [sin(2\pi \times 400t) + sin(2\pi \times 500t) + sin(2\pi \times 800t)] \tag{8.2}$$

In all experiments, the initial controller is to set $F^0 = 1.2$ and $H^0 = 0$. The typical initial states of the ANC system is illustrated as following Fig. 8.6 and Fig. 8.7, that have initial performance as $J^0 = 0.4956$ and $J^0 = 0.4378$, respectively.

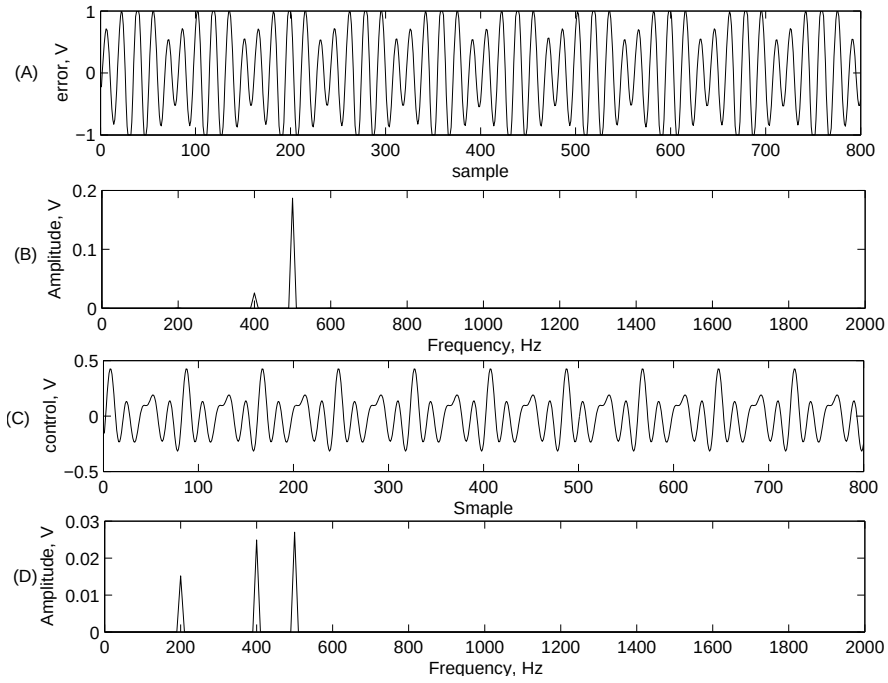

FIGURE 8.6: Initial state in duct system with 200,400 and 500Hz disturbance, (A) initial error output, (B) power spectrum of initial output, (C) initial control actions, (D) power spectrum of initial control

In the initial manual experiments, the manually set controllers are given by $F^i = F^0 + \Delta F^i, i \neq 0$, where the random parameter change $||\Delta F^i||_2 < 0.1||F^0||_2$.

In FIR-FD-IT, F is 40-th order FIR controller and H is 10-th order FIR controller. In FSF-FD-IT, 2nd order Butterworth filter is adopted, and the FIR tunable module in F and H are 16-th and 8-th order, respectively. In the linear FD-IT, the common periods is set as $N = 800$.

In the nonlinear case, FIR-FD-IT is adopted for its simplicity in practice, as it has the same controller structures as the linear FIR-FD-IT. The common period in nonlinear case is set as $N = 2000$.

Each iteration lasts $3N$ sampling periods including $2N$ sampling periods to let the system reach steady state, which means the iterations are at least 0.3s and 0.75s in linear and nonlinear cases, respectively.

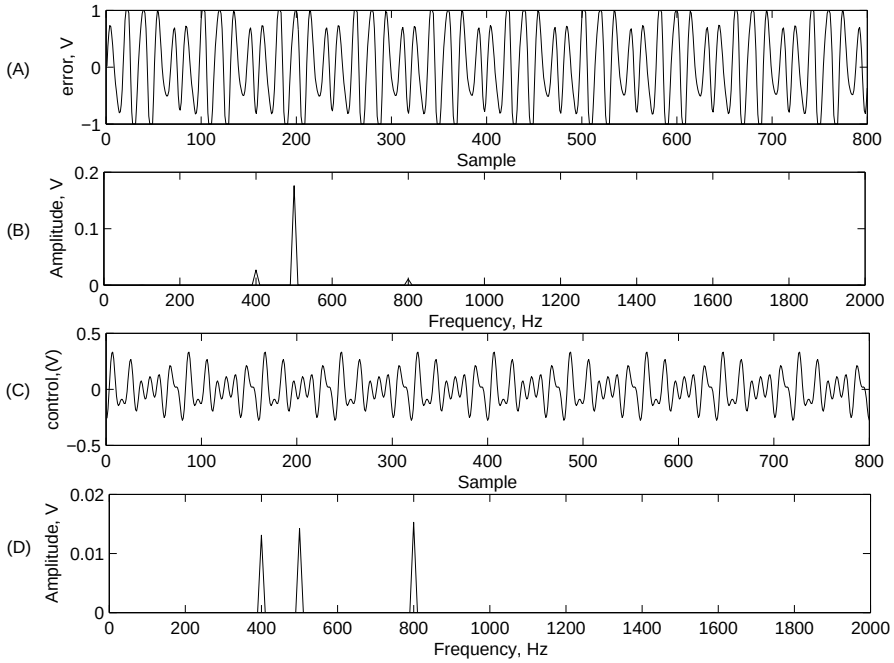

FIGURE 8.7: Initial state in duct system with 400,500 and 800Hz disturbance, (A) initial error output, (B) power spectrum of initial output, (C) initial control actions, (D) power spectrum of initial control

8.3.1 Experimental results of linear FD-IT in duct system

In this subsection, linear FD-IT algorithm is used to implement ANC in the duct system.

Firstly, $\boldsymbol{d}_{00}$ is used as source disturbance to stimulate the disturbance speaker.

In linear FIR-FD-IT, the tuning step sizes are set as $\mu_F = 0.20, \mu_H = 0.02$. After 80 experiments, the final performance $J^{80} = 0.0103$ with 16.8dB cancellation, which spend 59.23s. The final states is illustrated in Fig. 8.8. The performance curve for the tuning process is shown in Fig. 8.9.

The linear FSF-FD-IT is also tested in this platform, the tuning step sizes are set as $\mu_F = 0.8, \mu_H = 0.4$. After 50 experiments, the final is $J^{50} = 0.0108$ with 16.6dB cancellation after 45.17s of tuning. In this air-duct system, it did not demonstrate notable advantages about the tuning results, but brought more complicated FSF structures than FIR-FD-IT. In the following description, FIR-FD-IT is mainly discussed to compare the tuning performance of linear FD-IT and nonlinear FD-IT because it is used in both linear and nonlinear tuning.

Secondly, $\boldsymbol{d}_{10}$ is used as source disturbance to stimulate the disturbance speaker.

In linear FIR-FD-IT, the tuning step sizes are set as $\mu_F = 0.24, \mu_H = 0.024$. After 80 experiments, the final performance $J^{80} = 0.0429$ with 10.1dB cancellation, which spend 61.12s. The final states is illustrated in Fig. 8.10. The performance curve for the tuning process is shown in Fig. 8.11.

