Technical University of Denmark

Identification for Control

Developments in the Iterative Feedback Tuning Framework

Huusom, Jakob Kjøbsted; Jørgensen, Sten Bay; Poulsen, Niels Kjølstad

Publication date:
2008

Document Version
Publisher's PDF, also known as Version of record

Identification for Control

Developments in the Iterative Feedback Tuning Framework

Ph.D. thesis

Jakob Kjøbsted Huusom

August, 2008

Computer Aided Process Engineering Center
Department of Chemical and Biochemical Engineering
Technical University of Denmark

ISBN-10: 87-91435-88-9
ISBN-13: 978-87-91435-88-1
Printed by J & R Frydenberg A/S, Copenhangen, Denmark

Preface

This dissertation is submitted to the Technical University of Denmark (DTU) in partial fulfillment of the requirements for the degree of Doctor of Philosophy (Ph.D.) in Chemical Engineering. This research project has been running from February 2005 until August 2008. It has primarily been carried out at the Computer Aided Process Engineering Center (CAPEC) at the department of Chemical and Biochemical Engineering in collaboration with the department of Informatics and Mathematical Modeling, both at the Technical University of Denmark. The project has included a period at the department of Automatic Control at the Royal Institute of Technology, Sweden. The work has been financed by a research grant from the Technical University of Denmark.

I would like to acknowledge my supervisors Professor Sten Bay Jørgensen and Associate Professor Niels Kjølstad Poulsen for their collaboration, guidance and inspiring conversations throughout this project. I also owe a special thanks to Professor Håkan Hjalmarsson from the Royal Institute of Technology for hosting me in his group and providing some new perspective to my work.

Throughout my work, several people have offered their help, provided useful criticism and a few opportunities of direct collaboration have been established. To all these people I am very grateful. I would like to take this opportunity to give special thanks for her collaboration to Paloma Andrade Santacoloma during her Master thesis work. I also owe special thanks to Professor John Villadsen for taking time and effort to give me criticism on my research and his role in providing me with a position at the University at Trinidad and Tobago, for teaching process control to chemical engineering students. A position which has been undertaken during this Ph.D. project. This has not only improved my qualifications for my future career but also indirectly developed me as a researcher and improved my skills in scientific communication. Last but not least I would like thank all the good colleagues I have met during my work and especially in CAPEC. We have had some very good conversations, experiences and enjoyed relations which all have their root in our work environment, but often go beyond academia. I sincerely hope that these relation will survive "Post Doc", despite the geographic constraints.

Jakob Kjøbsted Huusom

Kgs. Lyngby, August, 2008

Abstract

The present thesis is concerned with optimization of the closed loop performance of controlled industrial processes. This is achieved through iterative schemes where input/output data is collected from the process during closed loop operation. A presentation of methods for achieving this iterative performance enhancement is given with a clear distinction between model based and data driven approaches. Special attention has been given to simple data driven strategies and the Iterative Feedback Tuning method in particular where a detailed study of methodological developments and tuning properties are given. Based on this analysis new developments for the Iterative Feedback Tuning method has been proposed.

In order to extend the application of this data driven tuning approach, the potential of Iterative Feedback Tuning has been analyzed and tested for control structures where this tuning method were novel. Results has been presented which show that the method is applicable for the nonlinear inventory control law and for a state space control system with state observes. For inventory control the proposal of applying a novel tuning method for the free parameters in the control law was interesting since classical tuning rules do not generally apply. The potential of the method was successfully illustrated by tuning step responses on a multivariable implementation of level control for a pilot scale four tank system. For the state space system analytical solutions to the feedback and the observer gain exist based on a plant model estimate. Tuning may therefore be relevant in case a mismatch exists between the true system and the model estimate used in the control design. The application of data driven tuning is interesting since most data driven approaches are focused on systems described by transfer functions. It is shown that the Iterative Feedback Tuning can be applied for tuning, and that the gains converge to the known analytical solutions.

A general disadvantage in using the Iterative Feedback Tuning method is that a large number of plant experiments may be required, hence the rate of convergence is an important issue. Slow convergence rate is often seen when tuning for disturbance rejection due to insufficient excitation of the system. It is proposed to improve the rate of convergence by utilizing external perturbation signals during the data acquisition. External perturbations will affect the operating condition and hence the performance cost function. The main idea is to shape the curvature of the cost function, rendering it less sensitive to noise, without introducing too much bias with respect to the optimum of the unperturbed problem. An algorithm which balance this in an optimal sense has been proposed and given the name Perturbed Iterative Feedback Tuning. It is further shown that for minimum variance control an optimal perturbation signal design exists which does not bias the optimization.

Resumé

Denne afhandling omhandler optimering af lukket sløjfe opførselen af regulerede industrielle processer. Denne optimering opnås gennem en iterativ metode, hvor processens input og udgang måles under feedbackregulering. Der er givet en præsentation af metoder til optimering ved iterative algoritmer, hvor disse er klassificeret i modelbaserede og datadrevene strategier. I særdeleshed er "Iterative Feedback Tuning" blevet behandlet, og et detaljeret studie af denne algoritmes udvikling og potentiale er givet. Baseret på denne analyse er ny forskningsresultater udviklet der bygger på "Iterative Feedback Tuning" algoritmen.

For at udvide anvendelsesområdet for den datadreven tunings algoritme er potentialet for Iterative Feedback Tuning blevet analyseret og testet, for reguleringsstrukturer hvor dennes anvendelse er ny. Resultater er blevet præsenteret, der viser at algoritmen kan anvendes på både den ulineare "inventory control" reguleringsstrategi og for reguleringssløjfer baseret på en tilstandsbeskrivelse med ufuldstændig tilstandsinformation, dvs. at estimation af tilstandene er en del af regulatoren. Specielt for "inventory control" har denne nye anvendelse til tuning af de frie parameter i regulatoren et potentiale, siden klassiske metoder ikke er generelt anvendelige for denne ulineære strategi. Tuningsmetoden er vist at kunne fungere og er eksemplificeret ved tuning af en multivariable niveauregulerings sløjfe på et pilotskala anlæg af et firetank system. For reguleringssløjfer på tilstandsform findes analytiske løsninger for de optimale værdier af forstærkningen i tilbagekoblingen og for tilstands estimationen. Datadreven tuning kan derfor have et potentiale når, der er uoverensstemmelse mellem det sande system og den estimerede model der anvendes til beregning af regulatoren. Datadreven tuning er også interessant for dette system, siden de fleste datadrevne metoder er udviklet til systemer på overføringsfunktions form. Det er vist, at "Iterativ Feedback Tuning" kan anvendes til tuning af systemer på tilstands form, og at forstærkningerne for tilbagekoblingen og tilstands estimatoren konvergerer til de velkendte analytiske løsninger.

En general ulempe ved anvendelse af "Iterative Feedback Tuning" er, at et større antal eksperimenter på anlægget er nødvendige og derfor bliver hastigheden hvormed metoden konvergere væsentlig. En langsom konvergenshastighed opleves ofte når reguleringssløjfen tunes til afvisning af forstyrrelser pga. mangelfuld ekscitation af systemet. Det er forslået at forbedre konvergenshastigheden ved, at benytte eksterne perturbationssignaler. Ekstern perturbation har en effekt på operationen af reguleringssløjfen og påvirker systemets kostfunktion. Den overordnede ide er, at forme krumningen af kostfunktionen således, at den bliver mindre følsom overfor stokastisk støj uden at ændre optimum for meget i forhold til det uperturberede problem, dvs. bias. En algoritme der balancere dette på en optimal vis er forslået og navngivet "Perturbed Iterative Feedback Tuning". Det er yderligger vist, at for minimalvarians regulering findes der et optimalt design for perturbationssignalet, der ikke introducerer bias i optimeringsproblemet.

Contents

Appendices

1
Introduction

The increasing competition on the global market has rendered optimizing process operation a necessity for new as well as existing production in the chemical industry. Optimized process operation means that the process is operated with higher yield, consumes less resources and is safe for both the environment and the personnel operating the plant. In practice a higher yield is often achieved by operating close to inherent process limitations where advanced control strategies and/or a very tight tuning is required. In many continuous processes the main control problem is that of disturbance rejection. Advanced control strategies are based on dynamic models for the specific process. These models play an important role in optimization of the performance of the production unit. In particular implementation of model predictive controllers (MPC) in recent years, have contributed to increased competition capabilities of the product supply chain in petrochemical and chemical industries. Identification of dynamic models plays a major role in an MPC application and it is estimated that more than 80 % of the time during an advanced control implementation is spent on modelling (Andersen *et al.*, 1991; Jørgensen and Lee, 2002). Control oriented process modelling is part of the frame work on application oriented modelling. System identification is an area that has received much attention in the last twenty years but especially within identification for control there is still room for improvement in development of systematic methods. Identification for control implies experiments where the collected data for identification are retrieved from a process operated under control i.e. in closed loop.

Traditionally process system identification has mostly been carried out in open loop. When using open loop estimation on processes with stabilizing single loop controllers, these will be considered as part of the plant, and cannot be tuned subsequently. In open loop operation the plant variations have to be monitored by the operators and controlled manually. In system identification for control it is advantageous to conduct the experiments in closed loop, and use an estimation method that estimates the parameters in the plant and the noise model within the loop from the input/output data. Besides obtaining a distinction between the plant and the controller in the loop, the plant variations are under control during the experiments. This situation may be demanded due to safety or quality concerns, or simply because the model for an operating point close to an unstable region has to been determined. Closed loop experiments are further motivated by the work of Schrama (1992b) and Hjalmarsson *et al.* (1994a) who have shown that more accurate controllers are achieved, when the control design criterion is the performance of the closed loop system. System identification for control aims at identifying a model for a continuous process or a local model set for a batch process, which yields satisfactory performance of the closed loop system given a desired operation point or trajectory.

The benefits and challenges in closed loop identification have been motivated several times in the literature. The key point is, that it is the performance of the closed loop that is object for the performance optimization (Hjalmarsson *et al.*, 1994a; Schrama, 1992b). Since then several research groups have worked on development of suitable systematic methods for handling an iterative procedure of closed loop experiments, model parameter estimation and enhanced control design (Åström and Nilsson, 1994; Van den Hof and Schrama, 1995; Gevers, 1996; Jørgensen and Lee, 2002; Gevers, 2002; Hjalmarsson, 2005).

In process control the typical control problems differ substantially at the different layers of the control hierarcy. At the sigle loop layer conventional Proportional, Integral and Differential controllers (PID) are typically - still - applied. Fairly recently alternative control strategies have been introduced also at the single loop layer. These include Inventory control (Farschman *et al.*, 1998) and MPC (Laachi and Rawlings, 2005). Both these strategies can also be extended to the multivariable case.

The many algorithms which have been proposed for optimizing the performance of the control loop fall into two categories. The first category estimates a model of the process from plant data, and uses this model in the control design assuming the certainty equivalence principle. Hence these algorithms will be referred to as indirect methods. The second category consists of the direct methods which optimize the loop performance by applying the data directly, without calculating a model estimate. The indirect methods are dominating since it is usually preferred to apply a model based control design as e.g. MPC. Having an estimate of the process model and the model uncertainty, also supplies the user with the possibility of verifying nominal and robust stability prior to implantation (Skogestad and Postlethwaite, 1996). A drawback of the indirect methods is, that an accurate model estimate is required in order to archive the designed performance. If the model estimate is inaccurate, the certainty equivalence principle constitutes too crude an assumption. A direct method is usually applied on linear, single input/single output control structures with restricted complexity such as PID controllers. The strength is that it can be applied when no process model information is available, and usually the computations are less complex which is an advantage for achieving widespread industrial use.

1.1 Model Based Control

All model based control designs aim at achieving some desired behavior or performance of the loop. Early developments in this area were based on first order, step or frequency response models which were used to select PI or PID control parameters (Ziegler and Nichols, 1942; Cohen and Coon, 1953). Assuming the plant model to be known, the closed loop behavior can be designed by selecting the parameters in the controller using the certainty equivalence principle. One criteria which is often used in simple control design is to shape the closed loop transfer function. For single input/single output systems, analytical tuning methods such as Direct Synthesis and Internal Model Control can be used to derive a transfer function for the controller (Åström and Hägglund, 1995; Garcia and Morari, 1982). These methods give the desired structure and dynamics of the resulting loop transfer function. This type of design criteria is also known as pole placement. For a large number of simple standard models, both Direct Synthesis and Internal Model Control, produce a controller transfer function which corresponds to a classical PID controller (Rivera *et al.*, 1986; Chien and Fruehauf, 1990; Chen and Seborg, 2002; Skogestad, 2003). For multivariable systems this procedure is more complex since each input/output relation corresponds to one controller in a controller matrix. Each controller will have to be designed individually.

Another approach based on optimal control is to minimize a performance cost function. A performance cost function includes the signals from the process, e.g. the input and output, to calculate an index for the loop performance level. The control design and parameterization which produce the lowest possible value of the performance cost, is referred to as an optimal control design (Anderson and Moore, 1989). Often a performance cost function is expressed as a weighed sum of the squared signal values. Let the loop consist of the system with a process model G, a noise model H and the controller C. A general quadratic cost function with penalty on the output and the control can be written as:

$$F(G,H,C) = \frac{1}{2N}\sum_{t=1}^{N} y_t^2 + \lambda u_t^2 \tag{1.1}$$

where λ determines the weight between the penalty on the output and the control effort, and N is the number of discrete data points. Given a fixed parameterization of the controller, the optimal set of parameters can be evaluated by a standard minimization algorithm. One advantage is that this criterion is applicable also in multivariable systems. The cost function will in the multivariable case be formulated on matrix form

$$F(\boldsymbol{G},\boldsymbol{H},\boldsymbol{C}) = \frac{1}{2N}\sum_{t=1}^{N} \boldsymbol{y}_t^T\boldsymbol{Q}\boldsymbol{y}_t + \boldsymbol{u}_t^T\boldsymbol{R}\boldsymbol{u}_t = \frac{1}{2N}\sum_{t=1}^{N}\|\boldsymbol{y}_t\|_{\boldsymbol{Q}}^2 + \|\boldsymbol{u}_t\|_{\boldsymbol{R}}^2 \tag{1.2}$$

where $\boldsymbol{Q}$ and $\boldsymbol{R}$ are weighting matrices. The cost function in (1.2) would be the multivariable equivalent of (1.1) if $\boldsymbol{Q} = \boldsymbol{I}$ and $\boldsymbol{R} = \lambda\boldsymbol{I}$. The design choice for this class of optimal controllers is to select a proper structure of the performance cost function and the values in the weight matrices. Different versions of the performance cost function can be formulated which insures good tracking properties, penalty on the control move etc. (Anderson and Moore, 1989)

Optimal controllers based on a performance cost function may be grouped into different classes. In case the cost function only penaltilize the output or the tracking error, the optimal controller is termed a minimum variance controller (MV) (Åström, 1970; Kučera, 1991). If the controller is designed to give optimal tracking with respect to a reference model it is referred to as a pole placement controller (PZ) (Chen, 1993). If frequency filters are applied in the cost function, it is possible to design in which frequency range high or low penalty is desired. A minimum variance controller which include frequency weighting in the cost function is referred to as a generalized minimum variance controller (GMV). For linear system and the general quadratic performance cost function in (1.2) where **y** may be considered as either the output or a tracking error, a classical formulation is referred to as the Linear Quadratic controller (LQ) (Kwakernaak and Sivan, 1972; Anderson and Moore, 1989). The optimal solution to the LQ problem is a linear feedback law, which can be evaluated from the solution to iterations in the Riccati matrix equation. It was the paradigm shift in the 1960's from describing system dynamics as differential equations in the input and output into state space descriptions, which gives a systematic handling of multivariable systems, who led to the development of LQ control, which in Willems (2007) is considered to be the main result in control theory of that time. In case of incomplete state information, the LQ controller can be combined with a Kalman filter assuming Gaussian distributed noise. This strategy is known as the Linear Quadratic Gaussian controller (LQG).

A different solution to the same type of problem, introduced by Clarke *et al.* (1987a,b), is labeled Generalized Predictive Control (GPC). In this strategy the optimal set of control signals are calculated by predicting future outputs. It is assumed that the controller will bring the system close, in some sense, to the reference signal within a fixed control horizon, and the control will stay constant over the remaining part of the prediction horizon used by the cost function. The first of the calculated controls is implemented, new measurements or estimates of the states are achieved and the problem is solved again for the next discrete time. The computations proceed as a moving horizon algorithm and Ydstie (1987) group this and similar predictive algorithms which emerged at the same time as *multi-step receding horizon control*. In case input or output constrains have to be considered as part of the control problem, the GPC or the LQ controller are conveniently extended to Model Predictive Control (MPC) (Garcia and Morari, 1982; Hallager *et al.*, 1984; Muske and Rawlings, 1993; Maciejowski, 2002). Two references in the pioneering stage of MPC control is the paper by Cutler and Ramaker (1980) and the classic application paper by Richalet *et al.* (1978). Willems (2007) states the following relation between MPC and the field of process control:

> "MPC is an area where essentially all aspects of the field, from modeling to optimal control, and from observers to identification and adaptation, are in synergy with computer control and numerical mathematics."

The close related LQ, GPC and MPC formulations are easily applied for multivariable systems since all three can be formulated using state space model formulations.

1.2 Identification for Control

Model identification for control ideally should be based on closed loop data, rather than open loop data. First of all, closed loop data means that the process is estimated under the conditions at which it has to be operated. Closed loop experiments also imply that the control loop can act on plant variations and reject disturbances during the data acquisition. Hence the risk of producing *off spec* product or unsafe operation is minimized. Several processes can not be operated in open loop, which means that stabilizing controllers would be estimated as an intrinsic part of the plat model when using open loop estimation techniques (Ljung, 1999). That implies that it would not be possible to tune these stabilizing loops based on the achieved model. Closed loop system identification circumvents this problem by taking the feedback mechanism into account in the estimation. Section 1.3 shows the estimation techniques used in closed loop system identification.

In the late 1980s, the system identification community started to shift their focus. Instead of seeking an estimate of the true system, an approximation to the true system is sought with the intended model application in mind (Gevers and Ljung, 1986; Van den Hof and Schrama, 1995; Gevers, 1996). When identifying a model for a process with a control application in mind, the aim is not to achieve the best approximation to the true system, but to achieve the best performance of the control loop. This loop contain the true system and a designed controller based on the model estimate. The performance will be affected by the experimental conditions i.e. external excitation of the system, number of data points and of cause whether the process is operated in open or closed loop. Hjalmarsson *et al.* (1996) conclude that closed loop system identification produces better performance than open loop identification, when the system is in the model class. I.e. no un-modelled dynamics. Forssell and Ljung (2000) link optimal control design and the optimal experimental design for closed loop identification. Using closed loop data in identification for control is intuitively reasonable since the process is identified under the same type of conditions as the new loop will be operated. This is in contrast to classical open loop step response experiments.

In both open and closed loop identification, it is required that the process is sufficiently excited in order to yield informative data (Ljung, 1999). Let $T(\mathbf{G}, \mathbf{C})$ be a stable feedback connection consisting of the possibly unstable system $\mathbf{G}$ and the controller $\mathbf{C}$. Due to the controller, the loop will reject disturbances and track set points. The performance of the loop can be evaluated through the performance cost function $F(\mathbf{G}, \mathbf{C})$. In order to sufficiently excite the system to reveal the dynamics, two external perturbation signals can be introduced to the system. The closed loop system with external probing signals is shown in Figure 1.1

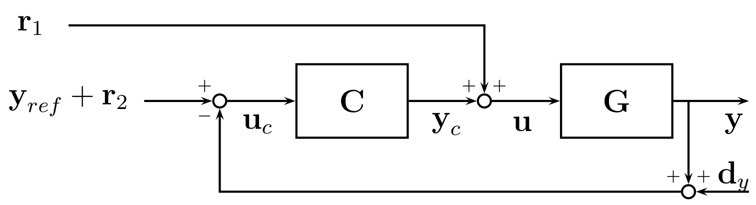


Figure 1.1. Block diagram for a feedback control loop. External perturbation signals are added to the reference and control signals in order to excite the process and obtain informative data.

The signal $\mathbf{r}_1$ introduces a deviation from the control input to the system, which will act as a known disturbance on the plant input. The second signal $\mathbf{r}_2$ acts as a known perturbation in the reference signal. This signal can therefore be used to move the process around to span a desired region of the output space. One or both of these signals can be used and designed according to the objectives of the experiments. In principle there is no difference between using either $\mathbf{r}_1$ or $\mathbf{r}_2$. If the controller is known, the filter $\mathbf{C}$ can be used such that the effect of a signal $\mathbf{r}_2$ is the same as if the filtered signal is used as $\mathbf{r}_1$. In general the choice of signals depend on the constraints and the model one intends to identify the parameters in. Given input constraints or unknown $B(q^{-1})$ parameters in a ARX like model, *Auto Regressive process with eXogenous input*, the $\mathbf{r}_1$ signal that acts on the input is advantageous. Likewise with $\mathbf{r}_2$ with respect to output constraints or unknown $A(q^{-1})$ parameters.

1.2.1 The Necessity of an Iterative Scheme

The design of optimal experimets and the optimal controller for loop performance, is a function of the unknown plant. Hence this design is infeasible which is the case for any optimal design problem in estimation, where the optimal design depends on the quantity one seek to estimate (Goodwin and Payne, 1977). The solution is to use an iterative scheme which, provided a set of conditions is fulfilled, will converge to the optimal design (Schrama, 1992b). Closed loop identification is an iterative procedure since it contains two essential elements which interact: Model estimation and control design. The estimated model will be affected by the implemented control which was operating during the data acquisition. When the estimated model is used to design a new controller, the resulting loop will perform differently from the previous. Hence data from this new loop may give a different model estimate compared to the previous. The identified plant model is used to design a new controller in order to enhance the performance of the loop. If the performance specifications are not met, repeated iterations will have to be performed according to the following scheme, until the performance is satisfactory.

- Closed loop experiment
- Estimation of a plant and noise model, $\mathbf{G}_i$ and $\mathbf{H}_i$
- Implement controller $\mathbf{C}_{i+1}$ based on the model estimates
- Evaluation of closed loop performance, $\|F(\mathbf{G}, \mathbf{H}, \mathbf{C}_{i+1})\|$

Identification in closed loop through the iterative scheme involves some inherent problems and design challenges, that need to be addressed in order to prevent divergence of the procedure (de Callafon, 1998). It must be required that the performance of the control loop is equal to or better than the performance of the loop for the previous iteration.

1.3 Model Estimation from Closed Loop Data

Three main approaches to model estimation from closed loop data exist, each with a number of advantages and disadvantages (Söderström and Stoica, 1989; Ljung, 1999).

- Direct identification
- Indirect identification

- Joint input/output identification

In direct identification, the data set $\{\mathbf{u}, \mathbf{y}\}$ is used to estimate the process model as in open loop identification. The basic principle of not having inputs that are correlated with the process noise is violated by this method. A consistent estimate is only produced by this method if the data are informative, and the model structure of the estimate contains the true model structure. That is rarely the case in practice, since it implies that a very high model order has to be chosen in order to avoid bias. The advantages of the direct estimation are that it is simple and applicable, independently of whether the controller is known and of its complexity. A disadvantage of the direct identification is, that even if the model set is large enough to contain the true process model, $\mathbf{G}$, but the noise model, $\mathbf{H}$, is not able to contain the noise dynamics, the estimate of the process model, $\hat{\mathbf{G}}$ will be biased due to the feedback (Gevers, 1996).

In indirect identification a model is estimated using $\{\mathbf{r}_i, \mathbf{y}\}$ which prevent the problem with correlations. Given this estimate of the closed loop, an estimate for the process is deduced using knowledge of the controller as shown in equation (1.3). This method requires a known linear controller without input saturation and anti wind up.

$$\hat{\mathbf{G}}_{cl} = \frac{\hat{\mathbf{G}}}{1 + \hat{\mathbf{G}}\mathbf{C}} \tag{1.3}$$

Indirect identification can also be applied given a nonlinear controller, but that requires computation of the model output $\hat{\mathbf{y}}(t|\boldsymbol{\theta})$ as a function of the open loop parameters, $\boldsymbol{\theta}$, the known controller and past dither signal values before forming the output error criterion (Ljung, 1999).

Joint input/output identification estimates the transfer from the excitation signals $\mathbf{r}_i$ to both $\mathbf{u}$ and $\mathbf{y}$. The estimation of the two transfer functions are straigt forward since they are both open loop estimation problems. The system model is then equal to the ratio between the two transfer functions. This can be done simultaneously to mimic the direct approach or in steps e.g. by the two step method suggested by Van den Hof and Schrama (1993) or by the coprime factor method (Van den Hof *et al.*, 1995). The joint input/output method can also be utilized for a system containing an unknown nonlinear controller (Ljung, 1999).

These methods only provide an estimate of a nominal model. In de Callafon (1998) it is shown that a set of models can be obtained by estimation of the nominal model using stable coprime factorizations and the model uncertainty by considering a perturbation in the dual-Youla parameterization. The estimated nominal model can easily have a too high complexity, in order to be used directly for control design. Hence a model reduction may be necessary. Gevers (2002) shows how the bias error of the estimate has to be tuned during the iterative procedure.

1.4 Research Objective and Aim

It has been the overall objective of this research project to investigate directions in identification for control, based on closed loop techniques. This investigation was focused on technology which is amenable to process industry. The aim of the project was then to develop methods and tools which are easily applied and general for process optimization. These methods would be based on the waste amount of theory developed in the areas of system identification and control theory.

The focus has been concentrated on direct data driven optimization and particularly on the Iterative Feedback Tuning method. This method qualifies with respect to low complexity in terms of the algorithm and can, and has, been used on various systems and industrial processes. The investigation of the tuning method has been focused on achieving a detailed knowledge on the theoretical foundation of the method and identify its limitations and area of application. Based on this analysis, the research has been following two main aims which are based on the following hypotheses:

- It is possible to extend the area of application of the Iterative Feedback Tuning method for processes and control structures which are novel to this tuning procedure.
- It is possible to extend the application range of the Iterative Feedback Tuning method beyond its presents practical limitation by changing the experimental procedures which is imbedded in the method.

1.5 Thesis Organization

This thesis is organized in three main parts excluding this introduction chapter, the final conclusions, suggestions for future works, appendices and references. The thesis is compiled as a collection of technical reports, scientific conference and journal papers which have been produced and presented/submitted as part of this research project. Hence all chapters containing novel results which have been produced as part of this Ph.D. project will reflect the content in one separate paper containing the contribution. Each such chapter will have the same title as the respective paper and the paper abstract on the first page. The remaining pages of these chapters will contain the contribution formatted to fit this thesis. All references given in these papers which then will be part of these chapters, are collected at the end of the thesis and not after each chapter.

Part 1 - Algorithms in Identification for Control

This part of the thesis intents to provide a general introduction to optimization of control loop performance by closed loop identification and model based control synthesis. It contains well known results and serves as an introduction to the area and a background for the remaining parts of the thesis. Several approaches for control loop optimization are discussed in chapter 2, which gives a broad overview over a range of methods and classifies these in direct and indirect methods for controller tuning. Chapter 3 gives a detailed introduction to Iterative Feedback Tuning, which is the tuning algorithm investigated in this thesis. The general method is outlined and the state of the art for this tuning algorithm is discussed with references to the numerous contributions which have been published by different researchers in the field. The chapter ends by listing the contributions of this thesis related to Iterative Feedback Tuning method.

Part 2 - Iterative Feedback Tuning for More Complex Control Structures
This part of the thesis is based on two conference paper contributions, where each have been given a separate chapter. These two chapters are both exploring the Iterative Feedback tuning method to tune control structures, where application of this method has not yet be reported. Chapter 4 investigates the tuning method for the nonlinear Inventory Control structure, and chapter 5 investigates the method for tuning the feedback gain and the observer gain in a state space control structure and compares the results to classical LQG design.

Part 3 - Iterative Feedback Tuning for Disturbance Rejection
This part of the thesis is based on one journal paper contribution and a technical rapport, where each contribution has been given a separate chapter. The first contribution, which is given in chapter 6, argues for introducing external perturbation in Iterative Feedback Tuning when tuning for disturbance rejection. The second contribution in chapter 7 contains a more detailed analysis of the convergence properties of Iterative Feedback Tuning in presence of external perturbations and provides an optimal perturbation signal design.

Concluding remarks and future work are presented in chapter 8.

1.6 Publication list

In this section all the scientific documentation produced as part of the research work associated with this Ph.D. study is listed. The following is divided in journal papers, peer reviewed conference publications and other publications. The chapters which are based on the contributions in these papers are given in bold font after the reference.

Journal Papers

Jakob Kjøbsted Huusom, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. Improving Convergence of Iterative Feedback Tuning. In Press for *Journal of Process Control*. DOI information: 10.1016/j.jprocont.2008.09.004 **Chapter 6**

Peer Reviewed Conference Publications

Jakob Kjøbsted Huusom, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. Data Driven Tuning of State Space Controllers with Observers. Submitted for the European Control Conference, Budapest, Hungary, Aug. 23-26, 2009. **Chapter 5**

Jakob Kjøbsted Huusom, Håkan Hjalmarsson, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. (2008) Improving Convergence of Iterative Feedback Tuning using Optimal External Perturbations. In *Proceedings of the 47th IEEE Conference on Decision and Control*, pages 2618 – 2623. **Chapter 7**

Jakob Kjøbsted Huusom, Paloma Andrade Santacoloma, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. (2007). Data driven tuning of inventory controllers. In *Proceedings of the 46th IEEE Conference on Decision and Control*, pages 4191 – 4196. **Chapter 4**

Jakob Kjøbsted Huusom, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. (2007). Iterative Controller Tuning for Processes with Fold Bifurcations. In *Proceedings for ESCAPE 17*, pages 835 – 840.

Jakob Kjøbsted Huusom, Denis Bonné, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. (2005). Process Identification Challenges for Nonlinear Model Predictive Control In *Proceedings of International Workshop on Assessment and Future Directions of Nonlinear Model Predictive Control*, pages 451 – 458.

Other Publications

Jakob Kjøbsted Huusom, Håkan Hjalmarsson, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. (2008). A Design Algorithm using External Perturbation to Improve Iterative Feedback Tuning Convergence. Technical Report PEC08-16, CAPEC and IMM at Technical University of Denmark and Automatic Control at Royal Institute of Technology, Sweden. http://orbit.dtu.dk/All.external?recid=221002. **Chapter 7**

Part I

Algorithms in Identification for Control

2

Directions in Identification for Control

Given the challenge of designing a suitable controller for a particular process, one is faced with a series of choices. In this chapter, it is assumed that an appropriate selection of actuators and process measurements has been made, and a control structure has been selected. The task of model identification, control design and tuning are discussed with references to a number of well known methods within identification for control. Some of these methods are tailored for a specific control structure, while others offer more flexibility. It is assumed that the data used for estimation, tuning etc. in the following algorithms is acquired during closed loop operation.

The first choice to be made is to select either a direct or a indirect i.e. model based method. A model based method involves estimation of unknown parameters in either a first principle engineering model or in some black box model structure. Given the process insight provided by the model, a controller can be designed. A direct method utilizes the data directly in tuning the control parameters, without the use of a model. In general, better performance of the control loop can be achieved from a model based method since the dynamic behavior of the process can be exploited in the control design. In case sufficiently detailed knowledge of the process dynamics is not available, using a direct tuning method may be a reasonable approach. In the following, algorithms for model based and direct controller optimization will be described separately.

2.1 Process Identification and Model Based Control Design

The joint problem of estimating parameters in some known model structure and design a model based controller becomes quite complex, when treated in an optimal sense. Gevers (1996) illustrates this complexity by dividing the task of identification for control into three categories of decreasing complexity.

Dual Control: The most ideal solution handles the dual problem of identification and control by posing an optimal control problem in which the updating of unknown parameters is embedded. Such a formulation will give an optimal handling of the trade off between tight control and the need for excitation for parameter estimation.

Optimal Experimental Design: The experimental design problem is focused on performing the identification such that the achieved performance of the loop is close to the ideal performance, based on knowledge of the true system. The optimal solution to this problem will therefore depend on the unknown process.

Identification and Model Based Control: This more realistic approach compares the achieved loop performance to the optimal nominal design, which is derived from a system model.

The dual control problem, which is the optimal solution to a stochastic control problem with unknown and possible time-varying parameters, is a computationally very hard problem. It was introduced in the 1960's by Fel'dbaum (1965), but since most general dual control problems can not be solved analytically, most attention has been given to suboptimal problems (Lindoff *et al.*, 1999; Li *et al.*, 2008). Both the dual control problem and optimal experimental design are too idealized to have a feasible solution for a general problem. The main interest of this section will focus on methods of identification and model based control.

The iterative procedure of model estimation, model based control synthesis and performance evaluation mentioned in Section 1.2.1 requires some preliminary choices. First a model structure for the plant and possibly the type of noise have to be selected. How and whether to include a description of the model uncertainty is important for the final robustness of the system. This choice is also related to the norms used for the identification criterion and in the control design. E.g. the $\mathcal{L}_2$-norm used in prediction error estimation of unknown parameters in a fixed structure versus the ∞-norm or the $G\Delta$ structure to represent an unstructured uncertainty description (Skogestad and Postlethwaite, 1996). These choices of estimation and control design norms affect the complexity of the resulting problem. It also determines whether a global optimization of the robust performance is achieved or a local optimization of the achieved performance. Optimal robust performance is a very attractive property, and $\mathcal{H}_\infty$ theory has been dominating in the last 15 years of the $20'th$ century (Willems, 2007). Selecting the less computational demanding optimization of nominal performance may yield better performance of the achieved loop, at the expense of guarantied robustness.

In the following a few examples of algorithms are presented which offer different solutions to this trade off. The robustness and hence the complexity of the methods will be successively increasing.

2.1.1 Optimizing Nominal Performance

In the general problem setting a model structure for the process is selected e.g. a nonlinear first principles engineering model or a standard input/output model such as the ARMAX model, a *Auto Regressive, Moving Average process with eXogenous input*. Several results on parameter estimation for specific model structures have been reported (Sjöberg *et al.*, 1995; Gopaluni *et al.*, 2004; Milanese and Taragna, 2005; Qin *et al.*, 2005). Ljung (1999) is a key reference for prediction error estimation of parameters in linear time invariant system, as the ARMAX structure. This class of linear model structures is often used to approximate processes in a control loop. Given a process model, the control design is performed using the certainty equivalence principle. The result is a controller which will minimize the performance of a designed loop consisting of the identified model, $F(\hat{G}, \hat{H}, C^{opt}(\hat{G}, \hat{H}))$. The actual performance of the loop will be $F(G, H, C^{opt}(\hat{G}, \hat{H}))$. The iterative procedure of model identification from closed loop data and subsequent re-design of the controller should ideally render the actual performance converge to the designed performance. An advantage of this approach is that the true model does not have to belong to the model set for the procedure to converge (Gevers, 1996). The model order can therefore be selected based on the trade off between bias and variance error of the plant model estimate.

In order to illustrate the steps involved in one iteration of identification for model based control, a basic example is outlined. First of all it is assumed that the process is currently working in closed loop, and performance optimization is needed since the current performance is dissatisfactory. The performance could be evaluated based on the cost function (1.1).

$$F(G, H, C) = \frac{1}{2N} \sum_{t=1}^{N} y_t^2 + \lambda u_t^2 \tag{1.1}$$

It is assumed that the process can be approximated by the ARMAX model structure

$$A(q^{-1})y_t = q^{-k}B(q^{-1})u_t + C(q^{-1})e_t \quad e_t \in \mathcal{N}(0, 1) \tag{2.1}$$

where $A(q^{-1}), B(q^{-1})$ and $C(q^{-1})$ are polynomials in the backshift operator q^{-1} which shifts the discrete time index as $\psi_{t-i} = q^{-i}\psi_t$. k is the discrete time delay through the process. The $A(q^{-1})$ and $C(q^{-1})$ polynomials are monic, hence their leading coefficient is equal to one. The unknown system parameters in the $A(q^{-1}), B(q^{-1})$ and $C(q^{-1})$ polynomials can the be collected in the vector $\boldsymbol{\theta}$ which has to be estimated. Performing the estimation using direct estimation and the prediction error framework means that the closed loop data and the model is used to generate a sequence of one step ahead prediction errors

$$\epsilon_t(\boldsymbol{\theta}) = y_t - \hat{y}_t(\boldsymbol{\theta}|\mathbf{Y}_{t-1}, \mathbf{U}_{t-1}) \tag{2.2}$$

where $\mathbf{Y}_{t-1}$ and $\mathbf{U}_{t-1}$ are two vectors containing information of the in- and outputs up till time $t-1$. An optimization algorithm can be used to find the optimal set of parameters which will minimize the prediction errors. The least squares solution, i.e. the $\mathcal{L}_2$-norm, is:

$$\boldsymbol{\theta}^{opt} = \arg\min_{\theta} \frac{1}{N} \sum_{t=1}^{N} \epsilon_t(\boldsymbol{\theta})^T \epsilon_t(\boldsymbol{\theta}) \tag{2.3}$$

For a linear system description, the prediction errors can be written as a linear regression problem, and an analytical solution for least squares optimal set of parameters exists (Ljung, 1999; Kailath *et al.*, 2000). For more general model descriptions, nonlinear regression may be required, where a numerical search algorithm can be used to evaluate the parameters (Ljung, 1999; Nocedal and Wright, 1999). Given a system model a certainty equivalence control synthesis can be performed based on the performance criterion (1.1). The criterion is used to evaluate the unknown optimal set of control parameters $\boldsymbol{\rho}$ in some control structure e.g. a PID controller or a structure which use the criterion more directly as LQG or minimal variance control i.e. $\lambda = 0$. For an arbitrary control structure, predicted values of the in- and outputs can be used in a numerical scheme equivalent to the model parameter estimation problem.

$$\boldsymbol{\rho}^{opt} = \arg\min_{\rho} F(\hat{G}, \hat{H}, C(\boldsymbol{\rho})) \tag{2.4}$$

which would then produce the controller $C^{opt}(\hat{G}, \hat{H})$. For an ARMAX or a state space model structure, the parameterization of the optimal controller for the minimal variance and the LQG controllers have been derived analytically, and the control parameter estimation problem is well defined (Åström, 1970). The optimal controller transfer function for the LQG problem for the ARMAX model (2.1) is given as (Poulsen, 1997):

$$u_t = \frac{-S(q^{-1})}{R(q^{-1})} y_t \tag{2.5}$$

where the polynomials $S(q^{-1})$ and $R(q^{-1})$ are given by solving the Diophantine equation

$$P(q^{-1})C(q^{-1}) = A(q^{-1})R(q^{-1}) + q^{-k}B(q^{-1})S(q^{-1}) \tag{2.6}$$

where $P(q^{-1})$ is the solution to the spectral factorization

$$P(q^{-1})P(q) = B(q^{-1})B(q) + \lambda A(q^{-1})A(q) \tag{2.7}$$

In order to have a unique solution is must be required that the order of the $R(q^{-1})$ polynomial n_r is equal to $n_b + k - 1$ and the order of $S(q^{-1})$, n_s equal to the maximum value of $n_a - 1$ or $n_p + n_c - (n_b + k)$. Further more it must be required that the roots of $P(q^{-1})$ are not located outside the unit circle.

The steps illustrated in this section would be similar if other process model descriptions and control designs where chosen. Multivariable industrial processes are often being approximated by a linear state space description which can be estimated using subspace identification (Verhaegen, 1994; Van Overschee and De Moor, 1994). For multivariable systems it is an advantage to select a control law which take interactions between the multiple actuators into account, e.g. LQ or MPC strategies (Luyben *et al.*, 1998). These strategies require full state information which may not be measured for an arbitrary black box state space description. This problem can be circumvented by introducing a Kalman filter for state estimation in the control law. The model estimate will then be used in both the construction of the state estimator and the controller, and the loop performance will depend on both.

2.1.2 Frequency Weighted, $\mathcal{L}_2$ Identification and $\mathcal{H}_2$ Control Design

One degree of freedom which the designer can use in the model estimation and in the control design is to include a stable filter in the criteria. The frequency weighting which is introduced by the data filter can be used to eliminate the effect of undesired high frequency noise from data (Ljung, 1999).

A data driven identification/control design scheme is presented by Zang *et al.* (1995), that iteratively aims to improve the closed loop performance. A least squares model identification is performed from closed loop data. The data are filtered in order to improve the model accuracy at those frequencies where the measures of robust stability or performance indicate that improvements are needed. A frequency weighted LQG control design is used to account for the imperfections of the estimated model. The effect of the weighting renders the controller cautious in the frequency bands where the data reflects a difference between the plant and the model.

The frequency weighted regulation objective become

$$J^C = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \left\{ \left[\left(\frac{\Phi_y}{\Phi_{y^c}} \right)^{1/2} y_t^c \right]^2 + \lambda^2 \left[\left(\frac{\Phi_u}{\Phi_{u^c}} \right)^{1/2} u_t^c \right]^2 \right\} \tag{2.8}$$

where Φ_y and Φ_u are the spectra of the signals obtained on the actual system where Φ_{y^c} and Φ_{u^c} are obtained from simulations of the closed loop system. When e.g. $\frac{\Phi_y}{\Phi_{y^c}}$ is large, it means that the model fit is poor and that disturbance rejection is not as good as expected from the designed system. This frequency will therefore be weighted more in the following control design. The method is in this way trying to compensate for the drawback with respect to robustness, which is inherent in the $\mathcal{L}_2$ identification and $\mathcal{H}_2$ control design. It achieves an attractive compromise between an infeasible ∞-norm design and the feasible 2-norm design without robustness guaranties. The Zang algorithm is not much harder to solve than the basic $\mathcal{L}_2$ identification and $\mathcal{H}_2$ control design.

A clear distinction is made between a robust stability and a performance objective in the identification step. An external perturbation signal r_t is entering the loop as a perturbation of the reference when data is being acquired for the identification step. When performance is the main issue, an excitation spectrum which is a scaled version of the noise spectrum is used, while a constant spectrum is used when the main issue is stability.

2.1.3 Parameter Uncertainty Regions

When the parameters in a fixed model structure are identified, computation of the parameter confidence region will give an indication of how far the model estimate may be from the true system. This analysis depends on whether the true system is contained in the model set Ljung (1999).

In Gevers *et al.* (2003a) a method for control oriented system identification is published and the method is illustrated in Gevers *et al.* (2003b). The derivation of the theory behind this algorithm limits the use to single input/single output, linear, time-invariant systems only. Given an estimate of a plant model, an uncertainty set (PE model set) is defined as the space spanned by the parameter α-confidence ellipsoids.

$$\boldsymbol{\theta} \in U = \{\boldsymbol{\theta} | (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T \boldsymbol{P}_{\boldsymbol{\theta}}^{-1} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) < \chi\}, \quad \alpha(q_{\boldsymbol{\theta}}, \chi) = Pr(\chi^2(q_{\boldsymbol{\theta}}) < \chi) \tag{2.9}$$

where $q_{\boldsymbol{\theta}}$ is the number of parameters in $\boldsymbol{\theta}$, and $\boldsymbol{P}_{\boldsymbol{\theta}}$ is an estimate of the covariance matrix of the parameter estimate. Results are given for necessary and sufficient conditions for a controller to stabilize all plants within the set and an upper bound for the worst case performance. These are labeled controller validation for stability and performance. A set of controllers then exits that is able to stabilize all plants within the PE model set. A control oriented identification experiment is here defined as an experiment that maximize the set of controllers that are able to stabilize the PE model set. This connection is made by evaluating the worst case chordal distance as a function of frequency or the worst case ν-gab introduced by Vinnicombe (1993). This method offers a convenient algorithm with a relative degree of robustness which is linked to the statistical properties of the model estimate through the confidence regions.

2.1.4 Unstructured Model Uncertainty

When a model is used to describe a system of higher order than the model structure, the estimate will be associated with a unstructured uncertainty. This under modeled dynamics can be described using the $G\Delta$ model formulation, where the Δ represent the uncertainty. As an example equation (2.10) shows an additive unstructured uncertainty description, where the true system G can be written of a reduced order representation G_r plus an additive term.

$$G = G_r + \Delta_{add} \tag{2.10}$$

The same form can be used to express model parameter uncertainty. Other forms as multiplicative input/output uncertainties, their inverse or a coprime factor uncertainty are given in Skogestad and Postlethwaite (1996) and Zhou and Doyle (1998). The $G\Delta$ uncertainty description is a powerful representation which allow robustness analysis and robust control synthesis.

2.1.5 Estimation with the Youla Parameterization

The Youla parameterization is based on the fractional representation of a system (Youla *et al.*, 1976; Desoer *et al.*, 1980). The model uncertainty is viewed as a coprime factor uncertainty. If the true system is stabilizable it can be parameterized as a controller based perturbation of some plant model, defined by the stable Youla parameter Q. This means that all controllers that stabilize a plant G can be given in terms of the Youla parameterization:

$$C_Q = \frac{N_c + QD}{D_c - QN} \tag{2.11}$$

where the plant, the noise model and the controller are given by the stable right coprime factorizations[1]:

$$G = ND^{-1} \tag{2.12}$$
$$H = MD^{-1} \tag{2.13}$$
$$C = N_c D_c^{-1} \tag{2.14}$$

This system description is based on the following general model

$$y_t = Gu_t + He_t \Rightarrow Dy = Nu_t + Me_t \tag{2.15}$$

[1]The coprime factorization $G = ND^{-1}$ means that N and D do not share any unstable zeros

Hansen *et al.* (1989) use this parameterization to solve the closed loop experimental design problem. This problem is transformed into an open loop estimation problem of the dual-Youla parameter. The dual-Youla parameter R produces all systems which yields a internally stable feedback connection with a controller C. S is a parameter that gives the corresponding noise dynamics. Since these two parameters can be estimated in a open loop fashion an estimate of the true plan model can be deduced given some auxiliary model $G_x = N_x D_x^{-1}$. This approach to system identification has since been known as the Hansen scheme.

$$N = N_x + D_c R \tag{2.16}$$
$$M = S \tag{2.17}$$
$$D = D_x - N_c R \tag{2.18}$$

One of the advantages of describing the system using the Youla parameters is clearly that it is guaranteed that the perturbation contains the true system, another is that the loop is guaranteed stable. Further more does the fractional approach convert the problem into an open loop identification problem. In Ansay *et al.* (1999) the signals for the identification of the two dual-Youla parameters are generated from the controller rather than from the plant, which makes it possible to estimate the parameters separately. In De Bruyne *et al.* (1998) and De Bruyne *et al.* (1999) considerations are presented for tuning the model order and a nonlinear version of the Hansen scheme.

2.1.6 Iterative Robust Performance Enhancement

In his thesis, de Callafon presents an algorithm for robust control enhancement (de Callafon, 1998). This method is based on system identification using a fractional approach to identify a plant and its uncertainty set $\mathcal{G}$ through the dual-Youla parameterization. This identification framework has been thoroughly treated in Schrama (1992a). The control design is based on μ-synthesis, which is a $\mathcal{H}_\infty$ control design (Packard and Doyle, 1993; Zhou and Doyle, 1998). Through an iterative procedure, the closed loop performance is optimized in a robust sense. Letting Γ_i and Γ_{tol} be the performance level for the i'th iteration and the tolerance respectively, the necessary demands in order to prevent divergence of the iterative scheme can be formulated as:

Identification

$$\|F(\hat{\mathbf{G}}, \mathbf{C}_i)\|_\infty \leq \Gamma_i \quad \forall \hat{\mathbf{G}} \in \mathcal{G}_i$$

Control design

$$\|F(\mathbf{G}, \mathbf{C}_{i+1})\|_\infty \leq \Gamma_{i+1} < \Gamma_i \quad \forall \mathbf{G} \in \mathcal{G}_i$$

This will be repeated until $\Gamma_{i+1} \leq \Gamma_{tol}$, if possible.

In the identification experiment the triangular inequality is used to define a control relevant identification problem. In Schrama (1992a) it is shown that

$$\begin{aligned} \big| \|F(\hat{G}, C(\hat{G}))\| - \|F(G, C(\hat{G})) - F(\hat{G}, C(\hat{G}))\| \big| &\leq \|F(G, C(\hat{G}))\| \\ \|F(G, C(\hat{G}))\| &\leq \|F(\hat{G}, C(\hat{G}))\| + \|F(G, C(\hat{G})) - F(\hat{G}, C(\hat{G}))\| \end{aligned} \tag{2.19}$$

where G is the true system, $\hat{G}$ is the nominal system model and $C(\hat{G})$ is the controller designed based on $\hat{G}$. Tight upper and lower bounds can be achieved by minimizing the performance degradation $\|F(G, C(\hat{G})) - F(\hat{G}, C(\hat{G}))\|$.

2.2 Direct methods

Opposite the indirect methods which rely on a model estimate, the direct tuning methods will adjust the controller directly based on the data acquired from the control loop. This more crude approach can be very useful when the process model structure is unknown. Direct methods may also be convenient for fine tuning of a controller, when the observed loop performance degrades slowly over time. The model based control design will always depend on the model quality. Hence modelling bias or time varying parameters may result in poor performance which can then be adjusted by direct tuning. Another advantage is that the direct tuning methods are often relatively simple and computationally easy to apply for simple control structures compared to a model based approach.

Classical tuning rules as the Ziegler-Nichols method and other step response or frequency response methods belong to the class of direct tuning methods. In this section only methods which can use closed loop data are considered and in the following three direct tuning methods will be presented. Two of these has been introduced within the last ten years.

2.2.1 Iterative Feedback Tuning - IFT

The Iterative Feedback Tuning method was suggested in the mid 1990's. It has proven to be an amenable tuning strategy and it has been a central topic in a vast number of publications. The key idea is to construct an unbiased estimated of the performance cost function gradient, based on closed loop data. This estimate is then used in a search algorithm which minimize the performance cost by tuning the control parameters. In each iteration of the Iterative Feedback Tuning, two experiments are required for one degree of freedom single input/single output controllers. One additional experiment is needed for a two degree of freedom controller. For multivariable systems the number of experiments which is required in order to form an unbiased gradient estimate grows with the complexity. The fact that numerous plant experiments have to be performed as part of the tuning is one of the main obstacles of the Iterative Feedback Tuning method. Advantages are that the method is very flexible in terms of optimization criteria and poses very few restrictions on the system. A thorough presentation of the Iterative Feedback Tuning method with the developments and applications will be given in Chapter 3.

2.2.2 Correlation-based Tuning - CbT

The Correlation-based Tuning method was proposed in Karimi *et al.* (2003). A detailed description can be found in Karimi *et al.* (2004) and in the thesis Mišković (2006). The algorithm assumes a linear time-invariant system, G, with an unknown transfer function. The system is controlled by a linear time-invariant controller,

$C(\boldsymbol{\rho})$. In this section a single input/single output system with a one degree of freedom controller will be assumed. Extension from this basic formulation is provided in Mišković (2006). Let G_d be some designed reduced order plant model, working in a feedback loop with the controller, C_d, which is designed such that the loop meets performance specifications. When both the actual and the design loop are subject to the same reference signal, the output error can be computed as the difference between the output signals of two loops.

$$\epsilon_{oe}(\boldsymbol{\rho}, t) = y(t) - y_d(\boldsymbol{\rho}, t) \tag{2.20}$$

$$= \left(\frac{C(\boldsymbol{\rho})G}{1 + C(\boldsymbol{\rho})G} - \frac{C_d G_d}{1 + C_d G_d} \right) r(t) + \frac{1}{1 + C(\boldsymbol{\rho})G} v(t) \tag{2.21}$$

since only the output from the true system will be affected by process noise. The two feedback loops and the generation of the output error are shown in Figure 2.1.

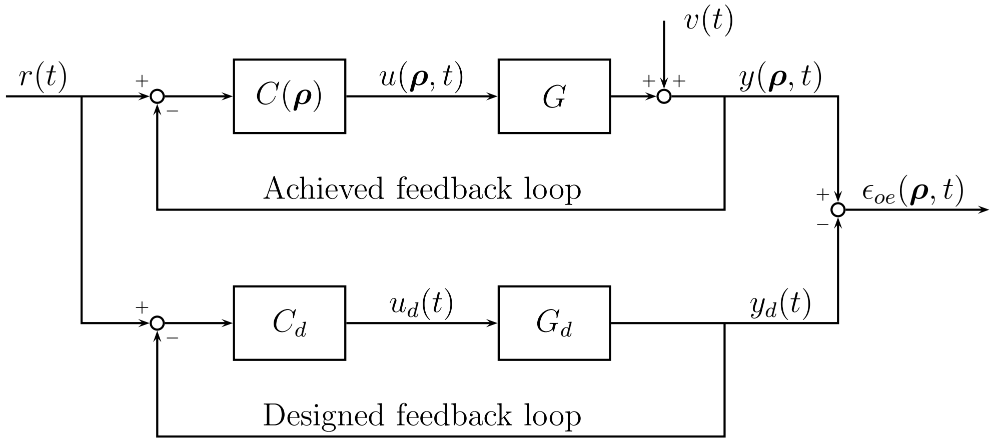


Figure 2.1. Block diagram for the achieved and the design feedback control loops used in the Correlation based Tuning algorithm.