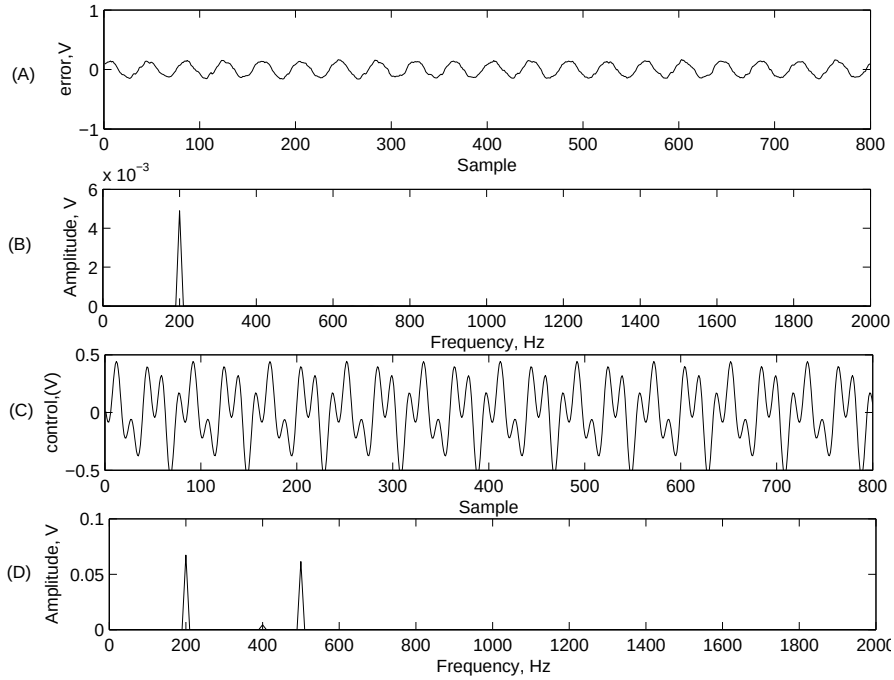

FIGURE 8.8: Final state in duct system using linear FIR-FD-IT with disturbance $\boldsymbol{d}_{00}$, (A) final error output, (B) power spectrum of final output, (C) final control action, (D) power spectrum of final control

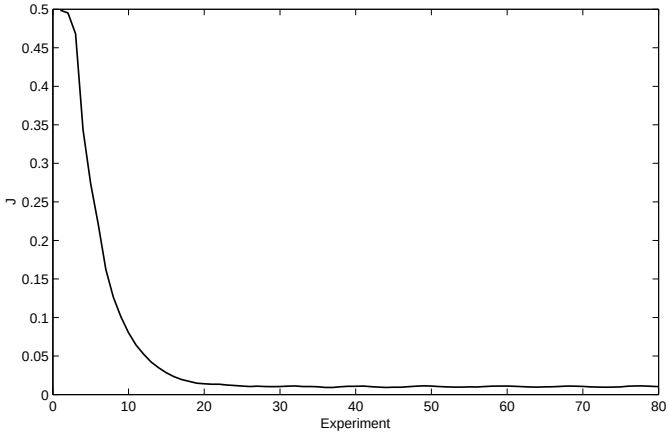

FIGURE 8.9: Performance curve for linear FIR-FD-IT with disturbance $\boldsymbol{d}_{00}$

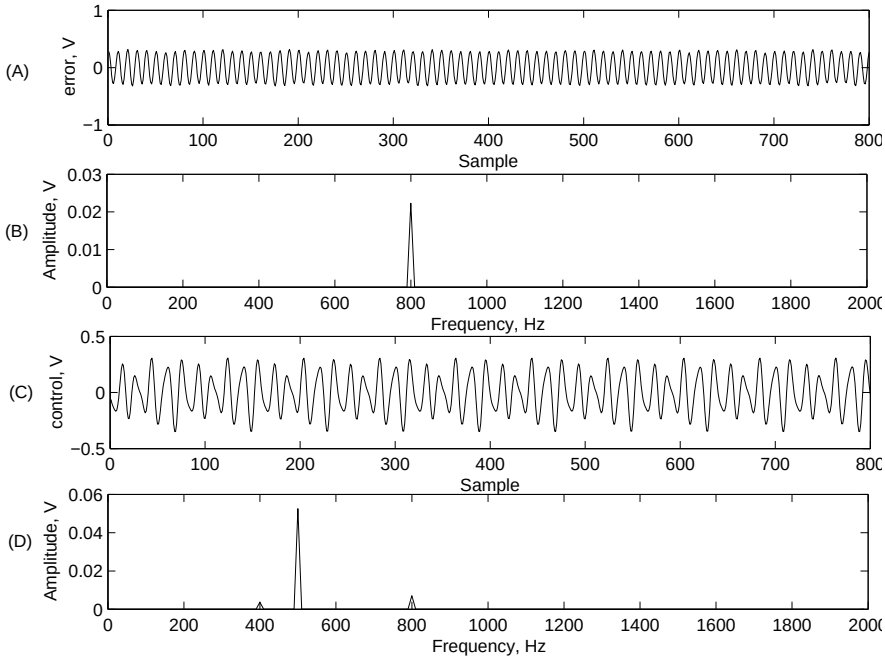

FIGURE 8.10: Final state in duct system using linear FIR-FD-IT with disturbance $\boldsymbol{d}_{10}$, (A) final error output, (B) power spectrum of final output, (C) final control action, (D) power spectrum of final control

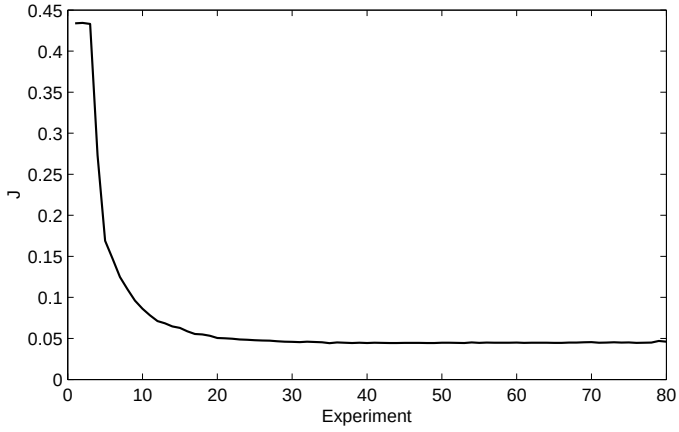

FIGURE 8.11: Performance curve for linear FIR-FD-IT with disturbance $\boldsymbol{d}_{10}$

8.3.2 Experimental results of nonlinear FD-IT in duct system

In following experiments, nonlinear FIR-FD-IT is used to implement nonlinear tuning on the duct system which achieves a higher level of cancellation than linear FD-IT.

Firstly, the duct system is disturbed by $\boldsymbol{d}_{00}$, and nonlinear FIR-FD-IT is adopted in tuning which has tuning step sizes as $\mu_F = 0.20, \mu_H = 0.02$.

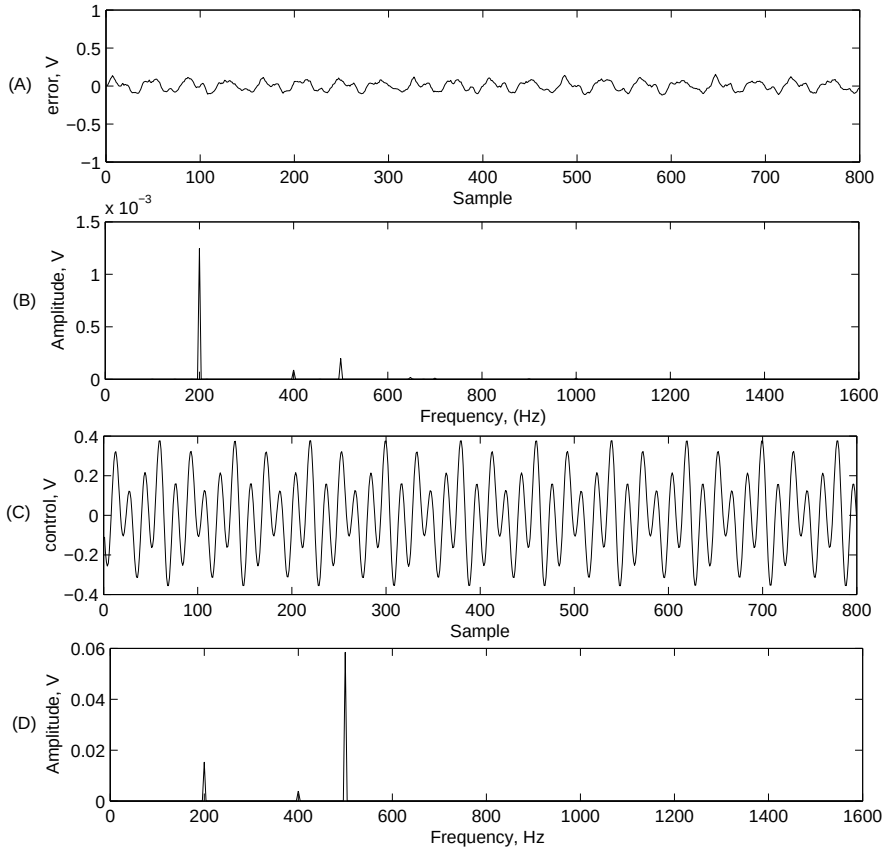

FIGURE 8.12: Final state in duct system using nonlinear FIR-FD-IT with disturbance $\boldsymbol{d}_{00}$, (A) final error output, (B) power spectrum of final output, (C) final control action, (D) power spectrum of final control

After 79 experiments, the tuned performance is $J^{79} = 0.00268$ with 22.6dB cancellation after 80.49s of tuning. The final outputs and control actions of the duct system are illustrated in Fig. 8.12, and the performance updating during the tuning is shown in Fig. 8.13.