In case the designed controller is implemented in the loop with the true plant, the output error will have a contribution from both the noise and the modelling error between G_d and the true system G c.f. equation (2.21). The main idea in the tuning method is then to adjust the controller parameters in order to de-correlate the output error with the reference signal. The analytical solution to this problem would be the controller:

$$C(\boldsymbol{\rho}) = C_d \frac{G_d}{G} \tag{2.22}$$

which depends on the unknown plant model G. Since the analytical design is infeasible a correlation equation is defined as

$$f(\boldsymbol{\rho}) = \mathrm{E}\{\bar{f}(\boldsymbol{\rho})\} = 0 \tag{2.23}$$

where $\mathrm{E}\{\cdot\}$ is the mathematical expectation and

$$\bar{f}(\boldsymbol{\rho}) = \frac{1}{N} \sum_{t=1}^{N} \zeta(t) \epsilon_{oe}(\boldsymbol{\rho}, t) \tag{2.24}$$

where N is the number of discrete data points and $\zeta(t)$ is a vector of instrumental variables which are correlated with the reference signal, but independent of the disturbance. The solution to the correlation equation will give the parameters in a decorrelating controller. This problem can be solved by a stochastic approximation procedure, and for a large number of data points using a Newton-Raphson search algorithm. For more details in solving the correlation equation and the choice of instrumental variables a discussion is provided in Mišković (2006).

The Correlation-based Tuning provide an attractive algorithm which uses an intuitively reasonable criterion to optimize the control parameters. It does not use the quadratic cost (1.1) directly, but it is left to the user to define G_d and C_d which leave some freedom to the design. In general, convergence is faster than a standard Iterative Feedback Tuning problem which require three experiments in each iteration, where Correlation-based Tuning only needs one (Karimi *et al.*, 2003). The two methods are related and have a similar type of equations for the parameter update. The Correlation-based Tuning is in its basic formulation not applicable when $r(t) = 0$, as would be the case for a disturbance rejection problem, since this reference signal would produce a trivial solution to the correlation equation. A reformulation of the method which tune the control parameters for disturbance rejection is given in Mišković *et al.* (2003). The disturbance is regarded as process noise uncorrelated with any measurement noise in the system. The aim of the tuning is to achieve a controller which can compensate for the process noise. Hence the output has to be uncorrelated with the process noise.

2.2.3 Virtual Reference Feedback Tuning - VRFT

Virtual Reference Feedback Tuning was first introduced in Campi *et al.* (2000) followed by the papers Campi *et al.* (2002) and Lecchini *et al.* (2002). In contrast to Iterative Feedback Tuning and Correlation-based Tuning, this method only needs one open or closed loop plant experiment to find a set of optimal control parameters. This methodology is applicable for linear, time-invariant, single input/single output, discrete time systems where the control is linear in the parameters. The main idea is to use a set of input/output data obtained from the true system, and a desired model for the loop. These will be used to form a set of input/output data which can be utilized to estimate the controller parameters in one step. The idea is based on the model reference criterion:

$$F_{MR}(\boldsymbol{\rho}) = \left\| \left(\frac{G(q^{-1})C(q^{-1},\boldsymbol{\rho})}{1+G(q^{-1})C(q^{-1},\boldsymbol{\rho})} - M(q^{-1}) \right) W(q^{-1}) \right\|_2^2 \tag{2.25}$$

where $M(q^{-1})$ is the reference model for the loop and $W(q^{-1})$ is user defined weighting function. The basic algorithm is a follows:

1. Select a reference model $M(q^{-1})$ with the desired closed loop dynamics.
2. Collect a sequence of input/output data in open loop from the true system.
3. Calculate the virtual reference signal $\bar{r}(t)$ such that $y(t) = M(q^{-1})\bar{r}(t)$.
4. Form the tracking error $e(t) = \bar{r}(t) - y(t)$.

5. Filter the error and the input signals through a suitable data filter $L(q^{-1})$. I.e. $e_L(t) = L(q^{-1})e(t)$ and $u_L(t) = L(q^{-1})u(t)$

6. Calculate the set of control parameters which minimize the following open loop identification criterion:

$$F_{VR}(\boldsymbol{\rho}) = \frac{1}{N}\sum_{t=1}^{N}(u_L(t) - C(q^{-1}, \boldsymbol{\rho})e_L(t))^2 \tag{2.26}$$

   which is a linear least squares problem with an analytical solution.

It can be shown that for a proper selection of the data filter $L(q^{-1})$, the criterion used in the algorithm approximates to the model reference criterion (Campi *et al.*, 2002; Lecchini *et al.*, 2002). As the algorithm is outlined here it is assumed that the data is noise free. In presence of noise a biased estimate is achieved. This can be circumvented by applying an instrumental variable method. A discussion on use of noisy data and closed loop experiments can be seen in Lecchini (2001). An extension to more general and nonlinear control structures is presented in Campi and Savareci (2006).

Virtual Reference Feedback Tuning is a very appealing direct tuning method due to the fact that only one experiment is required. Unfortunately it is suboptimal for restricted controller classes. As in the case for Correlation-based tuning this algorithm is well suited for optimizing the loop tracking performance, while optimal disturbance rejection may be difficult to handle due to lack of excitation in the input/output data when only noise perturb the system.

3

Iterative Feedback Tuning - IFT

The need for systematic algorithms for direct optimization of control loops with restricted complexity controllers have motivated the development of the Iterative Feedback Tuning method. The method is based on the ideas from optimal control, but addresses the problems of limited dynamic process information. The algorithm is easily applied in practice and can be consisted as an attractive alternative to classical tuning rules for e.g. PID control loop.

The basic idea is to formulate a performance cost function and use an experimental optimization algorithm to minimize this cost function, with respect to the control parameters for a process relevant control purpose. The optimization algorithm iteratively improves the performance based upon an estimation of the cost function sensitivity to the control parameters. Evaluation of the partial derivatives of the cost function with respect to the controller parameters, $\boldsymbol{\rho}$, is based on measurements taken form the closed loop system. This method of iterative performance enhancement does not include an estimate of the process model. The algorithm was presented first time in Hjalmarsson *et al.* (1994b) and have since been extended and tested in a number of papers i.e. Hjalmarsson *et al.* (1998); Lequin *et al.* (1999, 2003). See Hjalmarsson (2002) and Gevers (2002) for a more extensive overview of the development of the method and references to applications.

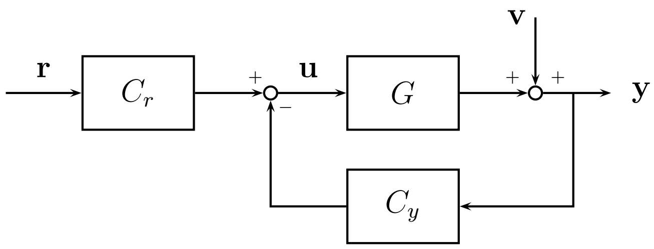


Figure 3.1. Feedback loop with a two degree of freedom controller.

The closed loop system depicted in Figure 3.1, implements a two degree of freedom controller, $\mathbf{C} = \{C_r, C_y\}$, on a discrete time, linear time-invariant and single input/single output system G. The system transfer functions are then given as:

$$\mathbf{y} = \frac{C_r G}{1 + C_y G}\mathbf{r} + \frac{1}{1 + C_y G}\mathbf{v} = T\mathbf{r} + S\mathbf{v} \tag{3.1a}$$

$$\mathbf{u} = \frac{C_r}{1 + C_y G}\mathbf{r} - \frac{C_y}{1 + C_y G}\mathbf{v} = SC_r\mathbf{r} - SC_y\mathbf{v} \tag{3.1b}$$

where $\mathbf{r}$ is the reference value for the measurements $\mathbf{y}$, $\mathbf{u}$ is the actuator variable and $\mathbf{v}$ is a noise signal for the system presented in deviation variables. S and T are the sensitivity and the complementary sensitivity functions respectively. Given a desired reference model for the closed loop T_d, the desired reference response from the loop is given as $\mathbf{y}_d = T_d\mathbf{r}$. The performance criterion can then be formulated as a typical quadratic cost function, $F(\boldsymbol{\rho})$, with penalty on $\tilde{\mathbf{y}} = \mathbf{y}(\boldsymbol{\rho}) - \mathbf{y}_d$ and the control effort. The optimal set of parameters will then require, that the partial derivative of the cost function with respect to the control parameters is zero, and this point represents the global minimum given the admissible parameter space. This optimal solution to the minimization problem can be obtained through an iterative gradient based search algorithm in case where the cost function is convex.

$$\begin{aligned} \boldsymbol{\rho}_{i+1} &= \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1} \frac{\partial F(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \\ &= \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1} \mathbf{J}(\boldsymbol{\rho}_i) \end{aligned} \tag{3.2}$$

where $\mathbf{R}_i$ is a positive definite matrix and $\mathbf{J}(\boldsymbol{\rho})$ is the gradient column vector of $F(\boldsymbol{\rho})$ with respect to the control parameters. The i'th step is then given by $\mathbf{h}_i = -\gamma_i \mathbf{R}_i^{-1}\mathbf{J}(\boldsymbol{\rho}_i)$. In case $\mathbf{R} = \mathbf{I}$ the algorithm steps in the steepest decent direction. In case $\mathbf{R} = \mathbf{H}(\boldsymbol{\rho}) = {}^{\partial^2 F(\boldsymbol{\rho})}/_{\partial \boldsymbol{\rho}^2}$ or an approximation to the Hessian, the Newton or Gauss-Newton algorithm appears. γ_i determines the step length and the choice of $\mathbf{R}$ and $\boldsymbol{\gamma}$ will thus affect the convergence properties of the method (Hjalmarsson *et al.*, 1994b; Nocedal and Wright, 1999).

The key contribution in Iterative Feedback Tuning is that it supplies an unbiased estimate of the cost function gradient, without estimating a plant model, $\hat{G}$, given that the noise $\mathbf{v}$ is a zero mean, weakly stationary random signal (Hjalmarsson *et al.*, 1998). Using an estimate of the $\boldsymbol{J}(\boldsymbol{\rho})$ in (3.2), instead of the analytical cost function gradient, as a stochastic approximation method will still make the algorithm converge to a local minimizer, provided that a minimizer exists, the estimate is unbiased and that the sequence of γ_i fulfills condition (3.3) (Robbins and Monro, 1951).

$$\sum_{i=1}^{\infty} \gamma_i^2 < \infty, \quad \sum_{i=1}^{\infty} \gamma_i = \infty \tag{3.3}$$

This condition is fulfilled by having $\gamma_i = {}^a/_i$ where a is some constant. This method has a convergence rate which is too slow for most industrial purposes (Mišković, 2006). In cases where the variance of the cost function gradient estimate approaches zero, due to a large number of data points, classical Gauss-Newton optimization with $\gamma_i = 1$, may be used instead in order to speed up the convergence.

Given the cost function

$$F(\boldsymbol{\rho}) = \frac{1}{2N} \mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=1}^{N} u_t(\boldsymbol{\rho})^2\right] \tag{3.4}$$

where the minimization criterion is

$$0 = \mathbf{J}(\boldsymbol{\rho}) = \frac{1}{N} \mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho}) \frac{\partial \tilde{y}_t}{\partial \boldsymbol{\rho}} + \lambda \sum_{t=1}^{N} u_t(\boldsymbol{\rho}) \frac{\partial u_t}{\partial \boldsymbol{\rho}}\right] \tag{3.5}$$

it is seen that estimates of $\partial\tilde{\mathbf{y}}/\partial\boldsymbol{\rho}$ and $\partial\mathbf{u}/\partial\boldsymbol{\rho}$ are needed in order to produce an estimate of $\boldsymbol{J}(\boldsymbol{\rho})$. Since $\boldsymbol{y}_d$ is not a function of the control parameters it holds that $\partial\tilde{\mathbf{y}}/\partial\boldsymbol{\rho} = \partial\mathbf{y}/\partial\boldsymbol{\rho}$. The partial derivatives of the in- and output with respect to the control parameters can be evaluated based on equation (3.1).

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_r}{\partial \boldsymbol{\rho}}T(\boldsymbol{\rho})\mathbf{r} - \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_y}{\partial \boldsymbol{\rho}}T(\boldsymbol{\rho})\mathbf{y} \tag{3.6a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \frac{\partial C_r}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})\mathbf{r} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})\mathbf{y} \tag{3.6b}$$

3.1 The Iterative Feedback Tuning Algorithm

Gradient estimates of the in- and output can be produced given data from three separate closed loop experiments on the system. The authors behind the Iterative Feedback Tuning method have published two different ways of conducting these experiments which will both be presented here in detail. In the paper Hjalmarsson *et al.* (1994b) where the Iterative Feedback Tuning framework was initially presented, a method was used that in this thesis will be referred to as the original Iterative Feedback Tuning formulation. In the paper Hjalmarsson *et al.* (1998) and later publications a slightly refined version was used which will be referred to as the refined Iterative Feedback Tuning formulation.

3.1.1 The Original Formulation of Iterative Feedback Tuning

The three experiments in the original formulation are designed as follows:

1) $\mathbf{r}^1 = \mathbf{r}$ i.e. the reference signal sequence in the first experiment is the same as for normal operation of the process.

2) $\mathbf{r}^2 = \mathbf{y}^1$ i.e. the reference signal sequence in the second experiment is the output from the first experiment

3) $\mathbf{r}^3 = \mathbf{r}$ i.e. the reference signal sequence in the third experiment is the same as for normal operation of the process, just as in the first experiment.

These experiments give the following in- and outputs

$$\begin{array}{lll} \text{Ex. no 1:} & \mathbf{y}^1 = T(\boldsymbol{\rho})\mathbf{r} + S(\boldsymbol{\rho})\mathbf{v}^1 & \mathbf{u}^1 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})\mathbf{r} - C_y(\boldsymbol{\rho})\mathbf{v}^1\right) \\ \text{Ex. no 2:} & \mathbf{y}^2 = T(\boldsymbol{\rho})\mathbf{y}^1 + S(\boldsymbol{\rho})\mathbf{v}^2 & \mathbf{u}^2 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})\mathbf{y}^1 - C_y(\boldsymbol{\rho})\mathbf{v}^2\right) \\ \text{Ex. no 3:} & \mathbf{y}^3 = T(\boldsymbol{\rho})\mathbf{r} + S(\boldsymbol{\rho})\mathbf{v}^3 & \mathbf{u}^3 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})\mathbf{r} - C_y(\boldsymbol{\rho})\mathbf{v}^3\right) \end{array}$$

The sequence of input/output data form these experiments $(\mathbf{y}^i;\mathbf{u}^i)$ $i\in\{1,2,3\}$ will be utilized as follows:

$$\tilde{\mathbf{y}} = \mathbf{y}^1 - \mathbf{y}^d \tag{3.7a}$$

$$\mathbf{u} = \mathbf{u}^1 \tag{3.7b}$$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{y}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{y}^2\right) \tag{3.7c}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{u}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{u}^2\right) \tag{3.7d}$$

It can be seen from (3.7a) and (3.7b) that the first experiment gives the measurement of $\tilde{\mathbf{y}}$ and $\mathbf{u}$ that is needed in the estimate of $\mathbf{J}(\boldsymbol{\rho})$. The estimate of the gradients of the input and output can be written as

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right) \tag{3.8}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right) \tag{3.9}$$

which means that the two terms in the estimate of the cost function gradient in (3.5) become:

$$\begin{aligned}\tilde{\mathbf{y}}\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} &= (T(\boldsymbol{\rho})\mathbf{r} - \mathbf{y}^d)\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + S(\boldsymbol{\rho})\mathbf{v}^1\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \\ &\frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left(T(\boldsymbol{\rho})\mathbf{r} - \mathbf{y}^d\right)\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] + \\ &\frac{S^2(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\mathbf{v}^1\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right]\end{aligned} \tag{3.10}$$

$$\begin{aligned}\mathbf{u}\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} &= S(\boldsymbol{\rho})C_r(\boldsymbol{\rho})\mathbf{r}\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} + S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})\mathbf{v}^1\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \\ &S^2(\boldsymbol{\rho})C_y(\boldsymbol{\rho})\mathbf{r}\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] - \\ &\frac{S^2(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\mathbf{v}^1\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right]\end{aligned} \tag{3.11}$$

From this result it can be seen that the noise signal $\mathbf{v}^1$ plays an active part in the optimization of the control parameters. This is due to the squared terms of $\mathbf{v}^1$ which is produced by $\tilde{\mathbf{y}}{\partial \mathbf{y}}/{\partial \boldsymbol{\rho}}$. The noise terms from the second and third experiment act as nuisance. Further more it is evident that having a zero mean, weakly stationary random noise sequence $\mathbf{v}$ acting on the process will ensure that the estimate of the cost function gradient is unbiased. The variance of this estimate is proportional to two times the variance of the noise signals.

From equation (3.6) it can be seen that if the controller is tuned for disturbance rejection hence $\mathbf{r} = \mathbf{0}$, it is only required to perform the first and second experiment. This is because the term in (3.6) where data from the third experiment are used, is zero and inclusion of this last experiment will only introduce the undesired noise sequence $\mathbf{v}^3$ in the estimate for the Jacobian.

3.1.2 The Refined Formulation of Iterative Feedback Tuning

This second Iterative Feedback Tuning formulation is based on a small change in the equations for the derivatives of $\mathbf{y}$ and $\mathbf{u}$ in equation (3.6) by adding and subtraction the same term.

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) T(\boldsymbol{\rho})\mathbf{r} + \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})(\mathbf{r} - \mathbf{y}) \tag{3.12a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) S(\boldsymbol{\rho})\mathbf{r} + \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})(\mathbf{r} - \mathbf{y}) \tag{3.12b}$$

This change motivates that the second experiments should be changed, and for the refined IFT formulation the experiments are:

1) $\mathbf{r}^1 = \mathbf{r}$ i.e. the reference signal sequence in the first experiment is the same as for normal operation of the process.

2) $\mathbf{r}^2 = \mathbf{r} - \mathbf{y}^1$ i.e. the reference signal sequence in the second experiment is the difference between the ordinary reference and the output from the first experiment

3) $\mathbf{r}^3 = \mathbf{r}$ i.e. the reference signal sequence in the third experiment is the same as for normal operation of the process, just as in the first experiment.

These experiments give the following in- and outputs:

Ex. no 1:	$\mathbf{y}^1 = T(\boldsymbol{\rho})\mathbf{r} + S(\boldsymbol{\rho})\mathbf{v}^1$	$\mathbf{u}^1 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})\mathbf{r} - C_y(\boldsymbol{\rho})\mathbf{v}^1\right)$
Ex. no 2:	$\mathbf{y}^2 = T(\boldsymbol{\rho})(\mathbf{r} - \mathbf{y}^1) + S(\boldsymbol{\rho})\mathbf{v}^2$	$\mathbf{u}^2 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})(\mathbf{r} - \mathbf{y}^1) - C_y(\boldsymbol{\rho})\mathbf{v}^2\right)$
Ex. no 3:	$\mathbf{y}^3 = T(\boldsymbol{\rho})\mathbf{r} + S(\boldsymbol{\rho})\mathbf{v}^3$	$\mathbf{u}^3 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})\mathbf{r} - C_y(\boldsymbol{\rho})\mathbf{v}^3\right)$

The sequence of input/output data from these experiments $(\mathbf{y}^i; \mathbf{u}^i)$ $i \in \{1, 2, 3\}$ will be utilized as

$$\tilde{\mathbf{y}} = \mathbf{y}^1 - \mathbf{y}^d \tag{3.13a}$$

$$\mathbf{u} = \mathbf{u}^1 \tag{3.13b}$$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{y}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{y}^2\right] \tag{3.13c}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{u}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{u}^2\right] \tag{3.13d}$$

As in the original formulation the first experiment gives the measurement of $\tilde{\mathbf{y}}$ and $\mathbf{u}$ that is needed in the estimate of $\mathbf{J}(\boldsymbol{\rho})$. The estimate of the gradients of the input and the output can be written as

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \tag{3.14}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \tag{3.15}$$

which means that the two terms in the estimate of the cost function gradient become

$$
\begin{aligned}
\tilde{\mathbf{y}}\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} &= (T(\boldsymbol{\rho})\mathbf{r} - \mathbf{y}^d)\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + S(\boldsymbol{\rho})\mathbf{v}^1\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \\
&\quad \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left(T(\boldsymbol{\rho})\mathbf{r} - \mathbf{y}^d\right)\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] + \\
&\quad \frac{S^2(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\mathbf{v}^1\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \qquad (3.16) \\
\mathbf{u}\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} &= S(\boldsymbol{\rho})C_r(\boldsymbol{\rho})\mathbf{r}\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} + S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})\mathbf{v}^1\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \\
&\quad S^2(\boldsymbol{\rho})C_y(\boldsymbol{\rho})\mathbf{r}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] - \\
&\quad \frac{S^2(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\mathbf{v}^1\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \qquad (3.17)
\end{aligned}
$$

From this result, equation (3.14) and (3.15) it can be seen that $\mathbf{v}^1$ plays the same role as in the optimization of the control parameters in the original Iterative Feedback Tuning formulation. The properties of $\mathbf{v}$ being a zero mean, weakly stationary random noise sequence acting on the process will still ensure that the estimate of $\boldsymbol{J}(\boldsymbol{\rho})$ is unbiased, but the variance of this estimate is likely to be smaller than in the original formulation.

As in the case with the original formulation, controller tuning for disturbance rejection with $\mathbf{r} = \mathbf{0}$ only requires the first and second experiment. Furthermore, the formulation of the derivatives in equation (3.12) renders the third experiment unnecessary if $C_r = C_y$ since the bracket in the first term becomes zero.

3.1.3 General Properties and Shortcomings of the Iterative Feedback Tuning Algorithm

Both Iterative Feedback Tuning formulations are applicable to all types of controllers when the data filters used in the estimators (3.7c) and (3.7d) or (3.13c) and (3.13d) can be formulated as proper, stable transfer functions. The filters needed to construct the gradient estimate of the in- and outputs for a typical implementation of a two degree of freedom PID controller with derivative filter are listed in Appendix A. The calculations in the algorithm are not restricted by the process as long as the system is closed loop stable. Appropriate choices of $\boldsymbol{\gamma}$ and $\mathbf{R}$ are needed since these affect the direction and the step size in the iterative improvement of the control parameters. The process and the controller obviously affect the signals $\{\mathbf{u}, \mathbf{y}\}$ which together with λ influence the shape of the cost function.

3.1.3.1 Robustness and stability

Since no model of the system is estimated during the Iterative Feedback Tuning, nominal or robust stability of the loop can not be evaluated prior to the implementation of the new controller. Problems may arise as a consequence of the design or through the optimization itself. Choosing a performance criterion where the stability margin for the optimal controller is too small will cause the algorithm to converge towards a non-robust controller. E.g. the disturbance rejection problem for a single input/single output system should not have a performance cost function without penalty on the input, since a minimum variance controller is inherently non-robust Hjalmarsson (2002).

Robust control theory operates with the following sensitivity matrix

$$\mathbf{T}(G, C_y(\boldsymbol{\rho})) = \begin{bmatrix} GS(\boldsymbol{\rho})C_y(\boldsymbol{\rho}) & GS(\boldsymbol{\rho}) \\ S(\boldsymbol{\rho})C_y(\boldsymbol{\rho}) & S(\boldsymbol{\rho}) \end{bmatrix} \tag{3.18}$$

In case the closed loop system is stable the generalized stability margin is

$$b_{G,C_y} = \|\mathbf{T}(G, C_y(\boldsymbol{\rho}))\|_\infty^{-1} \tag{3.19}$$

In Vinnicombe (1993) it is shown that C_2 will stabilize the loop if C_1 stabilizes $\mathbf{T}(G, C_1)$ and if $\delta_v(C_1, C_2) < b_{G,C_1}$. The Vinnicombe distance, δ_v is defined as

$$\delta_v = \|(1 + C_2 C_2^*)^{-1/2}(C_2 - C_1)(1 + C_1 C_1^*)^{-1/2}\|_\infty \tag{3.20}$$

provided that winding number condition (3.21) is satisfied or $\delta_v = 1$ otherwise[1]. De Bruyne *et al.* (1999) use this result to adjust the step length, γ_i, in order to ensure robust parameter update in the iterations of the Iterative Feedback Tuning algorithm.

$$\delta_v < 1 \Leftrightarrow \begin{cases} \|1 + C_2^* C_1\| \neq 0 \, \forall \omega, & \text{and} \\ \text{wno}(1 + C_2^* C_1) + \eta(C_1) - \eta(C_2) = 0 \end{cases} \tag{3.21}$$

where wno$(\cdot)$ is number of counterclockwise encirclements of the origin in the Nyquist plot, and $\eta(\cdot)$ is the number of unstable poles.

Extension of the Iterative Feedback Tuning algorithms in order to insure both performance enhancement and a robust control design have been proposed. The idea is to have a performance criterion which includes both the normal performance criterion plus a term that promotes robustness. In Veres and Hjalmarsson (2002) an approximation of the infinity norm of the sensitivity matrix provides robustness where Procházka *et al.* (2005) uses a weighted two norm in order to tune the stability for a critical frequency band.