Secondly, the duct system is disturbed by $\boldsymbol{d}_{10}$ and nonlinear FIR-FD-IT is adopted in tuning which has tuning step sizes as $\mu_F = 0.24, \mu_H = 0.024$.

After 71 experiments, the tuned performance is $J^{71} = 0.00011$ with 25.8dB cancellation, which spent 84.56 second. The final states of the duct system is illustrated in Fig. 8.14, and the performance updating during the tuning is shown in Fig. 8.15.

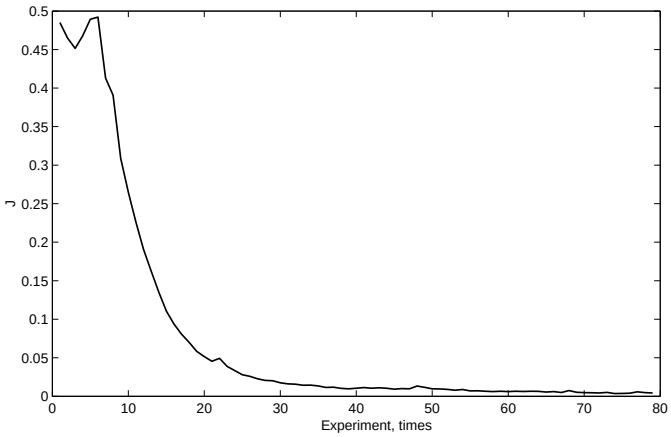

FIGURE 8.13: Performance updating in nonlinear FIR-FD-IT with disturbance $\boldsymbol{d}_{10}$

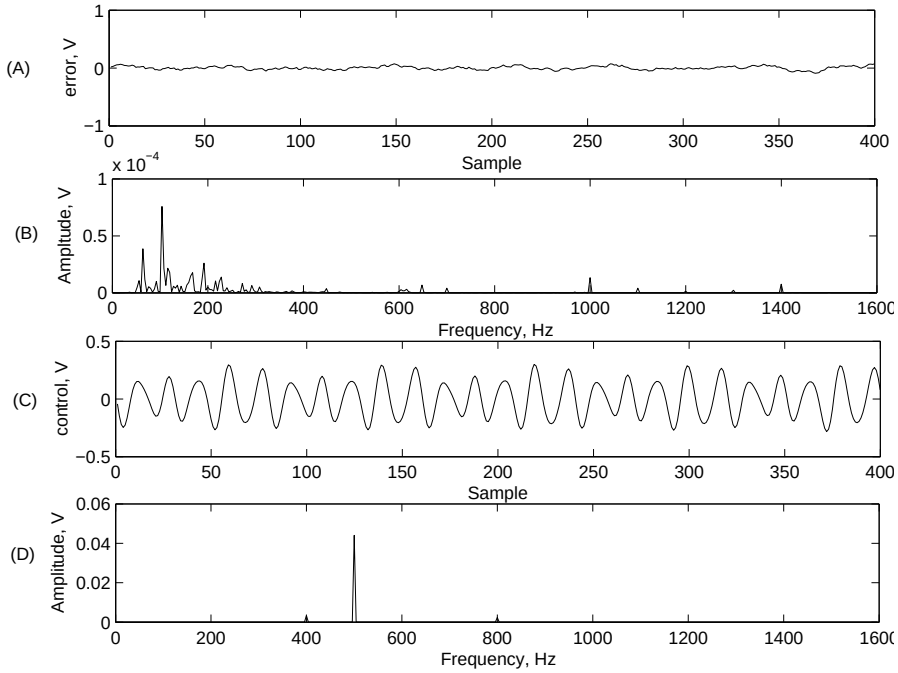

FIGURE 8.14: Final state in duct system using nonlinear FIR-FD-IT with disturbance $\boldsymbol{d}_{10}$, (A) final error output, (B) power spectrum of final output, (C) final control action, (D) power spectrum of final control

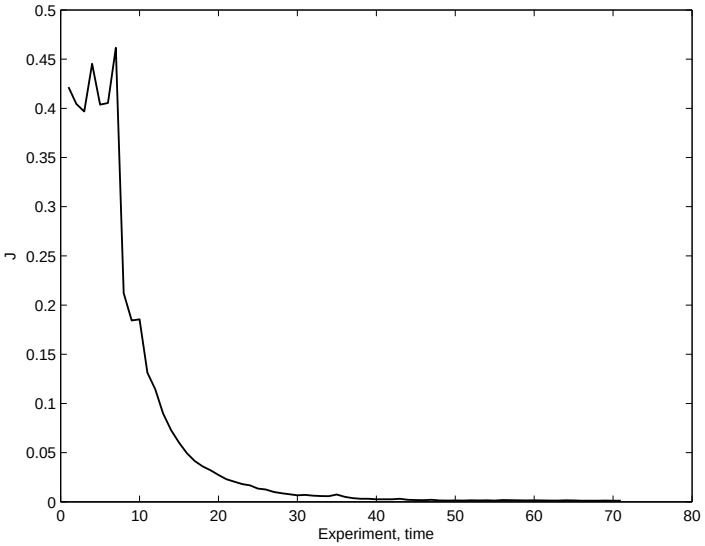

FIGURE 8.15: Performance curve for nonlinear FIR-FD-IT with disturbance $\boldsymbol{d}_{10}$

8.3.3 Comparison and discussion

According to the experiment results, it is easy to find that nonlinear FD-IT achieves better tuning result than linear FD-IT, which provides additional 5.8dB cancellation in $\boldsymbol{d}_{00}$ disturbance case and additional 15.7dB cancellation in $\boldsymbol{d}_{10}$ disturbance case.

Comparing Fig. 8.8 to Fig. 8.12, the power spectrums of control action are 0.059 at 200Hz, 0.005 at 400Hz and 0.067 at 500Hz in linear FD-IT, where they are 0.016 at 200Hz, 0.003 at 400Hz and 0.060 at 500Hz in nonlinear FD-IT. Comparing Fig. 8.10 to Fig. 8.14, the power spectrums of control action are 0.004 at 400Hz, 0.052 at 400 and 0.007 at 800Hz in linear FD-IT, while they are 0.003 at 400Hz, 0.044 at 400 and 0.002 at 800Hz in nonlinear FD-IT. The extra cancellation are supplied by the change in control action when the control signals are very small.

Recalling the FRF test experiments in the previous section (as Fig. 8.3), the FRF of the actuator path presents some dead zone nonlinearity when the stimulus signal is very small. Therefore, the extra cancellation achieved by nonlinear FD-IT can be well explained: When the control signal is some proper large, the duct system is in the range of good linear dynamics , which linear FD-IT tune the system well. When the control signal becomes small enough, the duct system presents some dead zone nonlinearity, which linear FD-IT is not applicable any more, and nonlinear FD-IT can make some further tuning.

According to the comparison of Fig. 8.9 with Fig. 8.13 and Fig. 8.11 with Fig. 8.15, some difference in the tuning process from linear FD-IT to nonlinear FD-IT can be observed:

- In linear FD-IT, there are only two initial manual extra experiments as shown in Fig. 8.9 and Fig. 8.11; In nonlinear FD-IT, there are more than two manual extra experiments

which is actually 6 initial manual experiments as shown in Fig. 8.13 and Fig. 8.15.

- In linear FD-IT, the tuning process is smooth as shown in Fig. 8.9 and Fig. 8.11 for the computation of Φ_G is simple and robust; In nonlinear FD-IT, the tuning process has some fluctuation as shown in Fig. 8.13 and Fig. 8.15 for the computation of $\Phi_{G'}$ is more complicated and more sensitive to noise.

8.4 Summary

This chapter demonstrates the experimental work to test FD-IT for ANC problems. An air duct system is built up. A simple Agent-based FD-IT system is realized with Matlab-C-DSP programming techniques.

In linear FD-IT, 16.8dB cancellation has been achieved to suppress 200, 400 and 500Hz disturbances, and 10.1dB cancellation has been achieved to suppress 400, 500 and 800Hz disturbances in the duct system.

Despite the dead zone nonlinearity in the actuator path, 22.6dB cancellation has been achieved to suppress 200, 400 and 500Hz disturbance, and 25.8dB cancellation has been achieved to suppress 400, 500 and 800Hz disturbance in the duct system.

The experimental results illustrate the effectiveness of FD-IT under the practical conditions.

Chapter 9

Conclusion and Future Work

The conclusions are concerned with three aspects of the thesis: first with the gradient estimation theory in the frequency domain, secondly with the iterative tuning method in the frequency domain, and thirdly with tests in simulation and experimental work.

The description future covers the topic of improving the gradient estimate, the theoretical analysis itself and also future implementation techniques of the iterative tuning algorithm.

9.1 Conclusions on the Gradient Estimate in Frequency Domain

The first part of the thesis proposed a new gradient estimate completely from the aspect of the frequency domain.

Before proposing the new gradient estimation theory, a general framework of Active Noise and Vibration Control system with hybrid feedback and feed-forward controllers was defined. This general framework, including signals, dynamics and cost functions was presented in the frequency domain.

Based on the proposed framework, the gradient of the system output spectrum with respect to controller parameters was obtained through local linearization of mapping the secondary path from the change of input spectrums to the change of output spectrum. The gradient estimate of the system output spectrum with respect to controller parameters is at the core of the proposed gradient estimation theory. Since the performance criterion function can always be expressed as a function of output and control actions in the frequency domain, the gradient estimate of performance with respect to the control parameters can be obtained. In the proposed gradient estimation theory a key term is the derivative matrix of frequency response of the closed-loop dynamics. While the closed-loop dynamics represents the mapping form the change of the control action to the change of system output, the proposed gradient estimation theory illustrates the chain rule of derivation in the frequency domain, which maps small (infinitesimal) change in

the controller parameters to a change in the spectrum of control actions, then maps to a change in the output spectrum, and finally to some change in control performance. When compared with the gradient estimate techniques of some popular ANVC methods, the new proposed frequency domain gradient estimate can be similar to the other gradient estimate in the time domain.

It is worth noting that the new proposed gradient estimate is not only suitable for ANVC problems, but also for the general control problems since common discrete signals can be expressed using discrete spectra in the frequency domain. It is particularly suitable for ANVC problem with periodic disturbances because the description of the system dynamics and signals in the frequency domain can be greatly simplified by focusing to a limited frequency set while the signals are often long data sequences.

As shown the new proposed gradient estimation theory can be extended to the MIMO case and also to nonlinear system. Given proper descriptions of signal spectra and an extended FRF in MIMO system and derivative matrix of GFRF in nonlinear system, the proposed gradient estimation theory is applicable to MIMO and nonlinear system as well.

9.2 Conclusions on the Iterative Tuning in the Frequency Domain

Based on the proposed gradient estimate, the key problem in ANVC is to solve derivative matrix of the FRF of closed loop dynamics which includes an unknown secondary path. The basic idea is to estimate the matrix through spectrum difference pairs of inputs and outputs.

This thesis proposed a new adaptive control method for ANVC: Iterative Tuning in the Frequency Domain (FD-IT). In order to obtain the closed-loop dynamics, spectrum difference pairs are produced through different experiments with different controller parameters, where the parameters can be either manually set or tuned using gradient-based methods. Therefore no additional path is required to inject extra signals unlike in TD-IFT. Except the extra manual (or automated) experiments in the initial stage of tuning, only one experiment is required to do gradient-based updating in subsequent tuning iterations. This is simple in control structures and convenient in control implementations.

FD-IT imposes no limitation on the controller format except that the controller should be a differtiable dynamics in the frequency domain. Using Frequency-Selective-Filtered (FSF) filters, FSF-FD-IT can concentrate on the tuning of FSF controllers for a finite frequency set that is relevant for the periodic disturbance. With Zeros-Poles-Gains (ZPK) format, it is easier to analyse and maintain the stability of feed-forward control. Considering stability of the closed-loop, one solution is to perform extra experiments to test the generalized stability margin of the current closed-loop dynamics. Another practical method is to use some hybrid cost function including the Vinnicombe distance of feedback controllers.

In comparison with some other adaptive control method for ANVC in the literature, FD-IT presents some notable advantages:

- It can be used to tune feedback controllers as well as feed forward controllers;
- It has flexible and simple control structure without additional signal injection path;
- It can perform gradient-based tuning through one experiment except some initial manually arranged set of experiments for initial tuning of parameters.

9.3 Conclusions on Tests in Simulation and Experimental Work

Tests of FD-IT in ANVC have been performed in simulation and on an experimental platform, both of these has proved the effectiveness of FD-IT in ANVC.

FIR-FD-IT and FSF-FD-IT have been test for the SISO and MIMO systems in simulation. While FIR-FD-IT has a simpler control structure, FSF-FD-IT offered much better tuning performance and stability of performance. The robustness of FSF-FD-IT against some errors in the frequency set has also tested in both SISO and MIMO systems. FSF-FD-IT has presented satisfactory robustness against errors in estimated common period. Two nonlinear simulated platforms were used to illustrate the applicability of FSF-FD-IT to nonlinear systems, in all cases significant cancellation of the disturbance was achieved.

In experimental test work, an air duct system is setup to test the performance of linear and nonlinear FD-IT in the real system. In the experiment works, linear FD-IT has achieved 16.8dB cancelation to suppress 200, 400 and 500Hz disturbance, and 10.1dB cancelation to suppress 400, 500 and 800Hz disturbance. Nonlinear FD-IT has achieved 21.6dB cancelation to suppress 200, 400 and 500Hz disturbance, and 25.8dB cancelation has been achieved to suppress 400, 500 and 800Hz disturbance.

9.4 Future Work

As the proposed gradient estimation theory and iterative tuning method is powerful, future work could continue along some lines of investigation as suggested here.

9.4.1 Extension of the gradient estimation theory

In this thesis the proposed idea concentrated on the ANVC problem with periodical disturbances. Although some brief statement about the extension of general framework from ANVC to common control problems has been presented to illustrate the applicability of the proposed idea in general control problems. Further study of the extension is worthwhile in future research. The extension can include different performance criterion functions than the average quadratic performance and focus on criteria suitable for servo control.

As mentioned above, one of the most notable advantages of the gradient estimate in the frequency domain is that when the system can be described in much simpler ways in the frequency domain, the gradient estimate can be greatly simplified through the proposed approach. Since the time domain and the frequency domain can be considered as two different integral operator domains in mathematics, this advantage can be extended to a general idea:

Given a general framework $\mathfrak{P}$ in the time domain, if we can find a proper integral operator F which the F transferred framework $F(\mathfrak{P})$ is simpler than $\mathfrak{P}$, the gradient estimate can be performed in the integral operatorred domain of F to simplify the computation work.

Obviously, the difficulty is how to find such a proper integral operator which can give simplified transferred representation of both signal and dynamics.

Noting there is no limitation about the integral operator, some techniques such as wavelet method and stochastic method can be some possible candidates. It is well worthy to investigate in theory further.

9.4.2 Improvement of iterative tuning algorithm

In this thesis the iterative tuning algorithmic description has mainly discussed how to process the output using a finite and fixed frequency set in the linear case. In the nonlinear case a realtime estimate of the frequency set has been introduced in FD-IT. How to process the output spectrum with infinite or varying frequency set is an interesting topic for control engineering. It can greatly extend the applicability of FD-IT from ANVC to more general control problems. The realtime frequency estimate can be just a very basic start of the theory and one may able to process varying frequency set signals. More detailed research work could be done to complete this topic.