3.1.3.2 Tuning a loop with a nonlinear process.

A nonlinear process may produce a more complex cost function which limits the Iterative Feedback Tuning algorithm's ability to converge to the global minimum, but it will not prevent the calculations contained in each iteration. For a non-convex optimization it is possible to end up in a local minimum. Having an initial controller which is far from the optimal, it can be beneficial to start with a reference trajectory which is closer to the initial, than the desired, and then gradually tighten the requirement through the iterations in order to avoid local minima. For optimization of the settling time an criterion that relax the penalty during the first part of the time horizon can be used in a similar fashion. This type of criterion is elaborated

[1] C_i^* is the complex conjugated transposed of C_i

in section 3.1.5. The Iterative Feedback Tuning algorithm can in the standard formulations cope with some types of nonlinear systems in closed loop with a linear controller (Hjalmarsson, 2002). This is due to the fact that the Iterative Feedback Tuning algorithm generates the true gradient of the in- and outputs up to first order. If the difference between experiments under normal conditions and the special experiment in the gradient formation is small, this approximation will be sufficiently good. It must be noted that a nonlinear system can be more sensitive to alternations in disturbance signal or changes in operational regions, compared to the conditions under which the controller where tuned. Sjöberg *et al.* (2003) suggest a modified set of experiments for gradient generation which are developed for tuning of nonlinear controllers and possible nonlinear processes. The number of experiments necessary for forming the gradient is in this algorithm $n+1$ where n is the number of parameters in the controller that is subject to tuning, which renders the method useful for optimizations of controllers with only a small number of parameters.

3.1.3.3 Tuning a multivariable loop.

In case the system under control has multiple inputs and outputs, Iterative Feedback Tuning can still be applied (Hjalmarsson, 1999). An example of a case study with tuning of a decoupling controller for a (2×2) system is presented in Gunnarsson *et al.* (2003). Given a description of a general, discrete time, linear, time-invariant, multiple input/multiple output system, which is depicted in figure 3.2(a), the model can be given as:

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{w} \end{pmatrix} = \mathbf{G} \begin{pmatrix} \mathbf{r} \\ \mathbf{v} \\ \mathbf{u} \end{pmatrix} \tag{3.22a}$$

with the control law

$$\mathbf{w} = \mathbf{C}(\boldsymbol{\rho})\mathbf{u} \tag{3.22b}$$

where $\mathbf{r} \in \mathbb{R}^{n_r}$ is a vector of external signals as set points or reference trajectories, $\mathbf{v} \in \mathbb{R}^{n_v}$ is unmeasurable process inputs as disturbances and noise, $\mathbf{w} \in \mathbb{R}^{n_w}$ is the all the measured outputs, $\mathbf{u} \in \mathbb{R}^{n_u}$ is all the control signals and $\mathbf{y} \in \mathbb{R}^{n_y}$ represents all variables which are included in the performance criterion, which then can be given as:

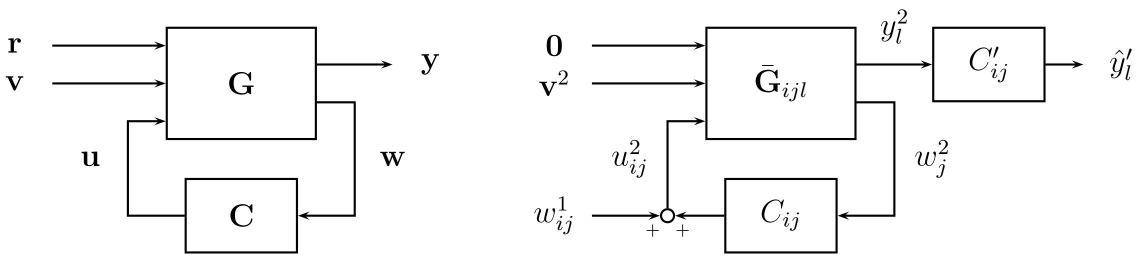
$$F(\boldsymbol{\rho}) = \frac{1}{2N} E\left[\sum_{t=1}^{N} \mathbf{y}(t,\boldsymbol{\rho})^T \mathbf{W}_t \mathbf{y}(t,\boldsymbol{\rho})\right] \tag{3.23}$$

This structure of the cost function will allow both penalty on deviation from set points and control signals. Assuming all the $n_w \times n_u$ blocks in the controller which produce the pairing between controlled signal and manipulated variables are independently parametized, then the relation between controller block C_{ij} and the scalar output y_l is

$$\begin{pmatrix} y_l \\ w_{ij} \end{pmatrix} = \bar{\mathbf{G}}_{ijl} \begin{pmatrix} \mathbf{r} \\ \mathbf{v} \\ u_{ij} \end{pmatrix} \tag{3.24a}$$

$$u_{ij} = C_{ij} w_{ij} \tag{3.24b}$$

where $\bar{\mathbf{G}}_{ijl}$ is the interconnection between the real plant and the remaining blocks in the controller. This system has the structure of a single input/single output system, and the gradients of the cost function can thus be evaluated by the same type of experiments as previously described. Only $1 + n_u \times n_w$ experiments are required in order to calculate an estimate of the gradients: First one normal experiment with the desired reference signals, $\mathbf{r}^1 = \mathbf{r}$, where the signals $\mathbf{w}^1$ and $\mathbf{y}^1$ are measured. This experiment corresponds to the block diagram in Figure 3.2(a). Then one experiment for each block, C_{ij} in the controller. In these special experiments, which are used to form the gradients, $\mathbf{r} = \mathbf{0}$ and the output w_{ij}^1 is added to the control signal u_{ij}. The measured output y_l^2 is then filtered through the gradient of the controller C_{ij} with respect to one of the parameters in $\boldsymbol{\rho}$, in order to form the gradient of the output y_l with respect to that particular parameter. Figure 3.2(b) shows how the gradients are produced by the second experiment. The number of experiments can be quiet large for a multivariable system. A number of alternative approaches to estimation of ${}^{\partial \mathbf{y}}/_{\partial \boldsymbol{\rho}}$, which is not unbiased but still useful, is presented in Jansson and Hjalmarsson (2004).



(a) Multivariable feedback loop (b) Gradient experiment

Figure 3.2. Figure (a) shows a block diagram for a general multivariable feedback loop. Figure (b) shows a block diagram of how the individual gradients can be formed by conducting a set of special experiments on the closed loop system.

3.1.4 Estimation of the Hessian

In order to take a Newton step in (3.2), it is necessary to know the Hessian of the cost function with respect to the control parameters. Since the process model is unknown, it is only possible to evaluate an estimate of the Hessian based on input/output measurements. Hjalmarsson *et al.* (1994b) suggest the approximation in (3.25) of the Hessian which can be calculated based on the available signals. This approximation is the second derivative of the cost function (3.4) where squared derivative terms have been disregarded. An advantage of using this approximation is that it will produce a positive definite matrix. This estimate is, unlike the estimate of the cost function gradient, biased. A Gauss-Newton iteration based on this approximation has proven to be superior to a pure gradient iteration using $\mathbf{R} = \mathbf{I}$ (Hjalmarsson *et al.*, 1994b).

$$\mathbf{R} = \frac{1}{N}\sum_{t=1}^{N}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\right)^T + \lambda\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\right)^T\right) \tag{3.25}$$

The estimate in (3.25) is dependent on which of the two formulations of the iterative feedback tuning that has been used to generate the gradients of the input and output. In case the original formulation has been used the gradients are given by (3.8) and (3.9) and the multiplications in (3.25) become:

$$\begin{aligned}
\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}}\right)^T =& \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\left(\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\right)^T + \\
& \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\left(\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^3 - \left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^2\right) + \left(\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\right)^T\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right)\right] + \\
& \frac{S^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})}\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\right)^T (\mathbf{v}^3)^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T (\mathbf{v}^2)^2\right] - \\
& \frac{S^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})}\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^3\mathbf{v}^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^2\mathbf{v}^3\right]
\end{aligned} \tag{3.26}$$

$$\begin{aligned}
\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}}\right)^T =& \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}\left(\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}\right)^T - \\
& \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}\left(\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^3 - \left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^2\right) + \left(\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}\right)^T\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\mathbf{v}^3 - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right)\right] + \\
& \frac{S^2(\boldsymbol{\rho})C_y^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})}\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\right)^T (\mathbf{v}^3)^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T (\mathbf{v}^2)^2\right] - \\
& \frac{S^2(\boldsymbol{\rho})C_y^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})}\left[\frac{\partial C_r}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^3\mathbf{v}^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^2\mathbf{v}^3\right]
\end{aligned} \tag{3.27}$$

If the refined formulation is used such that (3.14) and (3.15) describe the gradients of the input and output, the multiplications in (3.25) will be:

$$\begin{aligned}
\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}}\right)^T =& \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\left(\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\right)^T + \\
& \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\left(\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^3 + \left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^2\right) + \\
& \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left(\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}\right)^T\left(\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right) + \\
& \frac{S^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T (\mathbf{v}^3)^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T (\mathbf{v}^2)^2\right] - \\
& \frac{S^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\left(\frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^3\mathbf{v}^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)^T \mathbf{v}^2\mathbf{v}^3\right]
\end{aligned} \tag{3.28}$$

$$\begin{aligned}
\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} \left(\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}}\right)^T =& \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} \left(\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}\right)^T - \\
& \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})} \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} \left(\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right)^T \mathbf{v}^3 + \left(\frac{\partial C_y}{\partial \boldsymbol{\rho}} \right)^T \mathbf{v}^2 \right) - \\
& \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})} \left(\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} \right)^T \left(\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) \mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}} \mathbf{v}^2 \right) + \\
& \frac{S^2(\boldsymbol{\rho})C_y^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right)^T (\mathbf{v}^3)^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}} \left(\frac{\partial C_y}{\partial \boldsymbol{\rho}} \right)^T (\mathbf{v}^2)^2 \right] - \\
& \frac{S^2(\boldsymbol{\rho})C_y^2(\boldsymbol{\rho})}{C_r^2(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) \left(\frac{\partial C_y}{\partial \boldsymbol{\rho}} \right)^T \mathbf{v}^3\mathbf{v}^2 + \frac{\partial C_y}{\partial \boldsymbol{\rho}} \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right)^T \mathbf{v}^2\mathbf{v}^3 \right]
\end{aligned} \tag{3.29}$$

It can be seen that the bias in (3.25) is due to the squared terms of $\mathbf{v}^2$ and $\mathbf{v}^3$ which will give a positive contribution, where all other terms which involve these two noise signals will have an expected value of zero, if the noise is zero mean. The effect of the bias depends on how dominant these bias terms are compared to the exact realization of $\partial \mathbf{y}/\partial \boldsymbol{\rho} \, (\partial \mathbf{y}/\partial \boldsymbol{\rho})^T$ and $\partial \mathbf{u}/\partial \boldsymbol{\rho} \, (\partial \mathbf{u}/\partial \boldsymbol{\rho})^T$. The positive bias term will reduce the step size in (3.2).

In Solari and Gevers (2004) it is shown that an unbiased estimate of the Hessian can be obtained by increasing the number of experiments. As shown above, it is the squared noise signals in the estimate of the gradients that causes problems. The solution is to have an experiment no 4 and 5 which are conducted as experiments no 2 and 3 of either of the two formulations of the Iterative Feedback Tuning. In this way a second estimate of the gradients $\widehat{\partial \mathbf{y}/\partial \boldsymbol{\rho}}$ and $\widehat{\partial \mathbf{u}/\partial \boldsymbol{\rho}}$ can be formed based on the two last experiments. This will produce two sets of gradients which have different and uncorrelated noise content. The Hessian can then be formed without bias by multiplication of these two sets of independent gradient estimates. Solari and Gevers (2004) further show how the second order derivatives can be estimated from further close loop experiments on the system, but this approach is limited to the case where the controller is tuned for disturbance rejection only.

3.1.5 Alternative Optimization Criteria

By introducing an alternative formulation of the cost function, F, the Iterative Feedback Tuning method can be tailored to fit a specific optimization criterion. The demand that needs to be imposed on the formulation of the cost functions, is that it has to be differentiable with respect to the controller parameters. Further more, it is required that a consistent estimate of the cost function gradient can be constructed based on the designed experiments.

A frequency weighting can be included in the cost function in order to handle non-minimum phase or unstable controllers (Hjalmarsson *et al.*, 1998).

$$F(\boldsymbol{\rho}) = \frac{1}{2N} \mathrm{E} \left[\sum_{t=1}^{N} (L_y \tilde{y}_t(\boldsymbol{\rho}))^2 + \lambda \sum_{t=1}^{N} (L_u u_t(\boldsymbol{\rho}))^2 \right] \tag{3.30}$$

where L_y and L_u are filters. Lequin (1997) suggests a modification of the cost function where non-negative time dependent weights are multiplied on both the squared deviation between the desired and achieved output and on the squared control action.

$$F(\boldsymbol{\rho}) = \frac{1}{2N}\mathrm{E}\left[\sum_{t=1}^{N} w_t^y \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=1}^{N} w_t^u u_t(\boldsymbol{\rho})^2\right] \tag{3.31}$$

Letting these weights be equal zero for the $t \in \{1, t_0 - 1\}$ and one for $t \in \{t_0, N\}$ gives the following cost function where the time up till t_0 are referred to as the mask.

$$F(\boldsymbol{\rho}) = \frac{1}{2N}\mathrm{E}\left[\sum_{t=t_0}^{N} \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=t_0}^{N} u_t(\boldsymbol{\rho})^2\right] \tag{3.32}$$

This formulation of the cost function can be convenient when optimizing the settling time (Lequin *et al.*, 1999).

In case it is desired to penalize the control velocity and not the position, the cost function can be formulated using the control move.

$$\begin{aligned} F(\boldsymbol{\rho}) &= \frac{1}{2N}\mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=1}^{N} \Delta u_t(\boldsymbol{\rho})^2\right] \qquad (3.33) \\ &= \frac{1}{2N}\mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \left(\sum_{t=1}^{N} u_t(\boldsymbol{\rho})^2 + \sum_{t=1}^{N} u_{t-1}(\boldsymbol{\rho})^2 - 2\sum_{t=1}^{N} u_t(\boldsymbol{\rho}) u_{t-1}(\boldsymbol{\rho})\right)\right] \end{aligned}$$

The partial derivatives are then given by

$$\begin{aligned} \mathbf{J}(\boldsymbol{\rho}) =& \frac{1}{N}\mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho}) \frac{\partial \tilde{y}_t}{\partial \boldsymbol{\rho}} + \right. \\ & \left. \lambda \sum_{t=1}^{N} \left(u_t(\boldsymbol{\rho}) \frac{\partial u_t}{\partial \boldsymbol{\rho}} + u_{t-1}(\boldsymbol{\rho}) \frac{\partial u_{t-1}}{\partial \boldsymbol{\rho}} - u_t(\boldsymbol{\rho}) \frac{\partial u_{t-1}}{\partial \boldsymbol{\rho}} - u_{t-1}(\boldsymbol{\rho}) \frac{\partial u_t}{\partial \boldsymbol{\rho}}\right)\right] \\ =& \frac{1}{N}\mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho}) \frac{\partial \tilde{y}_t}{\partial \boldsymbol{\rho}} + \lambda \sum_{t=1}^{N} \left(u_t(\boldsymbol{\rho}) - u_{t-1}(\boldsymbol{\rho})\right) \left(\frac{\partial u_t}{\partial \boldsymbol{\rho}} - \frac{\partial u_{t-1}}{\partial \boldsymbol{\rho}}\right)\right] \qquad (3.34) \end{aligned}$$

It is seen from (3.34) that substituting the control value by the control move does not change the method significantly. The same signals are needed for the estimation of the cost function gradient, only the index is shifted. This index shift does not cause problems since both u_0 and y_0 are known.

3.2 Contributions to the Iterative Feedback Tuning Method

In the following chapters, novel contributions to the Iterative Feedback Tuning method will be presented. This chapter have indicated that Iterative Feedback

Tuning is a matured methodology, and that several researchers have contributed to analyze the properties of the method and extending both the basis and areas of application, e.g. robustness analysis and application for multivariable system. The results presented in the following falls into two separate categories, which therefore have been separated into two different parts.

The first contribution does not change the tuning method algorithm, but investigate the application on two control structures where application of the Iterative Feedback Tuning is novel. The first of these control structures is inventory control. In the control law for the inventory controller, the process model is imbedded, hence a nonlinear controller is possible. In order to correct for modeling errors the control law includes a term with a classical control action e.g. proportional and integral actions. The tuning of the proportional gain and the integral time in a inventory controller can not be performed based on classical tuning rules, due to the overall structure of the control law. Iterative Feedback Tuning of these parameters has been investigated and successfully performed for a multivariable level controller on a pilot plant. These results are presented in chapter 4. The second control structure where the application of Iterative Feedback tuning has been investigated is for tuning the feedback gain and the observer gain in a state space system description of a feedback loop. The optimal value of these two gains has an analytical solution in case the system model and the noise properties are known.

The second contribution is related to the general problem of tuning a controller for disturbance rejection. Iterative Feedback Tuning is in principle able to handle the disturbance rejection problem, since the noise present in the first experiment in an iteration drives the optimization. Unfortunately the tuning method may exhibit a very slow rate of convergence for this type of problems, since the variance of the gradient estimates of the in- and outputs may be large compared to their mean values. Solutions for speeding up the convergences, and hence prevent a very large number of plant experiments, have been focused on the application of pre-filters in the performance cost function. This thesis presents a solution based on the application of external perturbation signals in order to introduce more information content in data. This contribution is reported in chapter 6 and 7. In the latter chapter, a more elaborate analysis of the convergence properties of the Iterative Feedback Tuning method for the disturbance rejection problem is given. This includes both the nominal case and the tuning with external perturbations. An algorithm for design of the optimal spectral properties for the perturbation signal is part of this contribution.

Part II

Iterative Feedback Tuning for More Complex Control Structures

4

Data Driven Tuning of Inventory Controllers

Abstract

A systematic method for criterion based tuning of inventory controllers based on data-driven Iterative Feedback Tuning is presented. This tuning method circumvent problems with modeling bias. The process model used for the design of the inventory control is utilized in the tuning as an approximation to reduce time required on experiments. The method is illustrated in an application with a multivariable inventory control implementation on a four tank system.

4.1 Introduction

The purpose of this paper is to present a systematic method for tuning inventory controllers with Iterative Feedback Tuning. This data-driven tuning approach optimizes the actual closed loop performance hence circumventing problems due to modelling bias, that is part of any model for a real system, which would affect the control design. Furthermore, the process model will be utilized in the tuning algorithm in order to decrease time for plant experiments.

Inventory process control is based on passivity theory which states that the dynamical behavior of a system can be classified in terms of the conservation, dissipation and transport of positive extensive thermodynamic properties of the system. In passive systems, the stored amount of this property in any given time interval, is always lower or at most equal to the amount delivered to the system during the same time (Sira and Angulo, 1997). The theory is closely connected to optimization of just in time production of supply chains. In the work by Ydstie and coworkers, passivity theory was first applied on process systems and a formal connection was established between the macroscopic thermodynamics of process systems and passivity theory of nonlinear control (Ydstie and Alonso, 1997). In continuation Farschman *et al.* (1998) utilized the structure of first principle models in formulation of a nonlinear control law which has the form of output feedback linearization for which Byrnes *et al.* (1991) has proven closed loop stability through fulfillment of the passivity inequality for minimum phase systems and certain classes of nonlinear minimum phase systems. Inventory control has proven a useful methodology to synthesize a complex control law with a simple transfer function in the feedback and have been tested for a number of applications (Dueñas, 2004; Farschman *et al.*, 1998).

The problem of tuning the parameters in the feedback loop in the inventory control law is an area which has not received much attention. Dueñas (2004) states that classic tuning rules for linear systems can be applied in case where a perfect model of the system is available and all inventories are used for control, in which case perfect feedback linearization is achieved. Tuning of systems where a biased process model has been used in the design of the inventory controller will be addressed in this paper. The approach which will be presented uses the process model in the control design but a data driven method for tuning performance of the closed loop in order to compensate for modelling errors. Iterative Feedback Tuning, presented in Hjalmarsson *et al.* (1994b) for linear SISO systems, is an applicable methodology which have since been matured and developed (Hjalmarsson, 2002) and tested in a number of papers (Hjalmarsson *et al.*, 1998; Lequin *et al.*, 1999, 2003).

This paper is organized with a short introduction on the formulation of the control law for an inventory controller for a SISO system in the following section. A SISO formulation is used to ease notation but the remaining part of paper will focus on MIMO formulation due to the nature of the case study. Section 4.3 contains a formulation and problem statement for criterion based controller tuning which is followed by section 4.4 explaining Iterative Feedback Tuning. A case study on tuning a multivariable inventory controller implemented on a pilot scale of the quadruple tank process as given in section 4.5 before the concluding remarks.

4.2 Inventory Control

An inventory, υ, is represented by a physical extensive property and its general balance is given by

$$\begin{pmatrix} Accumul. \\ of \quad \upsilon \end{pmatrix} = \underbrace{\begin{pmatrix} Input \quad flow \\ of \quad \upsilon \end{pmatrix} - \begin{pmatrix} Output \quad flow \\ of \quad \upsilon \end{pmatrix}}_{\phi(d,x,u)} + \underbrace{\begin{pmatrix} Generation \\ of \quad \upsilon \end{pmatrix} - \begin{pmatrix} Consump. \\ of \quad \upsilon \end{pmatrix}}_{p(x)} \tag{4.1}$$

where a distinction is made between $\phi(d, x, u)$ and $p(x)$ which represent transport to the system and production in the system respectively. x, u and d is the state, the input and the disturbances for the associated general nonlinear dynamical system

$$\dot{x} = f(x) + g(d, x, u) \qquad x(0) = x_0 \tag{4.2a}$$
$$y = h(x) \tag{4.2b}$$

The function $f(\cdot)$ describes the internal state evolution due to generation or consumption, the function $g(\cdot)$ describes the external contribution to the state evolution and $h(\cdot)$ maps the state to the output. Let υ be an arbitrary inventory associated with the dynamic system (4.2), then the dynamic behavior is given by

$$\frac{d\upsilon(x)}{dt} = \underbrace{\frac{d\upsilon(x)}{dx} f(x)}_{L_f\upsilon} + \underbrace{\frac{d\upsilon(x)}{dx} g(d, x, u)}_{L_g\upsilon} \tag{4.3}$$

Where the terms $L_f\upsilon$ and $L_g\upsilon$ are directional derivatives. Equation (4.1) shows how ϕ and p are represented in the conservation law balance. Consequently, inventory systems can be written in the same form as the dynamic system (4.2).

$$\frac{d\upsilon}{dt} = p(x) + \phi(d, x, u) \tag{4.4a}$$
$$\upsilon = w(x) \tag{4.4b}$$

which is the notation used for inventory control (Farschman *et al.*, 1998). The term p denotes the production rate of the inventory and let p^* represent a stationary value of the production rate. The term ϕ is the supply rate for the system. The connection between these and the directional derivatives are given as

$$p(x) = L_f\upsilon(x) + p^*, \qquad p^* = p(0) \tag{4.5}$$
$$\phi(d, x, u) = L_g\upsilon(x) - p^* \tag{4.6}$$

The inventory controller with proportional action on the feedback $e(t) = (\upsilon(t) - \upsilon^{set}(t))$ or with on-off control is given by control laws (4.7) and (4.8) respectively.

$$\phi(d, x, u) + p(x) = -K_c e(t) \tag{4.7}$$

$$\phi(d, x, u) + p(x) = \begin{cases} \delta & if \quad e(t) < -\epsilon \\ 0 & if \quad -\epsilon \le e(t) \le \epsilon \\ -\delta & if \quad e(t) > \epsilon \end{cases} \tag{4.8}$$

In case a perfect model has been used for the inventory, a proportional controller will be sufficient and efficient in rejecting disturbances and tracking set points given a proper value of the proportional gain. In case the model is biased, which is the case for all real model based control implementations, it may be necessary to include integral action in the feedback control. This formulation is given in (4.9). Likewise, derivative action could be a part of the feedback, but given a reasonable process model the feed forward part of the inventory control renders such action unnecessary.

$$\phi(d,x,u)+p(x) = -K_c\left(e(t)+\frac{1}{\tau_I}\int_0^t e(\tau)d\tau\right) \tag{4.9}$$

It is seen that the inventory formulation can yield a complex and nonlinear controller depending on the model for the inventory. The problem of tuning the inventory controller is then to select proper parameters for the feedback part of the controller, which will provide sufficient closed loop performance.

4.3 Criterion Based Controller Tuning

Given a description of a closed loop system where the controller, $\mathbf{C}(\boldsymbol{\rho})$ is acting on the multivariable discrete linear time invariant system $\mathbf{G}$, the transfer functions are given as:

$$\begin{aligned}
\mathbf{y}(\boldsymbol{\rho}) &= (\mathbf{1}+\mathbf{C}(\boldsymbol{\rho})\mathbf{G})^{-1}\mathbf{C}(\boldsymbol{\rho})\mathbf{G}\mathbf{r}+(\mathbf{1}+\mathbf{C}(\boldsymbol{\rho})\mathbf{G})^{-1}\mathbf{v} \\
&= \mathbf{T}(\boldsymbol{\rho})\mathbf{r}+\mathbf{S}(\boldsymbol{\rho})\mathbf{v} \qquad (4.10\text{a}) \\
\mathbf{u}(\boldsymbol{\rho}) &= (\mathbf{1}+\mathbf{C}(\boldsymbol{\rho})\mathbf{G})^{-1}\mathbf{C}(\boldsymbol{\rho})\mathbf{r}-(\mathbf{1}+\mathbf{C}(\boldsymbol{\rho})\mathbf{G})^{-1}\mathbf{C}(\boldsymbol{\rho})\mathbf{v} \\
&= \mathbf{S}(\boldsymbol{\rho})\mathbf{C}(\boldsymbol{\rho})\mathbf{r}-\mathbf{S}(\boldsymbol{\rho})\mathbf{C}(\boldsymbol{\rho})\mathbf{v} \qquad (4.10\text{b})
\end{aligned}$$

where $\mathbf{r}$ is the reference value for the measurements $\mathbf{y}(\boldsymbol{\rho})$, $\mathbf{u}(\boldsymbol{\rho})$ is the actuator variable and $\mathbf{v}$ is a noise signal for the system which is presented in deviation variables. $\mathbf{S}(\boldsymbol{\rho})$ and $\mathbf{T}(\boldsymbol{\rho})$ are the sensitivity and the complementary sensitivity functions respectively. Given a desired reference model for the closed loop $\mathbf{T}^d$, the desired response from the loop is given as $\mathbf{y}^d = \mathbf{T}^d\mathbf{r}$. The performance criterion can then be formulated as a typical quadratic cost function

$$F(\boldsymbol{\rho}) = \frac{1}{2N}\mathrm{E}\left[\sum_{t=1}^{N}(\mathbf{y}_t(\boldsymbol{\rho})-\mathbf{y}_t^d)^2\right] \tag{4.11}$$

where $\mathrm{E}[\cdot]$ denotes the expectation with respect to a weakly stationary disturbance, since the measurement $\mathbf{y}(\boldsymbol{\rho})$ is affected by the process and measurement noise. The formulation in (4.11) gives minimal variance control. Penalty on the control position or its increments can also be part of such a performance criterion as well. The optimal controller will be the set of controller parameters, $\boldsymbol{\rho}$, that minimizes the cost function.

$$\boldsymbol{\rho}^{opt} = \arg\min_{\boldsymbol{\rho}} F(\boldsymbol{\rho}) \tag{4.12}$$

Given a convex cost function, this minimization is equivalent to solving

$$0 = \mathbf{J}(\boldsymbol{\rho}) = \frac{\partial F}{\partial \boldsymbol{\rho}} = \frac{1}{N}\mathrm{E}\left[\sum_{t=1}^{N}(\mathbf{y}_t(\boldsymbol{\rho}) - \mathbf{y}_t^d)^T \frac{\partial \mathbf{y}_t}{\partial \boldsymbol{\rho}}\right] \tag{4.13}$$

This equation can be solved iteratively by the following scheme

$$\boldsymbol{\rho}_{i+1} = \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1}\mathbf{J}(\boldsymbol{\rho}_i) \tag{4.14}$$

where $\mathbf{R}$ is some positive definite matrix. In case $\mathbf{R} = \mathbf{I}$ the algorithm steps in the steepest decent direction. In case $\mathbf{R} = \mathbf{H}(\boldsymbol{\rho}) = {}^{\partial^2 F}/_{\partial \boldsymbol{\rho}^2}$ or an approximation to the Hessian, the Newton or Gauss-Newton algorithm appears. γ_i determines the step length and the choice of $\mathbf{R}$ and $\boldsymbol{\gamma}$ will thus affect the convergence properties of the method (Hjalmarsson *et al.*, 1994b; Nocedal and Wright, 1999).