Convergence of the algorithms and robustness of control implementation are always two of the most important theoretical topics in adaptive control. Especially, the development of some exact and practical robustness tuning algorithms are very valuable in control engineering.

9.4.3 Improvement in control implementation

In order to implement the iterative tuning algorithms in control engineering practice, there is a lot work to be done, one of which is how to organize different functional modules to realize controller tuning. While a simple PC-DSP board based programs and FSF FIR controllers are used in experimental work, some artificial intelligence techniques can be incorporated to manage the whole controller tuning work, including to perform proper extra manual experiments, to determine if the tuning converges, to give proper step size.

An agent-based control system can be suitable to to manage such kind of iterative tuning control system, especially for LTI system in which the tuning can be performed for each single

frequency. Some other work, such as managements about manual experiment and step size, monitoring the system stability, can be allocated to some individual agents which can monitor and instruct the control system with reasoning and flexibly.

Some other types of AI controllers can also be used in FD-IT, especially in the nonlinear case. For instance, Neural Network (NN) controller which has been successfully used in Active Noise Control (ANC) can be used in FD-IT as well.

References

[1] H. Aarts, R. M. Greten and P. Swarte. A special form of noise reduction. In A. E. Soc., editor, *21st AES Conference*, St. Petersburg, Russia, 2002. Proceedings Internoise.

[2] N. Abe. Passive and active switching vibration control with pendulum type damper. In *Control Applications, Proceedings of the 2004 IEEE International Conference on*, volume 2, pages 1037 – 1042, 2004.

[3] M. Akhtar, M. Abe, and M. Kawamata. A new variable step size LMS algorithm-based method for improved online secondary path modeling in active noise control systems. *Audio, Speech and Language Processing, IEEE Transactions on*, 14(2):720 – 726, 2006.

[4] J. Barrett. The use of functionals in the analysis of nonlinear physical systems. *Journal of Electronics and Control*, 15:567–615, 1963.

[5] E. Bedrosian and S. Rice. The output properties of volterra systems (nonlinear systems with memory) driven by harmonic and gaussian inputs. In *Proceedings of the IEEE*, volume 59.

[6] J. Benesty and P. Duhamel. A fast exact least mean square adaptive algorithm. *IEEE Transactions on Signal Processing*, 40:2904–2912, 1992.

[7] S. A. Billing and J. C. Peyton Jones. Mapping nonlinear integro-differential equations into the frequency domain. *Internationl Journal of Contrl*, 52:863–879, 1990.

[8] S. A. Billing and K. M. Tsang. Spectral analysis for nonlinear systems, part I - parametric nonoinar spectral analysis. *Journal of Mechanical Systems and Signal Processing*, 3:319–339, 1989.

[9] D. Boling. *Programming Microsoft Windows CE .NET*. Microsoft Press, 2003.

[10] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2nd edition, 2004. ISBN 0521833787.

[11] M. Brilliant. Theory of the analysis of nonlinear systems. Technical Report 345, Laboratory of Electronics, MIT, 1958.

[12] R. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–59, 1991.

[13] P. B. Brugarolas, V. Fromion, and M. G. Safonov. Robust switching missile autopilot. In *Proc. 1998 ACC*, pages 3665–3669, Philadelphia, PA, 1998.

[14] A. Burton. Active vibration control in automotive chassis systems. *Computing & Control Engineering Journal*, 4(5):225 – 232, 1993.

[15] D. Q. Cao, R. W. Tucker, and C. Wang. A stochastic approach to cable dynamics with moving rivulets. *Journal of Sound and Vibration*, 268:291–304, 2003.

[16] J. C. Carmona and V. M. Alvarado. Active noise control of a duct using robust control theory. *Control Systems Technology, IEEE Transactions on*, 8:930 – 938, Nov. 2000.

[17] C.-J. Chien, F.-S. Lee, and J.-C. Wang. Enhanced iterative learning control for a piezoelectric actuator system using wavelet transform filtering. *Journal of Sound and Vibration*, 299:605–620, Feb. 2007.

[18] S.-B. Choi and S.-R. Hong. Active vibration control of a flexible structure using an inertial type piezoelectric mount. *Smart Mater. Struct.*, 16(1):25–35, Feb. 2007.

[19] M. O. T. Cole, P. S. Keogh, C. R. Burrows, and M. N. Sahinkaya. Adaptive control of rotor vibration using compact wavelets. *Journal of Vibration and Acoustics*, 128(5): 653–665, 2006.

[20] M. Costa, J. Bermudez, and N. Bershad. Stochastic analysis of the LMS algorithm with a saturation nonlinearity following the adaptive filter output. *Acoustics, Speech, and Signal Processing, IEEE Transactions on*, 49(7):1370 – 1387, 2001.

[21] D. Das and G. Panda. Active mitigation of nonlinear noise processes using a novel filtered-s LMS algorithm. *Speech and Audio Processing, IEEE Transactions on*, 12:313 – 322, May 2004.

[22] M. de Diego, A. Gonzalez, G. Pinero, M. Ferrer, and J. J. Garcia-Bonito. Subjective evaluation of actively controlled interior car noise. In *Acoustics, Speech, and Signal Processing, (ICASSP '01). Proceedings. 2001 IEEE International Conference on*, volume 5, pages 3225 – 3228, May 2001.

[23] V. E. DeBrunner and D. Zhou;. Active non-linear noise control with non-linearities in the secondary path. In *Statistical Signal Processing, 2003 IEEE Workshop on*, pages 210–213, Sept. 2003.

[24] J. Dorsey. *Continuous and Discrete Control System*. McGraw - Hill Companyies, Inc., 2002. ISBN 0-07-248308-3.

[25] S. Douglas. Reducing the computational and memory requirements of the multichannel filtered-x LMS adaptive controller. In *Proc. Nat. Conf. Noise Control Eng.*, .

[26] S. Douglas. Fast implementations of the filtered-x LMS and LMS algorithms for multichannel active noise control. *IEEE Trans. Speech Audio Processing*, .

[27] S. C. Douglas. An efficient implementation of the modified filtered-x LMS algorithm. *IEEE Signal Processing Letters*, 4:286–288, 1997.

[28] M. Duchemin, A. Berlioz, and G. Ferraris. Dynamic behavior and stability of a rotor under base excitation. *Journal of Vibration and Acoustics*, 128(5):576–585, 2006.

[29] G. E. Dullerud and F. Paganini. *A Course in Robust Control Theory*. Springer, 2nd edition, 2005.

[30] H. Durfee, P. G. Kenny, and K. C. Kluge. Integrated permission planning and execution for unmanned ground vehicles. *Autonomous Robots*.

[31] S. D. W. E. and A. R.M. Three-dimensional headphone sound reproduction based on active noise cancellation. In *AES 113th Convention*, Los Angeles, CA, USA, Oct. 2002.

[32] W. Edmonson, J. Principe, K. Srinivasan, and C. Wang. A global least mean square algorithm for adaptive iir filtering. *IEEE Circuits and Systems II*, 45(3):379 – 384, March 1998.

[33] T. Eguchi and T. Nakamiya. An improved component-mode synthesis method to predict vibration of rotating spindles and its application to position errors of hard disk drives. *Journal of Vibration and Acoustics*, 128(5):568–575, 2006.

[34] S. Elliott. *Signal Processing for Active Control*. ACADEMIC PRESS, 2001. ISBN 0-12-237085-6.

[35] S. J. Elliott. Filtered reference and filtered error LMS algorithms for adaptive feedforward control. *Mechanical Systems & Signal Processing*, 12(6):769–781, 1998.

[36] S. J. Elliott and P. A. Nelson. Active noise control. *IEEE Signal Proceeding*, 10(4):12–35, 1993.

[37] J. Eze, H. Ghenniwa, and W. Shen. Distributed control architecture for collaborative physical robot agents. In *Systems, Man and Cybernetics, 2003. IEEE International Conference*, volume 3, pages 2977 – 2982, 2003.

[38] G. Feng, Y. F. Fah, and Y. Ying. Active airborne noise control of hard disk drive. In *Magnetic Recording Conference, Digest of the Asia-Pacific*, pages TU–P–27–01 – TU–P–27–02, Aug. 2002.