The problem involved with the optimization of performance through this scheme is that the actual process model often is unknown. That implies that the sensitivity functions, $\mathbf{T}$ and $\mathbf{S}$, are unknown and it is therefore not possible to calculate ${}^{\partial \mathbf{y}}/_{\partial \boldsymbol{\rho}}$ and thus $\mathbf{J}(\boldsymbol{\rho})$. Iterative Feedback Tuning solves this problem, and offers a purely data driven algorithm. With respect to tuning of inventory controllers with imperfect process models, the true sensitivity function is unknown. This constitutes a problem since it is the performance of the actual loop that is subject to optimization, and hence motivates application of the Iterative Feedback Tuning.

4.4 Iterative Fedback tuning

The key contribution in Iterative Feedback Tuning is that it supplies an unbiased estimate of the cost function gradient without estimating a plant model, $\hat{\mathbf{G}}$, given that the noise $\mathbf{v}$ is a zero mean, weakly stationary random signal (Hjalmarsson *et al.*, 1998). Using an estimate of the Jacobian in (4.14) instead of the analytical Jacobian, as a stochastic approximation method, will still make the algorithm converge to a local minimizer, provided that the estimate is unbiased, the Jacobian, $\mathbf{J}(\boldsymbol{\rho})$, is a monotonically increasing function and the sequence of γ_i fulfills condition (4.15) (Robbins and Monro, 1951).

$$\sum_{i=1}^{\infty}\gamma_i^2 < \infty, \quad \sum_{i=1}^{\infty}\gamma_i = \infty \tag{4.15}$$

This condition is fulfilled by having $\gamma_i = {}^{a}/_{i}$ where a is some constant. This method however has a convergence rate which is too slow for most industrial purposes (Mišković, 2006). In cases where the variance of the Jacobian approaches zero due to a large number of data points classic Gauss-Newton optimization with $\gamma_i = 1$, may be used instead to speed up convergence.

By differentiation of equation (4.10) it can be shown that

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \mathbf{S}(\boldsymbol{\rho})\mathbf{G}\frac{\partial \mathbf{C}}{\partial \boldsymbol{\rho}}(\mathbf{r} - \mathbf{y}(\boldsymbol{\rho})) \tag{4.16}$$

The data needed for estimation of the gradient $\mathbf{J}(\boldsymbol{\rho})$ can therefore be generated from two types of closed loop experiments on the system. First the system is run in nominal mode which reflects the normal operation for which good performance is desired, and the sequence $\mathbf{y}_1$ is collected. Secondly a set of special experiments are performed in order to get information of $\partial \mathbf{y}/\partial \boldsymbol{\rho}$. Here the reference is set to zero and the signal $\mathbf{e} = \mathbf{r} - \mathbf{y}_1$ filtered through $\partial \mathbf{C}/\partial \boldsymbol{\rho}_i$ is added to the control signal in order to get an estimate of $\partial \mathbf{y}/\partial \boldsymbol{\rho}_i$ cf. (4.16). This type of experiment has to be performed as many times as the number of parameters in $\boldsymbol{\rho}$ in the controller (Hjalmarsson and Birkeland, 1998). For SISO systems the number of necessary experiments are reduced to one, since scalar linear operators commute.

$$\frac{\partial y}{\partial \boldsymbol{\rho}} = S(\boldsymbol{\rho})G\frac{\partial C}{\partial \boldsymbol{\rho}}(r-y) = C(\boldsymbol{\rho})^{-1}\frac{\partial C}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})GC(\boldsymbol{\rho})(r-y(\boldsymbol{\rho})) \tag{4.17}$$

that implies that the gradient estimate can be formed by filtering y_2 through $C(\boldsymbol{\rho})^{-1}\frac{\partial C}{\partial \boldsymbol{\rho}}$ when the reference signal in the gradient experiments is $r_2 = r_1 - y_1$ hence only one gradient experiment is since the filtering is not performed prior to the experiment. Jansson and Hjalmarsson (2004) suggests this strategy for MIMO system as an approximation and provides sufficient conditions for local convergence in the vicinity of the optimum. In case this approximation causes the algorithm not to step in a descent direction, due to the error caused by non commuting matrices, the full method have to be applied.

The requirements on the controller are that the controller transfer function itself and $\frac{\partial C}{\partial \boldsymbol{\rho}}$ or $C(\boldsymbol{\rho})^{-1}\frac{\partial C}{\partial \boldsymbol{\rho}}$ are proper stable filters. This is the case for tuning proportional and integral action in the feedback for inventory control. Tuning of the feed forward part can be performed too if this requirement is fulfilled.

In order to reduce the time spent on experiments in each iteration in Iterative Feedback Tuning of inventory controllers, a process model can be utilized. The first experiment which reflects the normal operation, for which good closed loop performance is desired, has to be performed on the actual system. The gradient experiments where data from the first experiments are used can then be performed by simulation. This will produce a biased but noise free estimate of the gradient of the output and hence $\mathbf{J}(\boldsymbol{\rho})$. Even though this approximation is biased, convergence may be faster since the gradient will be deterministic while the gradient estimate from classic Iterative Feedback Tuning may be affected by a poor signal to noise ratio and hence poor convergence properties (Huusom *et al.*, 2007a).

4.5 Case Study - Four Tank System

The quadruple tank process in Fig. 4.1 has received attention because it shows interesting multivariable characteristics which permit illustration and analysis of different control concepts. In spite of its simple model (4.18) derived by mass balances and the Bernoulli's flow equation, it exhibits both minimum phase and non-minimum phase behavior (Johasson, 2000). Water can be directed in different ways to the tanks dependent on the position of the three-way valves ϑ_i and the flow rate from the reservoir can be manipulated through a centrifugal pump. A pilot plant scale of this process is available at CAPEC, Dept. of Chem. Eng. for testing control structures for which the physical parameters are presented in table 4.1.

$$\frac{dV_1}{dt} = -a_1\sqrt{2gh_1} + a_3\sqrt{2gh_3} + \vartheta_1 F_1 \tag{4.18a}$$
$$\frac{dV_2}{dt} = -a_2\sqrt{2gh_2} + a_4\sqrt{2gh_4} + \vartheta_2 F_2 \tag{4.18b}$$
$$\frac{dV_3}{dt} = -a_3\sqrt{2gh_3} + (1-\vartheta_2) F_2 \tag{4.18c}$$
$$\frac{dV_4}{dt} = -a_4\sqrt{2gh_4} + (1-\vartheta_1) F_1 \tag{4.18d}$$

The model (4.18) is formulated in terms of the inventory being the liquid volume in each tank. The actual measurement from the process is the liquid level. This process therefore has a very simple transformation between the underlying dynamical system and the model in terms of inventories.

$$V_i = A_i h_i, \qquad i \in \{1, 2, 3, 4\} \tag{4.19}$$

In Andrade (2007) a centralized multivariable inventory control law for this system has been derived based on the model (4.18). The static model has been validated on steady state plant data and linear correlations, with a squared Pearson correlation coefficient of 0.999, have been fitted for h_i vs. F_j^2 in order to increase the model

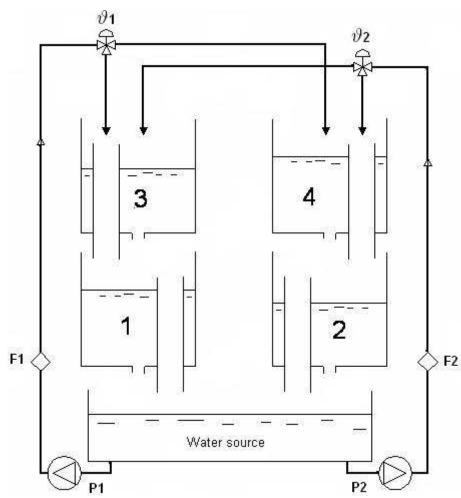


Figure 4.1. Schematic diagram of the quadruple tank process.

Symbol	**Value**	**Units**	**Parameter**
a_i	1.23	cm^2	Area of the outlet pipes
A_i	380	cm^2	Transversal area for each tank
g	981	cm/s^2	The acceleration of gravity

Table 4.1. Physical parameters for the four tank pilot plant

accuracy for the flow expressions, which in steady state are

$$h_i = \frac{1}{2ga_i^2} F_j^2 \tag{4.20}$$

The valve characteristics were investigated around the desired operation point but the model does not include the nonlinear behavior of the three way valves. These investigations show that despite some effort in the modeling of a relative simple system, the feed forward action from the inventory controller is not sufficient and both proportional and integral action will be required in the feedback in order to eliminate offset from e.g. a step response.

The objective is to control the inventories for the two lower tanks i.e. no. 1 and 2 see Fig. 4.1. The manipulated variables are the two flow rates F_1 and F_2 and the three way valves are considered as disturbance variables and will remain in a constant position through out the test. From the tank model the controlled inventories are given as:

$$\phi_1(\boldsymbol{\vartheta}, \mathbf{h}, \mathbf{F}) = -a_1\sqrt{2gh_1} + a_3\sqrt{2gh_3} + \vartheta_1 F_1 \tag{4.21a}$$

$$\phi_2(\boldsymbol{\vartheta}, \mathbf{h}, \mathbf{F}) = -a_2\sqrt{2gh_2} + a_4\sqrt{2gh_4} + \vartheta_2 F_2 \tag{4.21b}$$

since the production term is zero for this process. Utilizing the static formulation of (4.18)

$$\phi_1(\boldsymbol{\vartheta}, \mathbf{h}, \mathbf{F}) = -a_1\sqrt{2gh_1} + (1-\vartheta_2)F_2 + \vartheta_1 F_1 \tag{4.22a}$$

$$\phi_2(\boldsymbol{\vartheta}, \mathbf{h}, \mathbf{F}) = -a_2\sqrt{2gh_2} + (1-\vartheta_1)F_1 + \vartheta_2 F_2 \tag{4.22b}$$

Choosing both proportional and integral action on $e_i = V_i(t) - V_i^{set}$ in the feedback loop, and isolating the manipulated variable gives the following multivariable control law.

$$\begin{bmatrix} \vartheta_1 & (1-\vartheta_2) \\ (1-\vartheta_1) & \vartheta_2 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} = \begin{bmatrix} -K_1\left(e_1(t) + \frac{1}{\tau_{I_1}}\int_0^t e_1(\tau)d\tau\right) + a_1\sqrt{2g(h_1)} \\ -K_2\left(e_2(t) + \frac{1}{\tau_{I_2}}\int_0^t e_2(\tau)d\tau\right) + a_2\sqrt{2g(h_2)} \end{bmatrix} \tag{4.23}$$

It is clear that this control law can not be solved for any arbitrary setting of the three way valves, since $\vartheta_1 + \vartheta_2 = 1$ renders this matrix singular and hence not invertible. Johasson (2000) shows that $\vartheta_1 + \vartheta_2 < 1$ gives non-minimum phase behavior while the system is in minimum phase for $\vartheta_1 + \vartheta_2 > 1$.

Implementation of the inventory controller on the pilot plant can not be done directly since the flow rates are not free to be manipulated directly. A set of lower level SISO PI-controllers are implemented to adjust the speed of rotation for the centrifugal pumps in order to achieve the desired flow rates calculated from the inventory controller, which will act at a supervisory control layer for the underlying regulatory SISO loops. The control structure implemented on the tank system is depicted in Fig. 4.2. The tuning of the regulatory PI-control loops is performed based on IMC tuning rules and a first order model for the pump dynamics based on step response experiments. The parameters are $K_c = 0.8\, s^{-1}$ and $\tau_I = 8\, s$ for the loop controlling F_1. For the second loop controlling F_2 they are $K_c = 0.7\, s^{-1}$ and $\tau_I = 8\, s$. Both the control layers have been executed every 4 seconds. In practice this cascade structure is not effective if the underlying loops are not executed at least ten times faster than the outer loop (Wittenmark *et al.*, 1995).

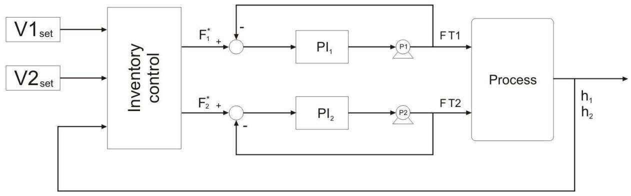


Figure 4.2. Diagram for the implemented control structure on the four tank pilot plant.

Tuning

The design objective for the tuning is chosen as a servo problem i.e. tracking a desired trajectory. A step change is introduced simultaneously to the two lower tanks, operating at steady state at the nominal operating point. After one hour the reference is stepped back to the nominal value and the experiment ends after a total of two hours. For a sample time of 4 seconds this gives 1800 data points for each of the output measurements. The nominal operating point, which is in the non-minimum phase region, is defined by

$$\begin{bmatrix} h_1^{set} \\ h_2^{set} \end{bmatrix} = \begin{bmatrix} 24\ cm \\ 21\ cm \end{bmatrix}, \quad \begin{bmatrix} \vartheta_1 \\ \vartheta_2 \end{bmatrix} = \begin{bmatrix} 0.14 \\ 0.24 \end{bmatrix},$$

The step changes applied in the reference signal are a decrease $\Delta h_i^{set} = 2\ cm$ i.e. to a level of 22 and 19 cm in the two tanks respectively.

It is desired that the two outputs of the lower tanks perform as the following second order transfer function

$$T_i^d(s) = \frac{K_{T^d}}{\tau_{T^d}^2 s^2 + 2\tau_{T^d}\xi_{T^d} s + 1}, \quad i \in \{1, 2\} \tag{4.24}$$

where $K_{T^d} = 1$, $\tau_{T^d} = 30$ and $\xi_{T^d} = 1.3$ in order to have a over damped system with DC-gain equal to one and rise time of approximately 150 seconds.

Initially the system is implemented with the following controller parameters $K_c = K_1 = K_2 = 0.0139\ s^{-1}$ and $\tau_I = \tau_{I_1} = \tau_{I_2} = 200\ s$. Performing the performance experiments on the pilot plant gave the responses in Fig. 4.3 for which the value of the cost function was evaluated to $F(\boldsymbol{\rho}_0) = 0.0574$. The initial set of controller parameters results in an over shoot, and a slower response than desired.

Tuning of a controller in operation on a real process requires several repeated experiments and is therefore rather time-consuming. To save time and avoid noise, the gradient experiment are simulated and only the first experiment in each iteration is conducted as a plant experiment. This is reasonable since a very good process model is available. The gradient experiment is further more the SISO formulation of the gradient experiment, which also introduces an error in the gradient experiment. This is necessary since the data filters from the gradient of the controller causes problems, while filtering though $\mathbf{C}^{-1}\frac{\partial \mathbf{C}}{\partial \boldsymbol{\rho}}$ does not. Despite these error sources, the tuning has been successfully performed. The results are presented in table 4.2 and the response using the final set of controller parameters are shown in figure 4.4.

Controller	$\mathbf{K}_1 \cdot 10^3$	$\boldsymbol{\tau}_{I_1}$	$\mathbf{K}_2 \cdot 10^3$	$\boldsymbol{\tau}_{I_2}$	$\mathbf{F}_{sim}$	$\mathbf{F}_{exp}$
C_0	13.9	200	13.9	200	0.0279	0.0574
C_1	15.2	171	19.5	505	0.0220	0.0502
C_2	18.6	237	19.8	1275	0.0128	0.0365
C_3	23.1	346	14.4	1401	0.0097	0.0384
C_4	26.5	483	12.3	1363	0.0100	0.0364

Table 4.2. Result of the iterative controller tuning. For each controller the parameters are presented together with the value of the performance cost function based on both a noise free simulation and an experiment on the pilot plant.

From the values of the cost function, F, it can be concluded that the method does decrease the specified performance cost based on evaluation of the cost from pilot plant experiments and noise free simulations. The value of the cost function has dropped 37 % in 4 iterations based on the pilot plant experiments and from Fig. 4.4 it is clear that the tuning has reduced the over shoot substantially. It is seen that the control has become more aggressive which corresponds well with the minimal variance design of the cost function. From the development of the controller parameters it is clear that the dynamics of the two separate lower tanks is different which renders the parameters from these two loops deviate. The coupling between the tanks through the three way valves may also contribute to produce a complicated curvature of the cost function, which is indicated by the non monotonous development of the parameters.

4.6 Conclusions

Criteria based tuning of inventory controllers has to rely on data driven methods due to modeling bias. Iterative Feedback Tuning has been shown to be an amenable method for tuning the feedback in inventory controllers, and the process model from the design of the inventory control can be used to simplify the steps in the iteration by simulating the gradient experiments. This approximation will give bias to the gradient estimate but the estimate will be noise free. Tuning of a multivariable inventory controller implemented on a four tank system show a clear improvement in performance in only 4 iterations.

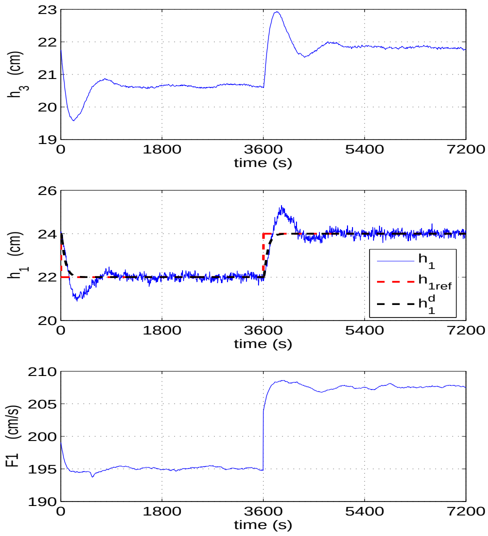


Figure 4.3. Dynamic response of the pilot plant to +2 *cm*, simultaneous step changes in the reference to the two lower tanks. The responses are shown for the liquid level in all four tanks together with the desired response on the lower tanks. Furthermore the responses in the manipulated variable from the inventory controller are given. The implementation of the controller is based on the initial set of controller parameters.

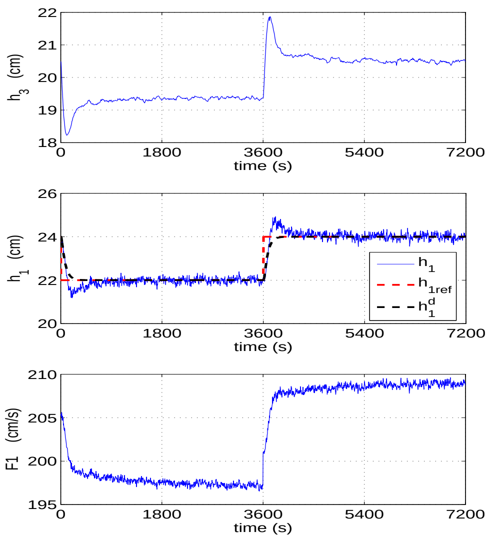


Figure 4.4. Dynamic response of the pilot plant to +2 *cm*, simultaneous step changes in the reference to the two lower tanks. The responses are shown for the liquid level in all four tanks together with the desired response on the lower tanks. Furthermore the responses in the manipulated variable from the inventory controller are given. The implementation of the controller is based on the tuned set of controller parameters after 4 iterations.

5

Data Driven Tuning of State Space Controllers with Observers

Abstract

Iterative Feedback Tuning is a purely data driven tuning algorithm for optimizing control parameters based on closed loop data. The algorithm is designed to produce an unbiased estimate of a performance cost function gradient and the control parameters are iteratively improved in order to achieve optimal loop performance. This tuning method has been derived for and is widely applied on systems using a transfer function representation.

In this paper equivalent forms are found for a control system in a state space representation, with state observer and proportional feedback, and in a transfer function representation. It is shown how the parameters in the transfer function, describing the feedback control of a state space system, can be tuned by Iterative Feedback Tuning. A simulation example illustrates that the tuning converges to known analytical solutions for the feedback control gain and the Kalman gain in the state observer.

5.1 Introduction

The need for optimal process operation has rendered methods for optimization of control loop parameters an active research area. Much attention has been directed in performing control oriented system identification, which implies model estimation from closed loop data (Schrama, 1992b; Hjalmarsson *et al.*, 1994a; Gevers, 2002). Optimizing the parameters in a control loop is an iterative procedure since the data from one experiment will depend on the current controller, and repeated iteration is necessary for the loop performance to converge to a minimum. Estimating a model from closed loop data requires special techniques (Ljung, 1999) and several algorithms have been published which handle the iterative scheme of closed loop system identification and model based control design (Zang *et al.*, 1995; Gevers *et al.*, 2003a; de Callafon, 1998). An alternative would be a direct data driven approach to tuning of the control parameters without utilizing a model estimate.

Data driven tuning methods have mainly been reported for systems given in transfer function form. Examples are the Iterative Feedback Tuning method (Hjalmarsson *et al.*, 1998) and in recent years the Correlation based Tuning (Karimi *et al.*, 2004) and Virtual Reference Feedback Tuning (Campi *et al.*, 2002). Controllers which are based on state space description of the system model are mainly tuned based on an estimated process model. Hence the potential advantages by using a direct tuning method are not exploited. These advantages are that direct tuning often is computational less demanding than model identification and model based control design. The direct tuning methods can be used even when insufficient knowledge of the model structure limits the performance, where the system is tuned based on the certainty equivalence principle.

This paper intends to investigate the use of the direct tuning method, Iterative Feedback Tuning, for optimization of the feedback gain and the state observer gain for a control loop based on a state space system description. Based on the certainty equivalence principle, an analytical solution for optimal values of these two gains exists. This renders the loop performance sensitive to model errors and bias. The data driven tuning in this paper will only be investigated for systems with full process insight.

The perspective for simple tuning methods for control structures based on state space descriptions is very interesting. The majority of advanced control strategies are today model based and rely on state space descriptions. Direct controller tuning may serve as an interesting alternative, when fine tuning a control loop or when a degrading loop performance is observed.

This paper is organized as follows. In Section 5.2 a short introduction to the system and control loop description is given together with the optimal model based design. In Section 5.3 the data driven tuning method, Iterative Feedback Tuning, is presented and Section 5.4 analyses the state space formulation in relation to the tuning method. An illustrative simulation example is given in Section 5.5 before final conclusions are drawn.

5.2 The State Space Control Loop

Given the following linear, discrete time, single input/single output, time-invariant system description:

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{A}\mathbf{x}_t + \mathbf{B}u_t + \mathbf{e}_t^p, \quad & \mathbf{e}_t^p &\in \mathcal{N}_{iid}(0, \mathbf{P}_{e^p}) \\ y_t &= \mathbf{C}\mathbf{x}_t + e_t^m, & e_t^m &\in \mathcal{N}_{iid}(0, \sigma_{e^m}^2) \end{aligned} \tag{5.1}$$

where $\mathbf{x}_t$ is the system states, u_t is the manipulated variable and y_t is the system output at time instant $t \in \mathbb{Z}$. $\mathbf{e}_t^p$ represents process noise and e_t^m is measurement noise. The cross correlation between $\mathbf{e}_t^p$ and e_t^m will be assumed zero in this paper. It is desired to control this system using the proportional feedback law

$$u_t = -\mathbf{L}\mathbf{x}_t + Mr_t \tag{5.2}$$

where $\mathbf{L}$ is a constant feedback gain matrix and M is a controller gain for the reference signal. Since the exact value of the states are not known, an observer is used to generate state estimates based on measurements of the process output and a process model

$$\begin{aligned} \mathbf{x}_{t+1} &= \hat{\mathbf{A}}\mathbf{x}_t + \hat{\mathbf{B}}u_t + \mathbf{e}_t^p, \quad & \mathbf{e}_t^p &\in \mathcal{N}_{iid}(0, \hat{\mathbf{P}}_{e^p}) \\ y_t &= \hat{\mathbf{C}}\mathbf{x}_t + e_t^m, & e_t^m &\in \mathcal{N}_{iid}(0, \hat{\sigma}_{e^m}^2) \end{aligned} \tag{5.3}$$

The observer has the form of the predictive Kalman filter with the constant gain matrix $\mathbf{K}$, assuming stationary conditions.

$$\begin{aligned} \hat{\mathbf{x}}_{t+1|t} &= \hat{\mathbf{A}}\hat{\mathbf{x}}_{t|t-1} + \hat{\mathbf{B}}u_t + \mathbf{K}\left(y_t - \hat{\mathbf{C}}\hat{\mathbf{x}}_{t|t-1}\right) \\ u_t &= -\mathbf{L}\hat{\mathbf{x}}_{t|t-1} + Mr_t \end{aligned} \tag{5.4}$$

The structure of the state space feedback loop with observer, consisting of equation (5.1) and (5.4), is shown in Figure 5.1. In order to have a static gain from the reference to the process output equal to one, the following requirements can be derived based on an assumption of full state information

$$M = \left[\hat{\mathbf{C}}\left[\mathbf{I} - \hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{L}\right]^{-1}\hat{\mathbf{B}}\right]^{-1} \tag{5.5}$$

Introducing the state estimation error $\tilde{\mathbf{x}}_t = \mathbf{x}_t - \hat{\mathbf{x}}_t$, and assuming full process knowledge, the system can be represented by (5.6) which provides a convenient description with a more clear distinction between feedback control and state estimation dynamics (Åström, 1970):

$$\begin{aligned} \mathbf{x}_{t+1} &= (\mathbf{A} - \mathbf{B}\mathbf{L})\,\mathbf{x}_t + \mathbf{B}(\mathbf{L}\tilde{\mathbf{x}}_t + Mr_t) + \mathbf{e}_t^p, \quad & \mathbf{e}_t^p &\in \mathcal{N}_{iid}(0, \mathbf{P}_{e^p}) \\ \tilde{\mathbf{x}}_{t+1} &= (\mathbf{A} - \mathbf{K}\mathbf{C})\,\tilde{\mathbf{x}}_t - \mathbf{K}e_t^m + \mathbf{e}_t^p, & e_t^m &\in \mathcal{N}_{iid}(0, \sigma_{e^m}) \\ u_t &= -\mathbf{L}(\mathbf{x}_t - \tilde{\mathbf{x}}_t) + Mr_t \\ y_t &= \mathbf{C}\mathbf{x}_t + e_t^m \end{aligned} \tag{5.6}$$

If the system (5.1) is stabilizable and detectable a set $\{\mathbf{L}, \mathbf{K}\}$ exists which renders the system stable (Kwakernaak and Sivan, 1972). Hence if optimal values for the feedback and Kalman filter gains are used stability is guaranteed. Computation of these optimal gains are shown in the following subsection.

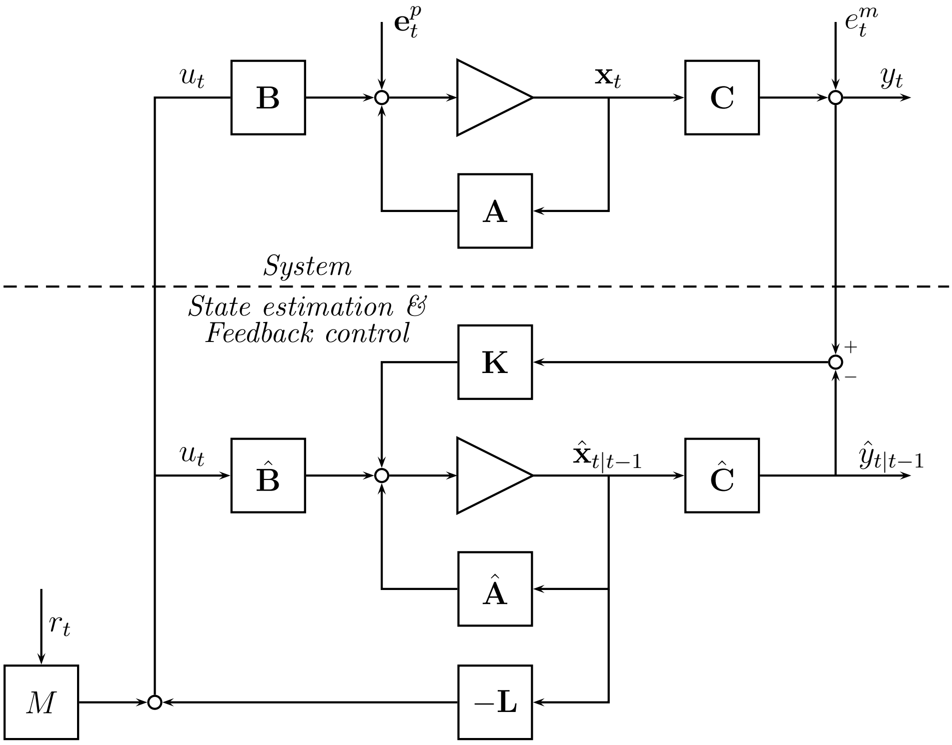


Figure 5.1. Structure of a state space feedback loop with an observer.

5.2.1 Optimal model based design

Optimal values for both the observer gain $\mathbf{K}$ and the feedback gain $\mathbf{L}$ exist and have known analytical solutions (Anderson and Moore, 1989; Grewal and Andrews, 1993).

The optimal, stationary value for the gain matrix in the predictive Kalman filter can be evaluated based on the process model and information of the noise intensity. The stationarity condition is indicated by the ∞ subscript on the gain matrix and the covariance matrix on the state prediction error $\mathbf{P}_\infty$.

$$\begin{aligned} \hat{\mathbf{K}}_\infty =& \hat{\mathbf{A}}\hat{\mathbf{P}}_\infty\hat{\mathbf{C}}^T \left[\hat{\mathbf{C}}\hat{\mathbf{P}}_\infty\hat{\mathbf{C}}^T + \hat{\sigma}^2_{e^m}\right]^{-1} \\ \hat{\mathbf{P}}_\infty =& \hat{\mathbf{A}}\hat{\mathbf{P}}_\infty\hat{\mathbf{A}}^T + \hat{\mathbf{P}}_{e^p} - \hat{\mathbf{A}}\hat{\mathbf{P}}_\infty\hat{\mathbf{C}}^T \left[\hat{\mathbf{C}}\hat{\mathbf{P}}_\infty\hat{\mathbf{C}}^T + \hat{\sigma}^2_{e^m}\right]^{-1} \hat{\mathbf{C}}\hat{\mathbf{P}}_\infty\hat{\mathbf{A}}^T \end{aligned} \tag{5.7}$$

The equation for the state prediction error variance matrix is an algebraic Riccati equation.

The optimal value for the controller gain depends on the optimization criterion. In this paper the control design will minimize the value of a cost function for the loop performance. For a single input/single output system

$$F(y,u) = \frac{1}{2N}\sum_{t=1}^{N} y_t^2 + \lambda u_t^2 \tag{5.8}$$

where λ determines the weighting between the penalty on the output and the control. For optimal tracking the output is replaced by the tracking error in the cost function. The optimal Linear Quadratic Gaussian controller (LQG) produces an optimal feedback gain for the quadratic cost function

$$F_{LQG}(y,u) = \frac{1}{2N}\sum_{t=1}^{N} \hat{\mathbf{x}}_t^T \mathbf{Q}_R \hat{\mathbf{x}}_t + \lambda u_t^2 \tag{5.9}$$

Using the linear system description in (5.3) with Gaussian noise, and assuming the horizon in the criterion approaches infinity, will produce the following stationary solution for the controller gain:

$$\begin{aligned}\mathbf{L}_\infty &= \left[\hat{\mathbf{B}}^T \mathbf{S}_\infty \hat{\mathbf{B}} + \lambda\right]^{-1} \hat{\mathbf{B}}^T \mathbf{S}_\infty \hat{\mathbf{A}} \\ \mathbf{S}_\infty &= \hat{\mathbf{A}}^T \mathbf{S}_\infty \hat{\mathbf{A}} + \mathbf{Q}_R - \hat{\mathbf{A}}^T \mathbf{S}_\infty \hat{\mathbf{B}} \left[\hat{\mathbf{B}}^T \mathbf{S}_\infty \hat{\mathbf{B}} + \lambda\right]^{-1} \hat{\mathbf{B}}^T \mathbf{S}_\infty \hat{\mathbf{A}}\end{aligned} \tag{5.10}$$

This set of equations are on the same form, as for the design of the Predictive Kalman filter. It can be seen that the weights $\mathbf{Q}_R$ and λ in the cost functions play the same role in the equations as the noise variance in the filter equations. In case $\mathbf{Q}_R = \mathbf{C}^T\mathbf{C}$ the cost function (5.9) is equivalent to (5.8).

5.3 Iterative Feedback Tuning

This data driven tuning method was introduced in Hjalmarsson *et al.* (1994b) in 1994 and further developed and refined in Hjalmarsson *et al.* (1998). An extensive overview of contributions and applications for this tuning method can be found in Gevers (2002); Hjalmarsson (2002). The tuning method optimizes the control parameters based on a performance cost function as (5.8). The main idea is to use closed loop data to form an unbiased estimate of the cost function gradient with respect to the control parameters, and use that in a gradient based search algorithm.

The Iterative Feedback Tuning method works with a system description for a feedback loop with a two degree of freedom controller, $C = \{C_r, C_y\}$, as shown in Figure 5.2 and Equation 5.11. The process model G is a discrete time transfer function and the noise v_t is a zero mean, weakly stationary random signal.

$$y_t = \frac{GC_r}{1+GC_y} r_t + \frac{1}{1+GC_y} v_t = T r_t + S v_t \tag{5.11a}$$

$$u_t = \frac{C_r}{1+GC_y} r_t - \frac{C_y}{1+GC_y} v_t = S C_r r_t - S C_y v_t \tag{5.11b}$$

where S and T is the sensitivity and the complementary sensitivity function respectively.

Based on the general cost function with penalty on the tracking error $\tilde{y}_t = y_t - y_t^d$

$$F(y,u) = \frac{1}{2N}\sum_{t=1}^{N} \tilde{y}_t^2 + \lambda u_t^2 \tag{5.12}$$

and the system description (5.11), Hjalmarsson *et al.* (1998) shows that the cost function gradient with respect to the control parameters is

$$\frac{\partial F}{\partial \boldsymbol{\rho}} = \frac{1}{N}\mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho}) \frac{\partial \tilde{y}_t}{\partial \boldsymbol{\rho}} + \lambda \sum_{t=1}^{N} u_t(\boldsymbol{\rho}) \frac{\partial u_t}{\partial \boldsymbol{\rho}}\right] \tag{5.13}$$

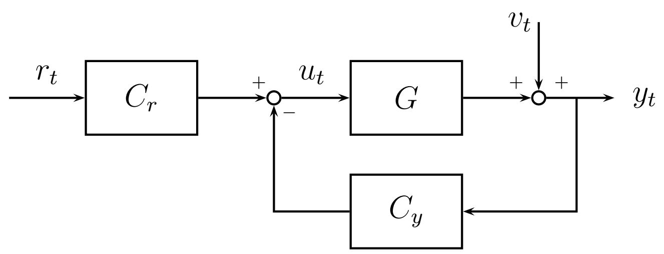


Figure 5.2. *Feedback loop with a two degree of freedom controller.*

where E[·] is the mathematical expectation. The derivative of the in- and outputs are given by

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) T(\boldsymbol{\rho})\mathbf{r} + \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})(\mathbf{r}-\mathbf{y}) \tag{5.14a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) S(\boldsymbol{\rho})\mathbf{r} + \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})(\mathbf{r}-\mathbf{y}) \tag{5.14b}$$

When the cost function (5.8) with $\lambda = 0$ is used, which was used in the LQG design, the problem reduces to a disturbance rejection problem where $r_t = 0$. Hence the expression for the gradients reduces to

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = -\frac{\partial C_y}{\partial \boldsymbol{\rho}} G S(\boldsymbol{\rho})\mathbf{y} \tag{5.15a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = -\frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})\mathbf{y} \tag{5.15b}$$

Given the cost function gradient estimate the update of the control parameters in the optimization are performed by iterations in

$$\boldsymbol{\rho}_{i+1} = \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1} \widehat{\frac{\partial F(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \tag{5.16}$$

where γ_i is the step length and $\mathbf{R}_i$ is some positive definite matrix, preferably the Hessian estimate of the cost function

$$\mathbf{R} = \frac{1}{N}\sum_{t=1}^{N}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\right)^T + \lambda \widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\right)^T\right) \tag{5.17}$$

This optimization will converge despite the stochastic nature of the cost function gradient as a stochastic approximation method as long as an optimum exists, the estimate is unbiased and the following condition on the step size is fulfilled (Robbins and Monro, 1951).

$$\sum_{i=1}^{\infty} \gamma_i^2 < \infty, \quad \sum_{i=1}^{\infty} \gamma_i = \infty \tag{5.18}$$

This condition is fulfilled by having $\gamma_i = {}^a\!/_i$ where a is some constant. In case where the number of data points, N, is very large the variance of the gradient estimate becomes small, and a faster converging gradient scheme than the stochastic approximation may perform well.

5.3.1 The tuning algorithm

In order to form the estimate of the cost function gradient in (5.14), measurements of the system in- and output and their derivatives with respect to the control parameters are needed. The following three closed loop experiments are performed on the system, where the superscripts refer to the experiment number.

$$
\begin{aligned}
\text{Ex. \# 1:} \quad & y_t^1 = T(\boldsymbol{\rho})r_t + S(\boldsymbol{\rho})v_t^1 \\
& u_t^1 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})r_t - C_y(\boldsymbol{\rho})v_t^1\right) \\
\text{Ex. \# 2:} \quad & y_t^2 = T(\boldsymbol{\rho})(r_t - y_t^1) + S(\boldsymbol{\rho})v_t^2 \\
& u_t^2 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})(r_t - y_t^1) - C_y(\boldsymbol{\rho})v_t^2\right) \\
\text{Ex. \# 3:} \quad & y_t^3 = T(\boldsymbol{\rho})r_t + S(\boldsymbol{\rho})v_t^3 \\
& u_t^3 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})r_t - C_y(\boldsymbol{\rho})v_t^3\right)
\end{aligned}
$$

where r_t is the reference signal during normal operation. The sequence of input/output data from these experiments $(y^j; u^j)$ $j \in \{1, 2, 3\}$ will be utilized as

$$\tilde{y}_t = y_t^1 - y_t^d \tag{5.19a}$$

$$u_t = u_t^1 \tag{5.19b}$$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{y}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{y}^2\right] \tag{5.19c}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{u}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{u}^2\right] \tag{5.19d}$$

The estimator of the in- and output gradients can be written as

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \tag{5.20a}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \tag{5.20b}$$

It can be seen from these equations that only the noises in the last two experiments contribute as a nuisance, since they contribute to the variance. The noise in the first experiment, in contrast, contributes to the analytical part of the gradients from (5.14).

In case the tuning algorithm is used for disturbance rejection, i.e. $r_t = 0$, the third experiment is redundant. The tuning algorithm can be summarized as

- *Collect* $(y_t^j; u_t^j)$ $j \in \{1, 2, 3\}$ *from the three closed loop experiments with the controller* $C(\boldsymbol{\rho}_i)$.

- *Evaluate the gradient of the cost function* ${}^{\partial F(\boldsymbol{\rho}_i)}/_{\partial \boldsymbol{\rho}}$*, the* $\boldsymbol{R}_i$ *matrix and update the control parameters to* $\boldsymbol{\rho}_{i+1}$.

- *Evaluate the performance* $F(\boldsymbol{\rho}_{i+1})$ *and repeat with* $i = i + 1$ *if the performance tolerance is not achieved.*

5.4 Transforming the State Space Formulation

The restrictions which the Iterative Feedback Tuning method sets on the control strategy used, are that the controller and the partial derivatives of the controller, with respect to the control parameters, can be formulated on transfer function form. It must be required that the filters in (5.19) are proper and stable. If the derivative of the controller is unstable it is required to include filters in the performance cost function to compensate and ensure a bounded output from the filtering (Hjalmarsson *et al.*, 1998).

Using the system description (5.1), the system model (5.3) and the observer based feedback law (5.4) a transfer function description of the system and the feedback connection can be produced by elimination of the states. The conversion from discrete time, state space description, to the equivalent discrete time transfer function from for the true system (5.1) is given by

$$G(q) = \mathbf{C}\left(q\mathbf{I} - \mathbf{A}\right)^{-1}\mathbf{B} \tag{5.21}$$

where q is the one step, forward shift operator, i.e. in general it applies that $\psi_{t+i} = q^i\psi_t$. The transfer function for the feedback connection is achieved by substitution of the observer based feedback law (5.4) in the equation for the system model (5.3) and performing the conversion

$$C_y(q) = \mathbf{L}\left[q\mathbf{I} - \left(\hat{\mathbf{A}} - \hat{\mathbf{B}}\mathbf{L} - \mathbf{K}\hat{\mathbf{C}}\right)\right]^{-1}\mathbf{K} \tag{5.22}$$

The controller, C_r, which represents the transfer from the reference to the control signal can also be derived from Equations (5.4) and (5.3). The controller will include the dynamics of the observer loop.

$$C_r(q) = M - \mathbf{L}\left[q\mathbf{I} - \left(\hat{\mathbf{A}} - \hat{\mathbf{B}}\mathbf{L} - \mathbf{K}\hat{\mathbf{C}}\right)\right]^{-1}\hat{\mathbf{B}}M \tag{5.23}$$

In the special case of a first order system, i.e. a scalar state vector, C_r simplifies to

$$C_r(q) = \frac{M(q - \hat{A} + K\hat{C})}{q - \hat{A} + \hat{B}L + K\hat{C}} \tag{5.24}$$

The interconnection between the plant and the controller transfer functions is depicted on Figure 5.2.

The transfer function description given in this section and for the Iterative Feedback Tuning method only includes additive noise on the output in contrast to the state space description which provides a clear distinction between process and measurement noise. From (5.6) it can be derived that the following identity render the two descriptions identical.

$$\begin{aligned} v_t =& \mathbf{C}\left[q\mathbf{I} - (\mathbf{A} - \mathbf{B}\mathbf{L})\right]^{-1}\left(\mathbf{I} + \mathbf{B}\mathbf{L}\left[q\mathbf{I} - (\mathbf{A} - \mathbf{K}\mathbf{C})\right]^{-1}\right)\mathbf{e}_t^p + \\ &\left(1 + \mathbf{C}\left[q\mathbf{I} - (\mathbf{A} - \mathbf{B}\mathbf{L})\right]^{-1}\mathbf{B}\mathbf{L}\left[q\mathbf{I} - (\mathbf{A} - \mathbf{K}\mathbf{C})\right]^{-1}\mathbf{K}\right)e_t^m \end{aligned} \tag{5.25}$$

5.4.1 Tuning potentials

It is seen from the transformation of the state space system into the transfer function form, that a controller can be derived which makes it possible to tune the control parameters using the Iterative Feedback Tuning method. The control parameters can be both the feedback and/or the observer gains, or it could also be the parameters in the model estimate since these also are an intrinsic part of controller. When full process knowledge is available, it does not make sense to find the optimal feedback and observer gain by data driven tuning. The purpose of this contribution is merely to show that it is possible and that the results from the tuning are consistent with the well known analytical results.

When full process knowledge is not available e.g. there is uncertainty in the estimated parameters or incomplete information about the noise intensity, then the values for the feedback and observer gains will be affected and so will the achieved closed loop performance. This performance deterioration is a consequence of the gains being evaluated based on the certainty equivalence principle. In case the only error is related to the information about the noise intensities it is straight forward to tune the gain in the Kalman filter, using this data driven approach. Hence Iterative Feedback Tuning provides an alternative method to tune a Kalman filter to that of direct estimation of noise intensities Åkeson *et al.* (2008). If there are errors in the parameters of the system model description as indicated in equation (5.4), the system can not be represented by (5.6). The correct representation would be

$$
\begin{aligned}
\mathbf{x}_{t+1} &= (\mathbf{A} - \mathbf{BL})\,\mathbf{x}_t + \mathbf{B}(\mathbf{L}\tilde{\mathbf{x}}_t + Mr_t) + \mathbf{e}_t^p, \quad && \mathbf{e}_t^p \in \mathcal{N}_{iid}(0, \mathbf{P}_{e^p}) \\
\tilde{\mathbf{x}}_{t+1} &= \left(\hat{\mathbf{A}} - \mathbf{K}\hat{\mathbf{C}}\right)\tilde{\mathbf{x}}_t - \mathbf{K}e_t^m + \mathbf{e}_t^p - \mathbf{\Delta}, && e_t^m \in \mathcal{N}_{iid}(0, \sigma_{e^m}) \\
\mathbf{\Delta} &= \left((\mathbf{A} - \hat{\mathbf{A}}) - \mathbf{K}(\mathbf{C} - \hat{\mathbf{C}})\right)\mathbf{x}_t + (\mathbf{B} - \hat{\mathbf{B}})(Mr_t - \mathbf{L}(\mathbf{x}_t - \tilde{\mathbf{x}}_t)) \\
u_t &= -\mathbf{L}(\mathbf{x}_t - \tilde{\mathbf{x}}_t) + Mr_t \\
y_t &= \mathbf{C}\mathbf{x}_t + e_t^m
\end{aligned}
\tag{5.26}
$$

Investigation on tuning the performance of this system with parametric uncertainty in the system model estimate is current work.

5.5 An Example

In order to illustrate the potential of using the Iterative Feedback Tuning method on a discrete time, state space system with observer and a proportional feedback gain, the following first order system is investigated.

$$
\begin{aligned}
x_{t+1} &= 0.98x_t + 0.02u_t + e_t^p, \quad && e_t^p \in \mathcal{N}_{iid}(0, \sigma_{e^p}^2) \\
y_t &= 1x_t + e_t^m, && e_t^m \in \mathcal{N}_{iid}(0, \sigma_{e^m}^2)
\end{aligned}
\tag{5.27}
$$

This system is characterized by its fairly slow dynamics and a static gain of one from the input to the output. The sample time for this system is 1 second.

The system will be implemented using the structure (5.6), where full knowledge of the process and the noise is assumed. For this system the feedback gain will be tuned for tracking and noise rejection. Furthermore both the feedback gain and the observer gain are tuned simultaneously for the noise rejection problem. The performance cost function used for the tracking problem has $\lambda = 0$, which produces the minimum variance controller. For the noise rejection problem $\lambda \in \{0; 0.001\}$.

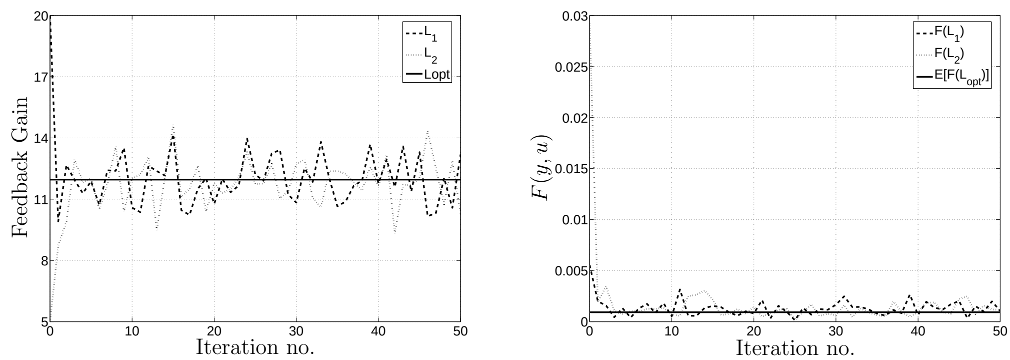
5.5.1 Tuning for Set Point Tracking

Initially the noise is characterized by $\sigma_{e^p} = 0.025$ and $\sigma_{e^m} = 0.01$ which are used with the process model to calculate the optimal Kalman filter gain by (5.7). It is desired to find the optimal feedback gain, which will render the step response of the closed loop system resemble that of a first order system with a settling time of 10 seconds. A time horizon of 20 seconds will be used in the cost function. Hence this closed loop system will have a two degree of freedom controller where C_y is given by (5.22) and C_r by (5.24). In the tuning, the control structure is treated as a two degree of freedom controller.

The optimal controller gain for this tracking problem has been determined numerically to $L_{opt} = 11.959$. For two different initial values of the feedback gain, 50 iterations in the data driven tuning have been performed, and the trajectories for the feedback gain and the performance cost function are shown in Figure 5.3. It can be seen in the figures that the tuning does converge in very few iterations to the level around the optimal value of the feedback gain and hence to the expected value of the loop performance.

5.5.2 Tuning for Disturbance Rejection

When tuning for disturbance rejection the process noise level has been increased so that $\sigma_{e^p} = 1$ and the time horizon used in the performance cost function N is extended to one hour. Since this is a disturbance rejection problem, only a one degree



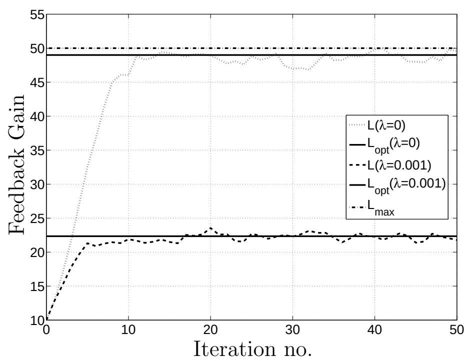
(a) Control parameter (b) Performance cost function

Figure 5.3. Development in the feedback gain and the loop performance cost when tuning the loop's step response for a tracking problem given two different initial values. The optimal value for the feedback gain and the corresponding value for the expected optimal performance are given as a full lines.

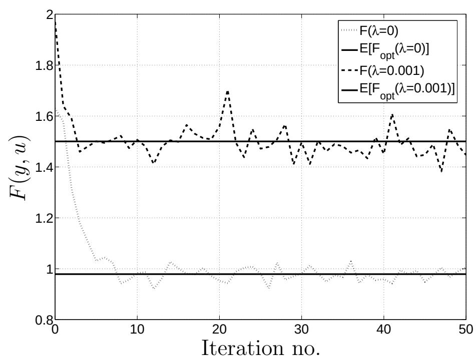
of freedom controller has been used in the tuning which means that only experiment no. one and two in the Iterative Feedback Tuning algorithm are required in each iteration, and only the gradient ${}^{\partial C_y}/_{\partial L}$ needs to be evaluated. Initially the optimal Kalman filter gain is used and only the feedback gain is tuned for both the minimum variance control problem and for $\lambda = 0.001$, Figure 5.4. The optimal feedback gain, $L_{opt}(\lambda = 0)$, is very close to a limit which would make the controller C_y unstable. The optimal values for the feedback gain is calculated by (5.10). A constraint is implemented in the control parameter update equation (5.18) which will decrease the step length γ_i from 1 in case it is predicted that $L_{i+1} > L_{max}$. L_{max} produces a controller with a pole on the stability limit. Results from simultaneous tuning of both gains with $\lambda = 0.001$ are shown in Figure 5.5. It is seen that in both cases that the tuning is able to converge to the level of the optimal values of the gains. The rate of convergence is not quite as fast as for the tracking problem. This is to be expected since the step response experiment perturbed the system more that the noise in the disturbance rejection case.