[39] A. Forrai, S. Hashimoto, H. Funato, and K. Kamiyama. Structural control technology: system identification and control of flexible structures. *Computing & Control Engineering Journal*, 12:257 – 262, Dec. 2001.

[40] K. Frampton. Distributed group-based vibration control with a networked embedded system. *Smart Materials and Structures*, 14:307–314, 2005.

[41] Y. Z. G., Z. W. Q., and S. T. T. A stochastic optimal semi-active control strategy for er/mr dampers. *Journal of Sound and Vibration*, 259:45–63, 2003.

[42] E. A. G. Bales, R. Glaese, and L. Shaw. Active damping and vibration control for aircraft fin and appendage structures. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, page 653, Reno, USA, Jan 2006.

[43] W. Gan, S. Mitra, and S. Kuo. Adaptive feedback active noise control headset: implementation, evaluation and its extensions. *Consumer Electronics, IEEE Transactions on*, 5(3):975– 982, 2005.

[44] W. S. Gan and S. M. Kuo. An integrated audio and active noise control headset. *Consumer Electronics, IEEE Transactions on*, 48:242 – 247, May 2002.

[45] C. B. Garcia and W. I. Zangwill. Determining all solutions to certain systems of nonlinear equations. *Mathematics of Operations Research*, 4:1–14, 1979.

[46] M. P. Georgeff and A. Lansky. Reactive reasoning and planning. In *Proc. of 6th National Conference on AI*, pages 677–682, Seattle, WA., 1987.

[47] M. Gevers. Towards a joint design of identification and control. In H. Trentelman and J. Willems, editors, *Essays on Control: Perspectives in the Theory and its Applications*, pages 111–151, Boston, MA, 1993. Birkhauser.

[48] L. Guoping and Z. Chunliang. Active vibration control of a isolation platform based on state space lqg. In *Robotics and Biomimetics. (ROBIO) 2005 IEEE International Conference on*, pages 427 – 431, 2005.

[49] S. M. Hashim, M. Tokhi, and I. M. Darus. Active vibration control of flexible structures using genetic optimisation. *Journal of Low Frequency Noise, Vibration and Active Control*, 25(3):195–207, Sept. 2006.

[50] S. Haykin. *Adaptive Filter Theory*. Prentice Hall, 4th edition, 2001.

[51] A. Helmicki, C. Jacobson, and C. Nett. Control oriented system identification: A worst-case/deterministic approach in H1. *IEEE Trans. Automatic control*, 36:1163–1176, 1991.

[52] H. Hjalmarsson. Iterative feedback tuning - an overview. *Int. J. Adaptive Control and Signal Processing*, 16:373 – 395, 2002.

[53] H. Hjalmarsson. Efficient tuning of linear multivariable controllers using iterative feedback tuning. *Int. J. Adaptive Control and Signal Processing*, 13:553–572, 1999.

[54] H. Hjalmarsson and L. O. Gevers M. Iterative feedback tuning: theory and applications. *IEEE Control Systems Magazine*, 18(4):26–41, 1998.

[55] M. Hodgson and J. Guo. Active local control of propeller-aircraft run-up noise. *Noise & Vibration Worldwide*, 37(3):9–13, 2006.

[56] J. Hu and J.-F. Lin. Feedforward active noise controller design in ducts without independent noise source measurements. *Control Systems Technology, IEEE Transactions on*, 8: 443 – 455, May 2000.

[57] Q. Hu, G. Ma, and C. Li. Active vibration control of a flexible plate structure using lmi-based H_∞ output feedback control law. In *Intelligent Control and Automation, . WCICA 2004. Fifth World Congress on*, volume 1, pages 738 – 742, Hangzhou,China, 2004.

[58] H. Jansson and H. Hjalmarsson. Gradient approximations in iterative feedback tuning for multivariable processes. *Int. J. Adaptive Control and Signal Processing*, 18(8):665–681, 2004.

[59] B. Jayawardhana, L. Xie, and S. Yuan. Active control of sound based on diagonal recurrent neural network. In *SICE Proceedings of the 41st SICE Annual Conference*, volume 5, pages 2666 – 2671, Aug 2002.

[60] N. Jennings and M. Wooldridge. Intelligent agents: Theory and practice. *The Knowledge Engineering Review*, 10(2):115–52, 1995.

[61] G. Jin, M. Sain, and J. Spencer, B.F. Frequency domain system identification for controlled civil engineering structures. *Control Systems Technology, IEEE Transactions on*, 13:1055 – 1062, Nov. 2005.

[62] K. A. W. Jingdou Wang, W Steve Shepard Jr and C. B. Gattis. Active vibration control of a plate-like structure with discontinuous boundary conditions. *Smart Mater. Struct.*, 15 (3):N51–N60, June 2006.

[63] S. Johansson and I. Claesson. Active noise control in propeller aircraft. In *proc. Conference for the Promotion of Research in IT*, pages 111–120, Ronneby, 2001.

[64] A. T. John Doyle, Bruce Francis. *Feedback Control Theory*. Macmillan Publishing Co. 1990, 1990.

[65] L. C. Kammer, R. R. Bitmead, and P. L. Bartlett. Direct iterative tuning via spectral analysis. *Automatica*, 36(9):1301–1307, 2000.

[66] B. Kavlicoglu, F. Gordaninejad, C. Evrensel, A. Fuchs, and G. Korol. A semi-active, high-torque, magnetorheological fluid limited slip differential clutch. *Journal of Vibration and Acoustics*, 128(5):604–610, 2006.

[67] K. Kido. Reduction of noise by use of additional sound sources. pages 647–650, Sendai, 1975. Proceedings Internoise.

[68] B. I. Kim, S. S. Heragu, and R. J. Graves. A hybrid scheduling and control system architecture for warehouse management. *IEEE trans. on robotics and automation*, 19(6): 991–1001, 2003.

[69] S.-M. Kim, S. J. Elliott, and M. J. Brennan. Decentralized control for multichannel active vibration isolation. *Control Systems Technology, IEEE Transactions on*, 9:93 – 100, Jan. 2001.

[70] R. Kosut. *Model Indetification and Adaptive Control: From Windsurfing to Telecommunications*. Springer-Verlag, 2001.

[71] R. Kosut, G. C. Goodwin, and M. Polis. Special issue on system identification for robust control design. *IEEE Trans. on Automatic Control*, 37(7):899–1008, 1992.

[72] R. L. Kosut and B. D. O. Anderson. Uncertainty model unfalsification. In *Proc. 997 CDC,*, San Diego, CA, 1997.

[73] R. L. Kosut, M. K. Lau, and S. P. Boyd. Set-membership identification of systems with parametric and nonparametric uncertainty. *IEEE Trans. Automat . Contr*, 37(7):929–941, 1992.

[74] V. Kota and M. Wright. Wake generator control of inlet flow to cancel flow distortion noise. *Journal of Sound and Vibration*, 295:94–113, 2006.

[75] T. Kouno, H. Ohmori, and A. Sano. New direct adaptive active noise control algorithms in case of uncertain secondary path dynamics. In *American Control Conference. Proceedings of the 2002*, volume 3, pages 1767 – 1772, May 2002.

[76] K. Kowalczyk, H.-J. Karkosch, P. Marienfeld, and F. Svaricek. Rapid control prototyping of active vibration control systems in automotive applications. In *Computer-Aided Control Systems Design, 2006 IEEE International Symposium on*, pages 2677 – 2682, Oct. 2006.

[77] L. Kraft and J. Pallotta. Real-time vibration control using cmac neural networks with weight smoothing. In *American Control Conference. Proceedings of the 2000*, volume 6, pages 3939 – 3943, Chicago, USA,, June 2000.

[78] B. C. Kuo and F. Golnaraghi. *Automatic Control Systems*. Wiley, 8nd edition, 2002.

[79] M. K. Kwak. Fuzzy-logic based vibarion suppression control experiments on active structures. *J. Sound and Vibration*, 191:15–28, 1996.

[80] K.-J. Lan, J.-Y. Yen, and J. A. Kramar. Sliding mode control for active vibration isolation of a long range scanning tunneling microscope. *Review of Scientific Instruments*, 75(11): 4367–4373, 2004.