5.6 Conclusions

Equivalent forms for a closed loop control system has been found for a state space system with observer and proportional feedback gain and a transfer function system description respectively. These equivalent forms mean that a transfer function description for the feedback controller in the closed loop state space system has been derived. Hence it is shown how the data driven controller tuning method Iterative Feedback Tuning is applicable also for state space control systems. In simulation studies it is shown that the tuning method converge to known analytical solutions for the feedback gain and the Kalman filter gain in the state observer.

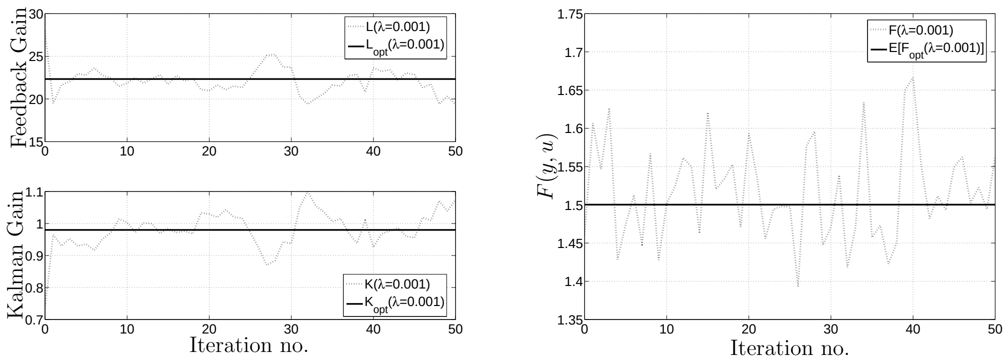


(a) Control parameter



(b) Performance cost function

Figure 5.4. Development in the feedback gain and the loop performance cost when tuning the loop for disturbance rejection given two different values of λ in the cost function. The optimal values for the gains and the corresponding values for the expected optimal performance are given as a full lines. The highest allowed value for the feedback gain L_{max} is also given.



(a) Control parameters

(b) Performance cost function

Figure 5.5. Development in both the feedback and Kalman gain in the controller and the loop performance cost when tuning the loop for disturbance rejection. The optimal values for the gains and the corresponding value for the expected optimal performance are given as a full lines.

Part III

Iterative Feedback Tuning for Disturbance Rejection

6

Introducing External Perturbations in Tuning for Disturbance Rejection

Abstract

Iterative Feedback Tuning constitutes an attractive control loop tuning method for processes in the absence of an accurate process model. It is a purely data driven approach aiming at optimizing the closed loop performance. The standard formulation ensures an unbiased estimate of the loop performance cost function gradient with respect to the control parameters. This gradient is important in a search algorithm. The extension presented in this paper further ensures informative data to improve the convergence properties of the method and hence reduce the total number of required plant experiments especially when tuning for disturbance rejection. Informative data is achieved through application of an external probing signal in the tuning algorithm. The probing signal is designed based on a constrained optimization which utilizes an approximate black box model of the process. This model estimate is further used to guarantee nominal stability and to improve the parameter update using a line search algorithm for determining the iteration step size. The proposed algorithm is compared to the classical formulation in a simulation study of a disturbance rejection problem. This type of problem is notoriously difficult for Iterative Feedback Tuning due to the lack of excitation from the reference.

6.1 Introduction

The increasing competition on the global market has rendered optimizing process operation a necessity for new as well as existing production in the chemical industry. Advanced control strategies are based on models for the specific process. These models play an important role for optimization. Control oriented process modeling is part of the advances of application oriented modeling. System identification is an area that has received much attention but within identification for control there is still room for improvement in development of systematic methods. Identification for control implies experiments where the collected data for identification are retrieved from a process operated under control i.e. in closed loop. It is however infeasible to derive rigorous dynamic models for any process. Consequently a number of data driven methods have been developed for control optimization such as Iterative Feedback Tuning, Virtual Reference Feedback Tuning and Correlation-based Tuning (Hjalmarsson *et al.*, 1994b; Campi *et al.*, 2002; Karimi *et al.*, 2004).

Two main paths have been pursued in the attempt to produce an useful algorithm for identification for control using closed loop data. The governing principle in one of these paths has been to ensure robust stability of the loop in all iterations. The paper Gevers *et al.* (2003a) handles parameter uncertainty in the estimated plant model using confidence ellipsoids and ensures robust stability of all systems within the spanned model set. During the iterations the worst case performance within the set is optimized. A similar methodology is used by de Callafon (1998), using the more conservative $\mathcal{H}_\infty$ strategy. The model and the uncertainty are identified through the dual Youla parameterization. The control design is based on μ-synthesis. These methods are attractive due to their robustness properties but they are computationally demanding, and the achieved performance may be poor due to optimization of the worst case performance. The other path optimizes the actual performance of the loop and addresses the issue of stability subsequently to an iteration between model identification and control design. In this category falls the Iterative Feedback Tuning method, which is the subject of the present paper. The key contribution presented here is an analysis of how to improve convergence of the Iterative Feedback Tuning and hence to reduce the required number of plant experiments. Improved convergence is achieved by applying an external probing signal to the process in order to optimize the information content in the data combined with the use of line search in the parameter update.

This paper is organized as follows: The coming section presents basic criterion based controller tuning and section 6.3 shows how Iterative Feedback Tuning fits into this category and how an unbiased gradient estimate is achieved from data. Section 6.4 presents the problem of lack of informative data when tuning for disturbance rejection. It is shown how probing signals can resolve this problem and how to design these. Section 6.5 presents a new algorithm for Perturbed Iterative Feedback Tuning with guaranteed informative data and discusses control parameter update. In the subsequent section simulation studies are presented before concluding remarks are given.

6.2 Criterion Based Controller Tuning

A description of a scalar closed loop system is depicted in Figure 6.1. The two degree of freedom controller, $C = \{C_r, C_y\}$, is implemented on the discrete linear time invariant system G, hence the transfer functions are given as:

$$y_t = \frac{GC_r}{1+GC_y} r_t + \frac{1}{1+GC_y} v_t = Tr_t + Sv_t \tag{6.1a}$$

$$u_t = \frac{C_r}{1+GC_y} r_t - \frac{C_y}{1+GC_y} v_t = SC_r r_t - SC_y v_t \tag{6.1b}$$

r_t is the reference value for the measurements y_t, u_t is the actuator variable and v_t is a noise signal for the system which is presented in deviation variables. S and T are the sensitivity and the complementary sensitivity functions respectively. Given a desired reference model for the closed loop T_d, the desired response from the loop is given as $y_t^d = T_d r_t$. The performance criterion, which will be a function of the control parameters, $\boldsymbol{\rho}$, can then be formulated as a typical quadratic cost function, $F(\boldsymbol{\rho})$, with penalty on deviations from the desired output and the control effort. The deviation of the outputs is given as

$$\tilde{y}_t = y_t(\boldsymbol{\rho}) - y_t^d \tag{6.2}$$

The optimal set of parameters will then require a minimization of $F(\boldsymbol{\rho})$. A solution to the minimization problem can be obtained through the iterative gradient based local search algorithm (6.3). In case the cost function is convex the minimization will converge to the global minimizer, but this is in general not true.

$$\begin{aligned} \boldsymbol{\rho}_{i+1} &= \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1} \frac{\partial F(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \\ &= \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1} \mathbf{J}(\boldsymbol{\rho}_i) \end{aligned} \tag{6.3}$$

where $\mathbf{R}_i$ is a positive definite matrix and $\mathbf{J}$ is the gradient column vector of $F(\boldsymbol{\rho})$ with respect to the control parameters $\boldsymbol{\rho}$. The i'th step is then given by $\mathbf{h}_i = -\gamma_i \mathbf{R}_i^{-1} \mathbf{J}(\boldsymbol{\rho}_i)$. In case $\mathbf{R} = \mathbf{I}$ the algorithm steps in the steepest descent direction. In case $\mathbf{R} = \mathbf{H}(\boldsymbol{\rho}) = {}^{\partial^2 F(\boldsymbol{\rho})}/_{\partial \boldsymbol{\rho}^2}$ or an approximation to the Hessian, $\mathbf{J}(\boldsymbol{\rho}_i)\mathbf{J}^T(\boldsymbol{\rho}_i)$, the Newton or Gauss-Newton algorithm appears respectively. The choice of $\mathbf{R}$ and $\boldsymbol{\gamma}$ will thus affect the convergence properties of the method (Hjalmarsson *et al.*, 1994b; Nocedal and Wright, 1999). The iteration step size γ_i can be determined using e.g. a line search method.

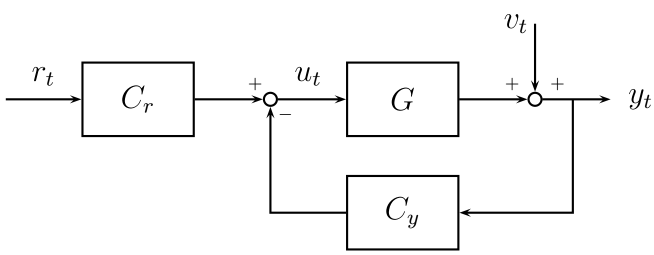


Figure 6.1. Feedback loop with a two degree of freedom controller.

6.3 Classical Iterative Feedback Tuning

This method of iterative performance enhancement does not include an estimate of the process model. The basic idea is to formulate a cost function and use the optimization algorithm (6.3) to minimize this cost function with respect to the control parameters. Evaluation of the partial derivatives of the cost function with respect to the control parameters, $\boldsymbol{\rho}$, are based on measurements taken from the closed loop system. The algorithm was first presented in Hjalmarsson *et al.* (1994b) and has been analyzed, extended and tested in a number of papers. References Gevers (2002) and Hjalmarsson (2002) provide an extensive overview of the development of the method and of some of its applications.

The key contribution in Iterative Feedback Tuning is that it supplies an unbiased estimate of the cost function gradient without requiring a plant model estimate, $\hat{G}$, given that the noise v is a zero mean, weakly stationary random signal (Hjalmarsson *et al.*, 1998). Using an estimate of the gradient vector in (6.3) instead of the analytical gradient vector, as a stochastic approximation method, will still render the algorithm converge to a local minimizer, provided that a local minimizer exists, the gradient estimate is unbiased and the sequence of γ_i fulfills condition (6.4) (Robbins and Monro, 1951).

$$\sum_{i=1}^{\infty} \gamma_i^2 < \infty, \quad \sum_{i=1}^{\infty} \gamma_i = \infty \tag{6.4}$$

This condition is fulfilled e.g. by having $\gamma_i = {}^a/_i$ where a is some positive constant. However this requirement has a convergence rate which is too slow for most industrial purposes (Mišković, 2006). In cases where the variance of the cost function gradient vector estimate approaches zero, classical Gauss-Newton optimization with $\boldsymbol{\gamma} = \mathbf{1}$ may be used instead in order to speed up the convergence. The variance of the estimate will approach zero as the number of data points approach infinity. The Gauss-Newton or other gradient based optimization methods are not guaranteed to converge when the stochastic realization of the gradient vector of the objective function change between iterations.

Given the cost function

$$F(\boldsymbol{\rho}) = \frac{1}{2N}\mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=1}^{N} u_t(\boldsymbol{\rho})^2\right] \tag{6.5}$$

where the minimization criterion is

$$\mathbf{0} = \mathbf{J}(\boldsymbol{\rho}) = \frac{1}{N}\mathrm{E}\left[\sum_{t=1}^{N} \tilde{y}_t(\boldsymbol{\rho})\frac{\partial \tilde{y}_t}{\partial \boldsymbol{\rho}} + \lambda \sum_{t=1}^{N} u_t(\boldsymbol{\rho})\frac{\partial u_t}{\partial \boldsymbol{\rho}}\right] \tag{6.6}$$

it is seen that estimates of ${}^{\partial \tilde{\mathbf{y}}}/_{\partial \boldsymbol{\rho}}$ and ${}^{\partial \mathbf{u}}/_{\partial \boldsymbol{\rho}}$ are needed in order to produce a reliable estimate of the cost function gradient vector. Since $\mathbf{y}^d$ is not a function of the control parameters, then ${}^{\partial \tilde{\mathbf{y}}}/_{\partial \boldsymbol{\rho}} = {}^{\partial \mathbf{y}}/_{\partial \boldsymbol{\rho}}$. The partial derivatives of the in- and output with respect to the control parameters can be evaluated based on equation (6.1).

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_r}{\partial \boldsymbol{\rho}}T(\boldsymbol{\rho})\mathbf{r} - \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_y}{\partial \boldsymbol{\rho}}T(\boldsymbol{\rho})\mathbf{y} \tag{6.7a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \frac{\partial C_r}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})\mathbf{r} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})\mathbf{y} \tag{6.7b}$$

or rewritten into the more favorable form.

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) T(\boldsymbol{\rho})\mathbf{r} + \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})(\mathbf{r}-\mathbf{y}) \tag{6.8a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) S(\boldsymbol{\rho})\mathbf{r} + \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})(\mathbf{r}-\mathbf{y}) \tag{6.8b}$$

where the estimates produced from this expression are expected to have a reduced variance compared to (6.7) in case $\frac{\partial C_r}{\partial \boldsymbol{\rho}} \approx \frac{\partial C_y}{\partial \boldsymbol{\rho}}$. The derivation of (6.7) and (6.8) are given in Hjalmarsson *et al.* (1998).

6.3.1 The Tuning algorithm

Estimates of the derivatives (6.8a) and (6.8b) can be realized through the following set of experiments where the superscripts refer to the experiment number.

1) $r_t^1 = r_t$ i.e. the reference in the first experiment is the same as for normal operation of the process.
2) $r_t^2 = r_t - y_t^1$ i.e. the reference in the second experiment is the difference between the ordinary reference and the output from the first experiment
3) $r_t^3 = r_t$ i.e. the reference in the third experiment is the same as for normal operation of the process, just as in the first experiment.

These experiments give the following in- and outputs

$$\begin{aligned}
\text{Ex. \# 1:} \quad & y_t^1 = T(\boldsymbol{\rho})r_t + S(\boldsymbol{\rho})v_t^1 \\
& u_t^1 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})r_t - C_y(\boldsymbol{\rho})v_t^1\right) \\
\text{Ex. \# 2:} \quad & y_t^2 = T(\boldsymbol{\rho})(r_t - y_t^1) + S(\boldsymbol{\rho})v_t^2 \\
& u_t^2 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})(r_t - y_t^1) - C_y(\boldsymbol{\rho})v_t^2\right) \\
\text{Ex. \# 3:} \quad & y_t^3 = T(\boldsymbol{\rho})r_t + S(\boldsymbol{\rho})v_t^3 \\
& u_t^3 = S(\boldsymbol{\rho})\left(C_r(\boldsymbol{\rho})r_t - C_y(\boldsymbol{\rho})v_t^3\right)
\end{aligned}$$

The sequence of input output data form these experiments $(\mathbf{y}^i; \mathbf{u}^i)$ $i \in \{1, 2, 3\}$ will be utilized as

$$\tilde{y}_t = y_t^1 - y_t^d \tag{6.9a}$$

$$u_t = u_t^1 \tag{6.9b}$$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{y}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{y}^2\right] \tag{6.9c}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{u}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{u}^2\right] \tag{6.9d}$$

The estimate of the gradients of the input and the output can be written as

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \tag{6.10a}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)\mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}}\mathbf{v}^2\right] \tag{6.10b}$$

It can be seen from these equations that only the noise in the last two experiments contribute as a nuisance, since they contribute to the variance. The noise in the first experiment, in contrast, contributes to the analytical part of the gradients from (6.8). The Iterative Feedback Tuning method is depicted in Figure 6.2. The performance check is evaluated by repeating the first experiment with the updated controller.

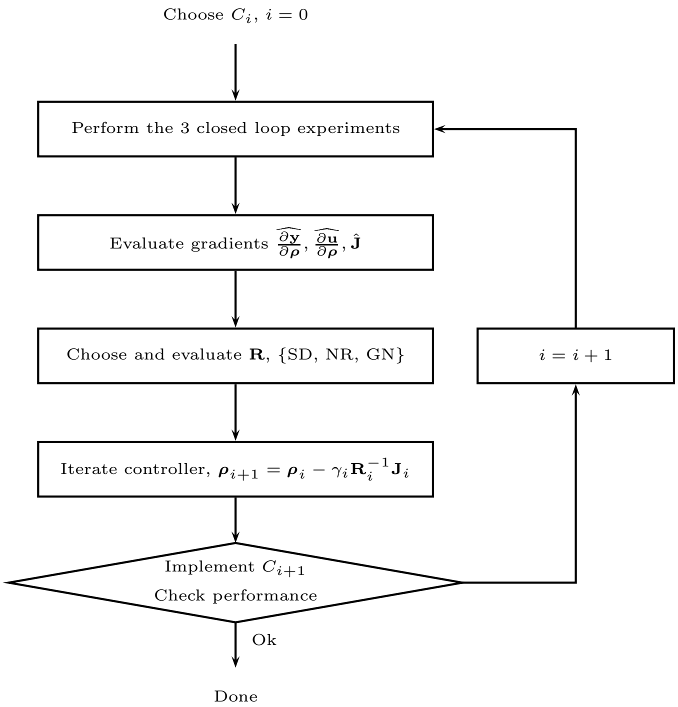


Figure 6.2. Schematic representation of the Iterative Feedback Tuning method.

6.3.2 Characteristics of Iterative Feedback Tuning

The Iterative Feedback Tuning method has several attractive properties which makes it useful for optimization of control performance when a model is unavailable. First of all Iterative Feedback Tuning utilizes closed loop data which is advantageous since it is the loop performance that is subject to optimization, and it renders the method amenable for processes where opening the loop is not an option. Secondly it is not restricted by the type of process. Even though the theory is developed for linear systems, the references Hjalmarsson (1998); Sjöberg *et al.* (2003) states that this method is applicable on some nonlinear processes as well, despite the fact that nonlinearities will generate a bias in the gradient estimate. The Iterative Feedback Tuning method does generate the true first order approximation of the gradients for a nonlinear process. The bias can be expected to be small for many practical applications (Sjöberg *et al.*, 2003) and successful tuning of PID loops for industrial processes has been reported in Lequin (1997); Hjalmarsson *et al.* (1997). The theory has furthermore been extended to cover optimization of multivariable processes, which implies that more experiments in each iteration are necessary (Hjalmarsson and Birkeland, 1998; Hjalmarsson, 1999; Jansson and Hjalmarsson, 2004) and to cover non-minimum phase and time delay systems (Lecchini and Gevers, 2002). Finally the only restriction on the control structure is that the closed loop is stable and that the transfer functions in (6.10), through which the data is filtered, are stable as well. This property extends the application beyond the classical PID control with derivative filter. The reference De Bruyne (2003) has applied the method with internal model controllers and the reference Huusom *et al.* (2007b) applies the method on a nonlinear inventory control structure. The filtering in (6.9) becomes a problem when the derivative of the controller with respect to the parameters is unstable or when C_r is non-minimum phase. Theory has been developed to cope with such difficulties by including frequency filters in the cost function which is illustrated in Hjalmarsson *et al.* (1998).

One disadvantage of using Iterative Feedback Tuning compared to model based optimization is that nominal stability can not be guaranteed. Even though the parameter update in equation (6.3) steps in a descent direction, the new controller may render the loop unstable. De Bruyne *et al.* (1999) provides an algorithm which ensures stability using the generalized stability margin, evaluated by using estimates of the closed loop transfer function. Veres and Hjalmarsson (2002); Procházka *et al.* (2005) go further and define two cost functions, one for performance and another for robustness. The performance cost is of the form (6.5) and the robustness criterion is minimizing some norm of the closed loop sensitivities, preferably the $\mathcal{H}_\infty$ norm.

Speed of convergence can also be an issue since each iteration requires a number of real plant experiments, hence the number of iterations has to be reasonably low. In case a process is tuned for disturbance rejection, it can be seen from the algorithm in section 6.3.1 and equations (6.9) and (6.10) that only the noise during the first experiment is driving the optimization. That implies that the analytic part of the gradients of the input and output may be small compared to the variance part. This poor signal to noise ratio will slow down convergence at best, compared to a situation where the reference is different from zero and the loop receives stronger excitation. In Hjalmarsson *et al.* (1998) it is shown how filtering of the reference signal before the two gradient experiments, and subsequently filtering of the input/output data from these experiments with the inverse of the filters, can improve the signal to noise ratio. E.g. let W_i^j be a set of stable and inversely stable filters for iteration i and with $j \in \{2,3\}$ as the experiment superscript in the algorithm. If $\mathbf{r}^2 = W_i^2(\mathbf{r}^1 - \mathbf{y}^1)$ and $\mathbf{r}^3 = W_i^3\mathbf{r}^1$ are used as reference signals in the gradient experiments, and if the signals $\{\mathbf{y}_i^j, \mathbf{u}_i^j\}$ are replaced by $\{(W_i^j)^{-1}\mathbf{y}_i^j, (W_i^j)^{-1}\mathbf{u}_i^j\}$ in (6.9c) and (6.9d) the filters will suppress the influence of the noise in the frequency band where it has a gain larger than one. Optimal design of the prefilters, W_i^j, have been investigated in Hildebrand *et al.* (2004, 2005b) where the asymptotic accuracy of the tuning method is improved by minimizing the covariance of the gradient estimate. An expression for this covariance is derived in Hildebrand *et al.* (2005a).

Virtual Reference Feedback Tuning (Campi *et al.*, 2002) and Correlation-based Tuning (Karimi *et al.*, 2004) are two data driven controller tuning methods which typically outperform Iterative Feedback Tuning in convergence rate and hence require fewer plant experiments. Virtual Reference Feedback Tuning only need one open or closed loop experiment to find a near optimal solution. The idea is to use a set of input/output data obtained from the plant, and a reference model for the loop. A virtual reference signal, which filtered through the reference model reproduces the plant output, is calculated and the tracking error between this virtual reference and the output can be formed. Estimation of the control parameters is then reduced to an open loop estimation problem, using the tracking error as input and the actual plant input data as output. The method is formulated using open loop and noise free data but the use of noisy data and closed loop experiments are discussed in Lecchini (2001). Correlation-based tuning uses a reduced order reference model with the desired closed loop properties to design a controller for the actual loop. Given a sequence for the reference, an output error signal can be formed as the difference between the output from the true system and the output from the designed loop. Only the output from the true system will be affected by process noise and the main idea in the tuning method is then to adjust the control parameters in order to de-correlate the output error with the reference signal. Despite the difference in criteria, this method is closely related to the Iterative Feedback Tuning method. A special formulation of the Correlation-based tuning algorithm which handles the disturbance rejection problem is given in Mišković *et al.* (2003). In the present contribution an alternative route for obtaining informative experiments and fast convergence is pursued.

6.4 Informative Experiments

The algorithm for Iterative Feedback Tuning ensures that the data from the three experiments can be used to form an unbiased estimate of the cost function gradient with respect to the control parameters. However the experiments are not necessarily performed such that a large signal to noise ratio is ensured, thus informative data is not guaranteed. From the system identification literature it is well known that external perturbation can be required in order to sufficiently excite a process. In order to identify a certain model structure and/or minimize the variance on the parameter estimate, data with sufficient information content is required (Söderström and Stoica, 1989; Ljung, 1999). This knowledge provides the inspiration to include external perturbation in the experiments conducted during each iteration of the Iterative Feedback Tuning when noise rejection is essential for closed loop performance.

External perturbation, indicated with subscript p, can be applied as a probe signal to either the reference or the control signal giving the following input output relations

$$\mathbf{y} = T(\mathbf{r} + \mathbf{r}_p) + S(\mathbf{v} + G\mathbf{u}_p) \tag{6.11a}$$

$$\mathbf{u} = SC_r(\mathbf{r} + \mathbf{r}_p) - SC_y\mathbf{v} + \mathbf{u}_p \tag{6.11b}$$

The derivatives of the input and output in (6.11) with respect to the control parameters is determined as in (6.7).

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_r}{\partial \boldsymbol{\rho}}T(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p) - \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_y}{\partial \boldsymbol{\rho}}T(\boldsymbol{\rho})\mathbf{y} \tag{6.12a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \frac{\partial C_r}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p) - \frac{\partial C_y}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})\mathbf{y} \tag{6.12b}$$

or rewritten into the more favorable form.

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})}\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) T(\boldsymbol{\rho})(\mathbf{r}+\mathbf{r}_p) + \frac{1}{C_r(\boldsymbol{\rho})}\frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})(\mathbf{r}+\mathbf{r}_p-\mathbf{y}) \tag{6.13a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right) S(\boldsymbol{\rho})(\mathbf{r}+\mathbf{r}_p) + \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})(\mathbf{r}+\mathbf{r}_p-\mathbf{y}) \tag{6.13b}$$

By conducting the three experiments described in section 6.3.1 with the addition of the signals $\{\mathbf{r}_p^i, \mathbf{u}_p^i\}$, where $i \in \{1,2,3\}$, it can be seen by applying the signals as in (6.9), that not every type of perturbation strategy should be recommended:

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)(\mathbf{v}^3 + G\mathbf{u}_p^3) + \frac{\partial C_y}{\partial \boldsymbol{\rho}}(\mathbf{v}^2 + G\mathbf{u}_p^2)\right] \tag{6.14a}$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})}\left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}}\right)(\mathbf{v}^3 + G\mathbf{u}_p^3) + \frac{\partial C_y}{\partial \boldsymbol{\rho}}(\mathbf{v}^2 + G\mathbf{u}_p^2)\right] \tag{6.14b}$$

The signals $\mathbf{u}_p^2$ and $\mathbf{u}_p^3$ will give a contribution to the variance part of the gradient estimate hence these signals should be avoided. Adding a probing signal to the reference will on the other hand always contribute to the analytic part of $\widehat{\partial \mathbf{y}/\partial \boldsymbol{\rho}}$ and $\widehat{\partial \mathbf{u}/\partial \boldsymbol{\rho}}$, but note that it must be required that $\mathbf{r}_p^2 = \mathbf{r}_p^3$ in order to construct equation (6.13) from (6.12).