[81] H.-J. Lee, Y.-C. Park, C. Lee, and D. H. Youn. Fast active noise control algorithm for car exhaust noise control. *Electronics Letters*, 36:1250 – 1251, July 2000.

[82] Y. Lesperance. Foundations of a logical approach to agent programming. In *Lecture Note in AI (LNAI)*, volume 1037, pages 331–346. Springer-Verlag, 1996.

[83] J. Lin. An activecpassive absorber by using hierarchical fuzzy methodology for vibration control. *Journal of Sound and Vibration*, 304:752–768, Jul. 2004.

[84] J.-Y. Lin and Z.-L. Luo. Internal model-based LQG/H_∞ design of robust active noise controllers for an acoustic duct system. *Control Systems Technology, IEEE Transactions on*, 8:864 – 872, 2000.

[85] L. Ljung. Model validation and model error modeling. In *The Astrom Symposium on Control*, Lund, Sweden.

[86] S. A. Long, Z. Q. Zhu, and D. Howe. Effectiveness of active noise and vibration cancellation for switched reluctance machines operating under alternative control strategies. *Energy Conversion, IEEE Transactions on*, 4:792 – 801, Dec. 2005.

[87] P. Lueg. Process of silencing sound oscillations, U.S. pantent 043,416, June 1934.

[88] K. Ma and M. Ghasemi-Nejhad. Simultaneous precision positioning and vibration suppression of smart structures - adaptive control methods and comparisons. In *Decision and Control, Proceedings of 42nd IEEE Conference on*, volume 3, pages 2204 – 2209, Dec. 2003.

[89] P. Marmarelis and V. Marmarelis. *Analysis of Physiological Systems*. Plenum, New York, 1978.

[90] J. Marzbanrad, G. Ahmadi, H. Zohoor, and Y. Hojjat. Stochastic optimal preview control of a vehicle suspension. *Journal of Sound and Vibration*, 274:701–724, 2004.

[91] J. D. McIntosh. Active noise cancellation aircraft headset system. Us patent 6278786, Telex Communications, Inc., 2001.

[92] T. Meurers. *Self-tuning and adaptive controllers for active sound and vibration control*. PhD thesis, School of Engineering Science, University of Southampton, 2002.

[93] T. Meurers and S. M. Veres. Iterative design for vibration attenuation. *Int. J. Acoustics and Vibration*, 4(2):79–83, 1999.

[94] T. Meurers, S. M. Veres, and S. J. Elliott. Frequency selective feedback for active noise control. *Control System Magazine, IEEE*, 22:32 – 41, Aug. 2002.

[95] T. Meurers, S. M. Veres, and A. C. H. Tan. Model-free frequency domain iterative active sound and vibration control. *Control Engineering Practice*, 11(0):1049C1059, 2003.

[96] S. Miyagi and H. Sakai. Mean-square performance of the filtered-reference/ filtered-error LMS algorithm. *Circuits and Systems I: Regular Papers, IEEE Transactions on, [see also Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on]*, 52:2454 – 2463, Nov. 2005.

[97] D. R. Morgan. An analysisi of multiple correlation cancellation loops with a filter in the auxiliary path. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 28:454–467, 1980.

[98] J. P. Muller, M. Pitschel, and M. Thiel. Modelling reactive behaviour in vertically layered agent architecture. In *Lecture Note in AI (LNAI)*, volume 890, pages 261–276, 1995.

[99] D. S. Nelson, S. C. Douglas, and M. Bodson. Fast exact adaptive algorithms for feedforward active noise control. *International Journal of Adaptive Control and Signal Processing*, 14:643 – 661, Aug. 2000.

[100] R. T. J. O'Brien, J. M. Watkins, G. E. Piper, and D. C. Baumann. H_∞ active noise control of fan noise in an acoustic duct. In *American Control Conference, Proceedings of the 2000*, volume 5, pages 3028 – 3032, 2000.

[101] H. F. Olson. Electronic control of noise, vibration and reverberation. *Journal of Acoustical Society of America*, 28:966–972, 1956.

[102] H. F. Olson and E. G. May. Electronic sound absorbers. *Journal of Acoustical Society of America*, pages 1130–1136, 1953.

[103] A. V. Oppenheim and A. S. Willsky. *Signals and Systems*. Prentice Hall, 2nd edition, 1996. ISBN 0138147574.

[104] R. H. J. M. Otten and L. P. P. P. van Ginneken. *The Annealing Algorithm*. Kluwer Academic Pulisher, 1989.

[105] H. OU and W. Zhang. Urban intelligent traffic control system based on multi-agent technology. *ACTA ELECTRONICA SINICA*, 28:52–55, 2000.

[106] P. M. Pardalos and J. B. Roseb. *Contstrained Global Optimization: Algorithms and Applications*. Springer-Verlag, 1987.

[107] S. Pigg and M. Bodson. Rejection of periodic disturbances with adaption to unkwon systems. In *ECC'07*, pages 2477–2483, 2007.

[108] K. Poolla, P. Khargonnekar, A. Tikku, J. Krause, and K. Nagpal. A time-domain approach to model validation. *IEEE Trans. Aut. Contr*, 39(5):951–959, 1994.

[109] A. Preumont. *Vibration Control of Active Structures, an Introduction*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

[110] N. Qian. On the momentum term in gradient descent learning algorithms. *Neural Networks*, 12(1):145–151, 1999.

[111] Z. Qizhi and J. Yongle. Active noise hybrid feedforward/feedback control using neural network compensation. *Journal of Vibration and Acoustics*, 124(1):100–104, Jan. 2001.

[112] S. J. Qu Wenzhong and Q. Yang. Active control of vibration using a fuzzy control method. *Journal of Sound and Vibration*, 275:917–930, Aug. 2004.

[113] P. Raghavan, A. Lad, and S. Neelakandan. *Embedded Linux System Design and Development*. CRC Press, 2005.

[114] S. Rao and M. Georgeff. BDI agents: From theory to practice. In E. V. Lesser, editor, *Proc. of the 1st International Conference on Multi-Agent Systems*, pages 312–319, San Francisco, CA, USA, 1995. AAAI Press.

[115] S. Rao and M. P. Georgeff. An abstract architecture for rational agents. In *Proc. of Knowledge Representation and Reasoning (KR&R-92)*, pages 439–449, 1992.

[116] L. Rayleigh. *The Theory of Sound Vol II, Chapter XIV: Two Sources of Like Pitch; Points of Silence; Experimental Methods*. MacMillan & Co., London, 1877.

[117] J. Richter. *Programming Applications for Microsoft Windows*. Microsoft Press, fourth edition, 1999.

[118] C. F. Ross and A. J. Langley. Active vibration control system for aircraft. Us patent 5568557, Noise Cancellation Technologies, Inc., 1996.

[119] W. J. Rugh. *Nonlinear System Theory - The Volterra/Wiener Approach*. The Johns Hopkins University Press, 1981. ISBN O-8018-2549-0.

[120] M. E. Russinovich and D. A. Solomon. *Microsoft Windows Internals*. Microsoft Press, fourth edition, 2006.

[121] M. G. Safonov and T. C. Tsao. The unfalsified control concept: A direct path from experiment to controller. In B. A. Francis and A. R. Tannenbaum, editors, *Feedback Control, Nonlinear Systems and Complexity*, pages 196–214. New York: Springer-Verlag, 1995.

[122] R. Sajeeb, C. Manohar, and D. Roy. Use of particle filters in an active control algorithm for noisy nonlinear structural dynamical systems. *Journal of Sound and Vibration*, 306: 111–135, 2007.

[123] M. V. Sandor and S. W. Derek. *Synergy and duality of identification and control*. Taylor & Francis, 2000.

[124] S. Sastry and M. Bodson, editors. *Adaptive Control: Stability, Convergence, and Robustness*. Prentice-Hall,, New Jersey, USA, 1994.

[125] M. Schetzen. *The Volterra and Wiener Theories of Non-linear Systems*. New York: John Wiley & Sons, 1980. ISBN 0-471-04455-5.

[126] D. W. E. Schobben and R. Aarts. Personalized multi-channel headphone sound reproduction based on active noise cancellation. *Acta Acoustica*, 91(3):440–450, 2005.

[127] N. Sellen, M. Cuesta, and M.-A. Galland. Noise reduction in a flow duct: Implementation of a hybrid passive/active solution. *Journal of Sound and Vibration*, 297:492–511, 2006.

[128] S. Shaffer and C. Williams. The filtered error LMS algorithm. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '83.*, volume 8, pages 41 – 44, April 1983.

[129] R. S. Smith and M. Dahleh. The modeling of uncertainty in control systems. In *Lecture Notes in Control and Information Sciences*, volume 192. Springer-Verlag, 1994.

[130] S. D. Snyder and N. Tanaka. Active control of vibration using a neural network. *IEEE Trans. Neural Networks*, 6:819–828, 1995.

[131] S. D. Sommerfeldt and J. Tichy. Adaptive control of a two-stage vibration isolation mount. *The Journal of the Acoustical Society of America*, 88:938–944, Aug. 1990.

[132] U. Stoebener. Active vibration isolation for highly sensitive measurement equipment. In *Adaptronic Congress*, Gottingen, May 2006.

[133] J. C. Sun, X. G. Wang, and F. J. Xi. Sliding mode active vibration control of circular saws. In *Control Applications, Proceedings of the 2000 IEEE International Conference on*, pages 953 – 958, Sept. 2000.

[134] J. C. Sun, X. G. Wang, and F. J. Xi. Sliding mode active vibration control of circular saws. In *Control Applications. Proceedings of the 2000 IEEE International Conference on*, pages 953 – 958, Sept 2000.

[135] C. Tan and H. Tachibana. Nonlinearity-tolerated active noise control using an artificial neural network. In *Applications of Signal Processing to Audio and Acoustics. 1997 IEEE ASSP Workshop on*, page 4, Aug. 1997.

[136] L. Tan and J. Jiang. Adaptive volterra filters for active control of nonlinear noise processes. *Signal Processing, IEEE Transactions on*, 49:1667 – 1676, August 2001.

[137] T. Tao and K. Frampton. Experiments on distributed active vibration control of a simply supported beam. *Journal of Smart Materials and Structures*, 15:1858–1862, 2006.

[138] O. Tokhi and S. M. Veres, editors. *Active sound and vibration control: theory and applications*. IEE, 2002.

[139] S. Veres. Unfalsification based iterative control design: a basic scheme. *International Journal of Control*, 78:231–252, 1999.

[140] S. Veres and J. Luo. A class of BDI agent architectures for autonomous control. In *Proceedings of the 43rd IEEE conference on decision and control*, volume 5, pages 4746–4751, Paradise Island, Bahamas, 2004. Institute of Electrical and Electronic Engineers.

[141] S. Veres and J. Luo. Formal verification of autonomous control agents. In *Proceedings of the 6th IASTED international conference: intelligent systems and control*, pages 140–144. International Association of Science and Technology for Development, 2004.

[142] S. M. Veres. Adaptive harmonic control. *Int. J. Control*, 74(12):1219–1225, 2001.

[143] S. M. Veres. Self-tuning control by model unfalsification (part ii). *Int. J. of Control*, 173: 1560–1571, 2000.

[144] S. M. Veres. Self-tuning control by model unfalsification(I). *Int. J. of Control*, 173: 1548–1559, 2000.

[145] S. M. Veres. Iterative identification and control redesign via model unfalsification: a basic scheme. *Int. J. of Control*, 72:887–903, 1999.

[146] S. M. Veres and H. Hjalmarsson. Tuning for robustness and performance using iterative feedback tuning. In *42nd IEEE Conference on Decision and Control.*

[147] J. Verschelde. *Homotopy Continuation Methods for Solving Polynomial Systems*. PhD thesis, Katholieke Universiteit Leuven, Leuven, Belgium, May 1996.

[148] G. Vinnicombe. Frequency domain uncertainty and the graphy topology. *IEEE Trans. Aut. Contr*, 38(9):1371–1383, 1993.

[149] V. Volterra. *Theory of Functionals and of Integral and Integro-differential Equations*. Dover, New York, 1958.

[150] E. Wan. Adjoint LMS: An efficient alternative to the filtered-x LMS and multiple error LMS algorithms, May. 1996.

[151] Y. Wan, T. J. Dodd, and R. F. Harrison. Identification of infinite degree volterra series in the time and frequency domains. In *2005 IFAC World Congress*.

[152] H. Wang. *Bounded Dynamic Stochastic Distributions Modelling and Control*. Springer-Verlag (London) Ltd, 2000.

[153] L. F. Wang. Fault-tolerant design in a networked vibration control system. In *Industrial Electronics Society. (IECON '03) The 29th Annual Conference of the IEEE*, volume 3, pages 2823 – 2828, Nov. 2003.

[154] J. Werner, J. Sotelo Jr, R. Lima, and T. Fogarty. Active noise control in ducts using genetic algorithms. In *ACTIVE 2002*, page 12, Southampton, UK, July 2002.

[155] J. E. F. Williams. Noise, anti-noise and fluid flow control. *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, pages 821–832, 2002.

[156] R. Woodley, J. P. How, and R. L. Kosut. Direct unfalsified controller design - solution via convex optimization. In *ACC*, San Diego, CA, 1995.

[157] R. Woodley, R. L. Kosut, and J. P. How. Uncertainty model unfalsification with simulation. In *Proc. 1998 ACC*, pages 2754–2755, Philadelphia, PA, 1998.

[158] W. Wu, J. Yuan, and L. Cheng. Multi-high-frequency perturbation effects on flow-induced vibration control. *Journal of Sound and Vibration*, 305:226–242, 2007.

[159] H. Xia and S. M. Veres. Improved efficiency of adaptive robust control by model unfalsification. *Automatica*, 35:981–986, 1999.

[160] D. Xiaoming, Y. Tao, and S. Huihe. A study of hybrid control based on $h - \infty$ synthesis technique of active noise control. In *Decision and Control. Proceedings of the 40th IEEE Conference on*, volume 3, pages 2586 – 2587, Dec. 2001.

[161] X. Xie and F. Ding. *Adaptive Control System*. Tsinghua Uninversicty Press, 2002.

[162] Z. Yang and D. L. Hicks. Active noise attenuation using adaptive model predictive control. In *Intelligent Signal Processing and Communication Systems. (ISPACS 2005). Proceedings of 2005 International Symposium on*, pages 241 – 244, Dec. 2005.

[163] J.-Y. Yen, K.-J. Lan, and J. A. Kramar. Active vibration isolation of a large stroke scanning probe microscope by using discrete sliding mode control. *SENSORS AND ACTUATORS A: PHYSICAL*, 121(1):243–250, 2005.

[164] Z. Ying, Y. Ni, and J. Ko. A bounded stochastic optimal semi-active control. *Journal of Sound and Vibration*, 304:948–956, 2007.

[165] T. C. L. Yuan H. Guana and W. S. S. Jr. Experimental study on active vibration control of a gearbox system. *Journal of Sound and Vibration*, 282:713–733, April 2005.

[166] A. Z., E. M.M., and A. A.A. Active vibration control of a flexible rotor. *Computers and Structures*, 53(3):499–511, 1996.

[167] W. Zhang and G. Meng. Active vibration control of micro-cantilever beam in MEMS. In *Intelligent Mechatronics and Automation. Proceedings. 2004 International Conference on*, pages 272– 276, Aug 2004.

[168] C. Zhao, L. Chen, and D. Chen. Semi-active static output feedback variable structure control for two-stage vibration isolation system. *Journal of Vibration and Acoustics*, 128 (5):627–634, 2006.

[169] D. Zhou, V. DeBrunner, L. S. DeBrunner, J. D. Baldwin, M. Ta, J. Fuller, Y. Wang, P. Wang, T. Hohenberger, S. Pelot, and L. Zuniga. Semi-active control algorithms for a smart shock absorber. In *Intelligent Vehicles Symposium. Proceedings. IEEE*, pages 813 – 818, June 2005.

[170] K. Zhou and J. C. Doyle. *Essentials of Robust Control*. Prentice Hall, 1st edition, 1997.

[171] S. Zhou and J. Shi. Active balancing and vibration control of rotating machinery: A survey. *The Shock and Vibration Digest*, 33(5):361–371, 2001.