Which, and how many of the probing signals, $\{\mathbf{r}_p^1, \mathbf{r}_p^2, \mathbf{r}_p^3, \mathbf{u}_p^1\}$, one should use in Iterative Feedback Tuning is less evident. One could argue that it would not be an advantage to apply perturbation in the two last experiments since the second experiment is already perturbed with the output from the first. Even though all experiments are conducted in closed loop, it is desired not to disturb the process more than necessary. Choosing between perturbing either the reference or the control signal in the first experiment is of little consequence. Identical results in the output can be achieved using $\mathbf{r}_p^1$ or $\mathbf{u}_p^1 = C_r\mathbf{r}_p^1$. Intuitively it seems more reasonable to use $\mathbf{u}_p^1$ rather than $\mathbf{r}_p^1$ since this choice will not affect $\mathbf{y}^d$ in the cost function.

When applying the Iterative Feedback Tuning method for performance optimization, the achieved set of control parameters will be, $\boldsymbol{\rho}_n$, which is a stochastic variable. Hence there will be an error between the achieved set of control parameters and the optimal set, $\boldsymbol{\rho}^{opt}$, which will minimize the expected performance cost. Assuming n to be large this error will only be associated with the noise of the system while the expected value $\mathrm{E}\left[\boldsymbol{\rho}_n - \boldsymbol{\rho}^{opt}\right]$ will be equal to zero, hence the difference between the expected value of the achieved performance cost and the optimal is a variance error. Introducing perturbations on e.g. the reference in the first experiment in the Iterative Feedback Tuning algorithm imply that the method achieves the set of control parameters $\boldsymbol{\rho}_n(\mathbf{r}_p)$. For large values of n the expected value of $\mathrm{E}\left[\boldsymbol{\rho}_n(\mathbf{r}^p) - \boldsymbol{\rho}^{opt}\right]$ is in general non zero. The difference between the expected value of the achieved performance cost and the optimal will be associated with both a bias and a variance error. The bias error is due to the fact that it is the performance of the perturbed process, and not the performance for the normal operation that becomes subject to optimization. This means that the objective transforms to a disturbance rejection problem with both process noise and an external perturbation signal. Introducing external perturbation will in general be associated with a bias error, but the variance error will decrease due to better signal to noise ratio in the data used by the tuning method. Hence the aim is to find a perturbation signal which balance these two errors and render

$$\mathrm{E}\left[F(\boldsymbol{\rho}_n(\mathbf{r}_p))\right] < \mathrm{E}\left[F(\boldsymbol{\rho}_n)\right] \tag{6.15}$$

The bias error will in general be a consequence of introducing external perturbation. Design of perturbation signals which renders unbiased or minimal bias is current work. Adding perturbations will change the curvature of the performance cost function, hence it may change the location of the optimum and should change the rate of convergence. The perturbed problem will, as the classical, converge to a local minimum of the performance cost, if the perturbations signal is bounded. Hence the two problems belong to the same class of optimization problems for which convergence has been established (Hjalmarsson, 2002).

6.4.1 Probe signal design

Design of the probing signal aims at obtaining as rich an information content in data as possible, without disturbing the process more than necessary. Therefore the probe signal design will be formulated as a constrained optimization problem. A high information content will correspond to shaping the Hessian of the cost function, i.e. rendering it large in some sense and make the optimum more distinct (Goodwin and Payne, 1977). The information content may be evaluated by the numerical value of a scalar function of the Hessian e.g. the trace or the determinant of the matrix. The value of the cost function with the current controller and for a given perturbation signal should be constrained by a maximum value. An alternative constraint condition could be to limit the intensity of the perturbation signal itself. The subject for optimization will be a parameterization, $\boldsymbol{\vartheta}$, of the probing signal e.g. the parameters in a data filter or the amplitude and frequency for a number of sinusoids. Choosing the determinant as the scalar function the design of probing signal can be formulated as

$$\begin{aligned} \boldsymbol{\vartheta}^{opt} &= \arg \underset{\boldsymbol{\vartheta}}{max} \det(\mathbf{H}) \\ s.t. \qquad & F(\boldsymbol{\rho}, \boldsymbol{\vartheta}) \leq F^{max} \end{aligned} \tag{6.16}$$

In order to compute $\boldsymbol{\vartheta}^{opt}$ and thus generate an optimal probing signal, it is necessary to be able to evaluate both the cost function, F, and a full rank Hessian or Hessian approximation for any given set of $\boldsymbol{\vartheta}$. That would require knowledge of the true system or an evaluation based on the Iterative Feedback Tuning. The latter would imply an unreasonably large number of experiments. Instead the optimization is based on a model estimate. This model estimate will be used only as an approximation of the true system for the optimization of $\boldsymbol{\vartheta}$ to define the perturbation signal. It will not be utilized for the gradient evaluation in the Iterative Feedback Tuning that optimizes $\boldsymbol{\rho}$ based on the true closed loop performance. The reason being that Iterative Feedback Tuning is a model free tuning method, which can be applied in cases where a model based control design is not possible due to lack of a sufficiently good model. Using a very crude model estimate in the perturbation signal optimization may be sufficient to produce a perturbation signal which significantly improves the convergence of the tuning. Knowing the process gain may be useful e.g. in determining the intensity of the perturbation signal. Performing a model based control design based on the crude model estimate could serve as an initial starting point for the control parameters, but it is in this context assumed that this initial controller does not perform sufficiently well and that the data driven tuning is necessary.

Having the optimization of the perturbation signal based on an approximate model implies that using this signal on the true process, the input output data may not satisfy the constraint exactly. How large such a constraint violation can become will of cause depend on the accuracy of the model, but in cases where this might be of concern, a more conservative choice of F^{max} may be appropriate. The plant model mismatch will also affect $\boldsymbol{\vartheta}^{opt}$. This is unavoidable, but the effect is judged to be of limited consequence. The system will in any case be perturbed, hence faster converge of the tuning is achieved.

It will be necessary to apply this optimization of perturbation signals before each iteration in the Iterative Feedback Tuning. Since the control is tightened through the iterations, stronger perturbations can be allowed as one proceeds through the iterations, while satisfying the performance constraint. Since the design of the perturbation signal is model based, it can be calculated offline before a new iteration begins. The optimization problem (6.16) does not restrict neither the system nor the parameterization of the perturbation. Convexity is therefore not guaranteed.

6.5 Perturbed Iterative Feedback Tuning

Applying perturbation in Iterative Feedback Tuning calculated based on a plant model estimate introduces a few new elements in the algorithm, as shown in Figure 6.3 with gray shaded background. In this illustration of the algorithm it is assumed that the perturbation is added to the control signal as a filtered white noise signal.

The workflow in Figure 6.3 shows that the optimization of the probe signal is performed after each update of the controller, when a new iteration is required. The initial probe signal can also be based on the optimization if a plant model is determined a priori. In absence of a model estimate $\boldsymbol{\vartheta}_0$ will have to be selected by the user. The data from the perturbed experiment can give the basis for estimation of a new model consecutively through the iterations. Whether it is preferred to update the model estimate in each iteration or use the same a priori estimate in the optimization will depend on how well the updated models can be expected to be. Since the controller is changing through the iterations, it would be expected that better models can be achieved by consecutive update. This has to be viewed in relation to the fact that the data from the experiments is optimized for the controller tuning algorithm which may not provide the best data for model estimation.

Having an approximate model estimate provides the option of ensuring nominal stability of the loop before implementing the updated controller, C_{i+1}. Estimation of the plant model can be performed using closed loop system identification (Söderström and Stoica, 1989; Ljung, 1999).

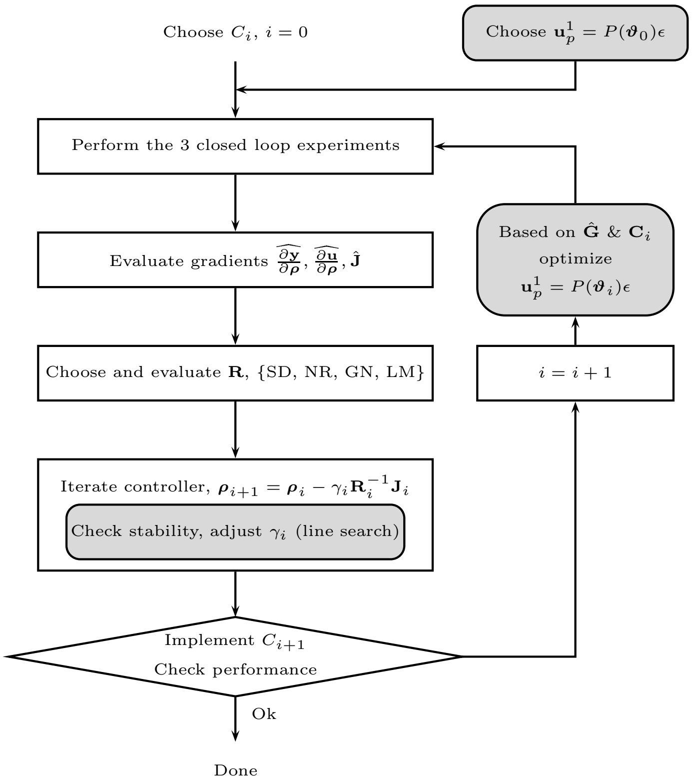


Figure 6.3. Workflow in the novel Perturbed Iterative Feedback Tuning. The new elements are shown with a gray shaded background. The parameters, $\boldsymbol{\vartheta}_i$, are the filter coefficients.

6.5.1 Control parameter update

The control parameter update in the Iterative Feedback Tuning method influences the convergence properties and hence the number of required experiments. In Hjalmarsson *et al.* (1994b) equation (6.17) was suggested as an estimate of the Hessian of the cost function with respect to the control parameters, that can be estimated from the experiments. This estimate is biased due to occurrence of squared terms of the noise signals $\mathbf{v}^2$ and $\mathbf{v}^3$. This problem was resolved in Solari and Gevers (2004) by conducting experiment # 2 and 3 twice in order to form two uncorrelated estimates of $\partial \mathbf{y}/\partial \boldsymbol{\rho}$ and $\partial \mathbf{u}/\partial \boldsymbol{\rho}$ to be used in (6.17). Including yet an extra experiment gives the option of including second order terms in the Hessian estimate as well. The advantage of having an unbiased Hessian estimate, or including second order terms has to be weighted against the disadvantage of an additional experiment in each iteration and the possible loss of the positive definiteness property.

$$\hat{\mathbf{H}} = \frac{1}{N}\sum_{t=1}^{N}\left[\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\right)^T + \lambda \widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\right)^T\right] \tag{6.17}$$

The Gauss-Newton optimization method is known to perform very well in the vicinity of the optimal solution Nocedal and Wright (1999). When the initial controller gives a performance which is far from optimal, the curvature of the performance cost with respect to the control parameters may be more complicated, hence a more cautious algorithm may be preferred. An obvious solution could be to include a damping factor, μ, as a regularization in the Hessian estimate as suggested by Levenberg (Levenberg, 1944).

$$\mathbf{R}_i = \hat{\mathbf{H}}_i + \mu_i \mathbf{I} \tag{6.18}$$

A starting value for the damping coefficient is

$$\mu_0 = \tau \max\left(\mathrm{diag}(\hat{\mathbf{H}}_0)\right) \tag{6.19}$$

where τ is 10^{-6} for a good initial guess and 10^{-3} to 1 if the guess is expected to be poor Madsen *et al.* (2004). The update of the damping coefficient can then be evaluated based on the quality of the previous step. The gain ratio, ϱ, is the ratio between the actual and the expected improvement in the cost function:

$$\varrho = \frac{F_{i-1} - F_i}{\frac{1}{2}\mathbf{h}_i^T(\mu_i \mathbf{h}_i - \mathbf{J}_i)} \tag{6.20}$$

where $\mathbf{h}_i$ is the step in the control parameters given by $-\gamma_i \mathbf{R}_i^{-1}\mathbf{J}(\boldsymbol{\rho}_i)$. As in update of the step length in trust region methods the damping coefficient can be updated using the gain ratio which has given name to Levenberg-Marquardt optimization. The update strategy suggested by Marquardt (Marquardt, 1963) is

$$\mu_{i+1} = \begin{cases} 2\mu_i, & \varrho \in]-\infty; 0.25] \\ \mu_i, & \varrho \in]0.25; 0.75[\\ \mu_i/3, & \varrho \in [0.75; \infty[\end{cases} \tag{6.21}$$

In Nielsen (1999) the damping coefficient is updated by

$$\begin{aligned} \varrho < 0 &\begin{cases} \mu_{i+1} &= \mu_i \nu_i \\ \nu_{i+1} &= 2\nu_i \end{cases} \\ \varrho \geq 0 &\begin{cases} \mu_{i+1} &= \mu_i \max\left(\frac{1}{3}, 1 - (2\varrho - 1)^3\right) \\ \nu_{i+1} &= 2 \end{cases} \end{aligned} \tag{6.22}$$

where $\nu_0 = 2$. This scheme is a continuous version of the strategy suggested by Marquardt (1963) but converges generally faster. The two updating strategies are illustrated in Figure 6.4. Both strategies decrease the step length in the parameter update by increasing μ if the value of the cost function is increasing from one step to another or not sufficiently decreased. The use of Levenberg-Marquardt optimization in Iterative Feedback Tuning is attractive since it provides a systematic method for handling ill conditioned Hessians, which otherwise can lead to large steps in Gauss-Newton optimization that may render the loop unstable. This problem was encountered in Lequin *et al.* (1999) for optimization of step responses. The solution chosen by the authors was to gradually truncate the initial part of the time horizon in the calculation of the cost function and thereby changing the curvature of the performance cost with respect to the controller parameters. The cost function used in Lequin *et al.* (1999) was

$$F(\boldsymbol{\rho}) = \frac{1}{2N}\mathrm{E}\left[\sum_{t=t_0}^{N} \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=1}^{N} u_t(\boldsymbol{\rho})^2\right] \tag{6.23}$$

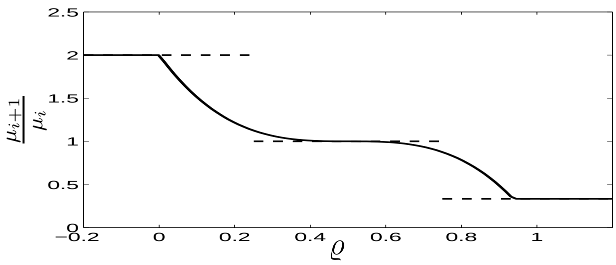


Figure 6.4. Updating of the damping coefficient as function of the gain ratio. The straight dashed curves is the strategy proposed in Marquardt (1963) cf. equation (6.21) and the full line is the strategy proposed by Nielsen (1999) cf. equation (6.22).

where an initial time for the output deviation in the cost function, t_0, was decreased from some initial large value through the iterations. This strategy was effective but the problem can be overcome by using the Levenberg-Marquardt method which optimize the cost function with $t_0 = 1$ through all the iterations.

The more cautious Levenberg-Marquardt method compared to classical Gauss-Newton is advantageous when a model of the system is not available. In case a process model is available, a more attractive update method will include a line search algorithm. Line search is only a real option for Iterative Feedback Tuning if a model is available, since several cost function evaluations are required, which otherwise would demand plant experiments.

6.6 Simulation Examples

Test cases with Iterative Feedback Tuning are performed given the following two degree of freedom PID controller

$$C_r : \frac{U(s)}{R(s)} = K_c \left[1 + \frac{1}{\tau_I s} \right] \tag{6.24a}$$

$$C_y : \frac{U(s)}{Y(s)} = K_c \left[1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{0.1\tau_D s + 1} \right] \tag{6.24b}$$

working in closed loop on the linear time invariant second order process model, (6.25a), affected by Gaussian white noise $v_t \in \mathcal{N}(0, 0.05^2)$ filtered through the first order noise model (6.25b). See Figure 6.1.

$$G(s) = \frac{1}{s^2 + 0.1s + 1} \tag{6.25a}$$

$$H(s) = \frac{1}{s + 1} \tag{6.25b}$$

This process model was used in Lequin *et al.* (1999) to illustrate Iterative Feedback Tuning for the settling time problem. In the first simulation example the same settling time problem is considered and the use of Levenberg-Marquardt optimization is demonstrated. In the subsequent simulation case, in section 6.6.2, the loop is tuned for noise rejection. It is demonstrated how probing signals and the line search algorithm can improve the convergence of the Iterative Feedback Tuning algorithm. Different noise realizations are used through the iterations but the same set of different realizations are use between different trials of the tuning in order to keep comparable conditions. In Monte Carlo simulations performed for performance evaluation all realizations of the noise are independent.

The initial controller is chosen identical to the example in Lequin *et al.* (1999), which gives a very poorly tuned loop, but helps to illustrate some of the inherent problems in Iterative Feedback Tuning.

$$\begin{bmatrix} K_c & \tau_i & \tau_D \end{bmatrix} = \begin{bmatrix} 0.025 & 2 & 1 \end{bmatrix}$$

6.6.1 Optimizing of Settling Time

For the settling time problem a unit step change is introduced in the reference and it is desired to optimize the controller such that the closed loop response resembles that of a first order process with a settling time of 20 seconds, hence $T_d = {}^{3}\!/_{(20s+3)}$. The simulation time is 200 seconds. In optimization of settling times, a cost function without penalty on the control is used, hence $\lambda = 0$ in (6.5). When the classical Iterative Feedback Tuning method is used with the simple Gauss-Newton optimization, the biased Hessian estimate (6.17) and $\boldsymbol{\gamma} = \mathbf{1}$, then the first iteration produces a controller which renders the loop unstable.

As a solution to this problem, and in order to avoid local minima, Lequin *et al.* (1999) uses the cost function (6.23) with an initial time for the output deviation in the cost function on $t_0 = 80$ sec. in the first iteration. This initial time is lowered by 20 sec. until $t_0 = 20$. The values of γ_i remains equal to one. Simulation results based on this strategy but with a final mask of $t_0 = 1$ are presented in Table 6.1, where the control parameters are presented with the corresponding value of the cost function from the first experiment in the iteration and the corresponding mask width.

Iteration	K_c	τ_I	τ_D	Mask	$F(\boldsymbol{\rho}_i) \cdot 10^3$
Initial	0.025	2	1	80	21.713
No. 1	0.0382	1.5344	0.4247	60	7.0074
No. 2	0.0514	1.1304	0.2513	40	4.2786
No. 3	0.0516	0.4671	0.2909	20	2.0616
No. 4	0.0422	0.3742	0.8757	1	1.0989
No. 5	0.0312	0.2599	1.5292	1	1.1534

Table 6.1. Control parameters and the value of the performance cost function for each iteration with the corresponding mask as suggested by Lequin *et al.* (1999). The cost function is evaluated based on the first experiment in the iterations. The Gauss-Newton method is used for optimization of the loop performance

In this paper the above problem is solved using Levenberg-Marquardt optimization, with $\tau = 10^{-4}$ in (6.19) and the update strategy for μ is based on (6.22). The results are presented in Table 6.2. Comparing the results in Tables 6.1 and 6.2, it is seen that the performance after 5 iterations is very close despite the different

Iteration	K_c	τ_I	τ_D	Mask	$F(\boldsymbol{\rho}_i) \cdot 10^3$
Initial	0.025	2	1	1	75.717
No. 1	0.0501	1.9268	0.8011	1	27.655
No. 2	0.0944	1.9029	0.7259	1	8.1042
No. 3	0.1555	1.8930	0.7113	1	2.3160
No. 4	0.1823	1.8904	0.7111	1	1.3894
No. 5	0.2113	1.8862	0.7110	1	0.9797

Table 6.2. Control parameters and the value of the performance cost function for each iteration. The mask is only one sample for all iterations. The cost function is evaluated based on the first experiment in the iterations. The Levenberg-Marquardt method is used for optimization of the loop performance

development of the control parameters through the iterations. The main difference between two methods is, that the method of Lequin *et al.* (1999) changes the cost function that is minimized, when the initial time, t_0, in the cost function is changed. With the proposed Levenberg-Marquardt method the cost function remains the same. That is reflected in the performance cost. In the latter case the improvement from one iteration to the next is reflected by the value of the cost while these are not comparable when the mask t_0 in the cost function is changed.

Remark:
From iteration 4 and 5 in Table 6.1 the value of the performance cost is increased slightly for the same cost function. This is due to the different stochastic realizations of the noise. This behavior is an indication of being close to optimal tuning.

6.6.2 Perturbations in Iterative Feedback Tuning

In this example the process is tuned for disturbance rejection, hence $r_t = 0$ and only the noise present in the first experiment drives the tuning. $\lambda = 0.01$ is used in the cost function. Again the Levenberg-Marquardt optimization is used where $\tau = 10^{-4}$ in (6.19) and the update strategy for μ is based on (6.22). Three trials with different strategies for the tuning are performed for the performance optimization of the process. 10 iterations are used in each of the trials. In the first trial the Iterative Feedback Tuning method is applied in its standard form but having the Levenberg-Marquardt parameter update. In the second trial external perturbations on the control signal are included in the first experiment of each iteration in order to increase the information content in data. The perturbation signal is given by

$$u_p = P(\boldsymbol{\vartheta})\epsilon, \quad \epsilon_t \in \mathcal{N}(0,1), \quad t \in \{1, 2, .., N\} \tag{6.26}$$

where $P(\boldsymbol{\vartheta})$ is the stable first order data filter

$$P(\boldsymbol{\vartheta}) = \frac{\vartheta_1}{\vartheta_2 s + 1}, \quad \boldsymbol{\vartheta} \in \mathbb{R}_+^2 \tag{6.27}$$

The optimization performed in each iteration is based on an estimated plant model and subject to the constraint $F^{max} = 0.02$. In the last trial the same perturbation strategy is applied as in trial 2 and the Levenberg-Marquardt parameter update is extended with an exact line search in each iteration for evaluating γ_i. The line search is performed on the estimated plant and noise model of the system. The results are presented in Figure 6.5 which shows the value of the performance cost function for each of the control loops through the ten iterations as an average of 100 Monte Carlo runs, $\bar{F}_{MC}(\boldsymbol{\rho}_i)$. Tables 6.3 and 6.4 shows the result as control parameters and observed performance for the second and third trial for all ten iterations. This information is omitted for the first trial since no significant changes occur. The process model estimate and the noise model is produced prior to the tuning from closed loop data with the initial controller in the loop. Data points have been collected from one hour simulation with a pseudo-random binary reference signal. This signal was generated with a low pass frequency band from 0 to 0.01 hence a clock period of 100. The amplitude was 0.4 such that the constraint on F^{max} was not violated during the experiment. Two thirds of the data was used to estimate a Box Jenkins model with the true model structure as an open loop estimation problem using the prediction error method, i.e. direct identification. The remaining one third of the data was used for validation and showed white residual for the auto- and cross-correlation functions and a model fit for the one step ahead prediction of 93.7 %.

Iteration	K_c	τ_I	τ_D	$\bar{F}_{MC}(\boldsymbol{\rho}_i) \cdot 10^3$
Initial	0.025	2	1	0.2328
No. 1	0.0180	2.6910	2.4212	0.2319
No. 2	0.0202	3.0130	3.0618	0.2300
No. 3	0.0259	3.1818	3.2934	0.2254
No. 4	0.0349	3.2657	3.4148	0.2191
No. 5	0.0488	3.3067	3.4685	0.2136
No. 6	0.0654	3.3315	3.4918	0.2060
No. 7	0.0935	3.3389	3.4998	0.1956
No. 8	0.1292	3.3397	3.5007	0.1862
No. 9	0.1464	3.3398	3.5011	0.1794
No. 10	0.1577	3.3398	3.5015	0.1761

Table 6.3. Controller parameters and the value of the performance cost function for each iteration in the second series of Perturbed Iterative Feedback Tuning. The cost function is evaluated based on 100 Monte Carlo runs, $\bar{F}_{MC}(\boldsymbol{\rho}_i)$.

From the results of the three trials it can be seen that hardly any improvement of the performance can be observed over the 10 iterations of the tuning for the first trial with classical Iterative Feedback Tuning. The control parameters moved very little in each iteration. In the second trial with Perturbed Iterative Feedback Tuning the performance is improved from one iteration to the other and provides a controller after ten iterations that is clearly superior to the initial trial. The rate of approach towards the local minimizer is slowing down through the iterations which is also due to the update strategy of the damping coefficient. In the third trial where both Perturbed Iterative Feedback Tuning and exact line search are used, significant

Iteration	K_c	τ_I	τ_D	$\bar{F}_{MC}(\boldsymbol{\rho}_i) \cdot 10^3$
Initial	0.025	2	1	0.2328
No. 1	0.0221	2.2873	1.5909	0.2313
No. 2	0.0231	5.2606	8.5949	0.2251
No. 3	0.2475	6.1356	2.1208	0.1599
No. 4	11.438	7.7009	0.8945	0.1492
No. 5	11.437	7.7030	0.7431	0.1475
No. 6	11.437	7.7030	0.7431	0.1481
No. 7	11.437	7.7030	0.7431	0.1485
No. 8	11.437	7.7030	0.7427	0.1478
No. 9	11.437	7.7030	0.7421	0.1471
No. 10	11.437	7.7030	0.7421	0.1472

Table 6.4. Controller parameters and the value of the performance cost function for each iteration in the third series of Perturbed Iterative Feedback Tuning with line search. The cost function is evaluated based on 100 Monte Carlo runs, $\bar{F}_{MC}(\boldsymbol{\rho}_i)$.

improvements can be observed between consecutive iterations until the fifth iteration where the update seems to have converged. After the fifth iteration the line search could not find improvement in the search direction for these iterations and did not approach the minimum further. The theoretical value of the local minimizer has been evaluated numerically to $7.636 \cdot 10^{-5}$ based on full process knowledge.

6.7 Conclusion

An extension to the Iterative Feedback Tuning algorithm imposing external probing signals to the predefined experiments is proposed. Perturbed Iterative Feedback Tuning is an advantage when tuning for disturbance rejection. Perturbing the process can yield more informative data and thereby improve convergence properties of the tuning method for disturbance rejection, hence reducing the number of plant experiments. The use of the Perturbed Iterative Feedback Tuning algorithm is outlined. It is motivated to generate this external probing signal from a constraint optimization utilizing a plant model, which is not necessary in the standard formulation of the tuning method. Having a plant and a noise model of the system renders the use of a line search algorithm for the parameter update possible, which is demonstrated to significantly improve convergence. Furthermore availability of a model allows a check on nominal stability of the loop. The use of Levenberg-Marquardt optimization is advocated and illustrated for controller tuning of a step response problem. The advantages of the proposed algorithm with probing and line search is illustrated on a disturbance rejection problem, which is notoriously difficult for classical Iterative Feedback Tuning.

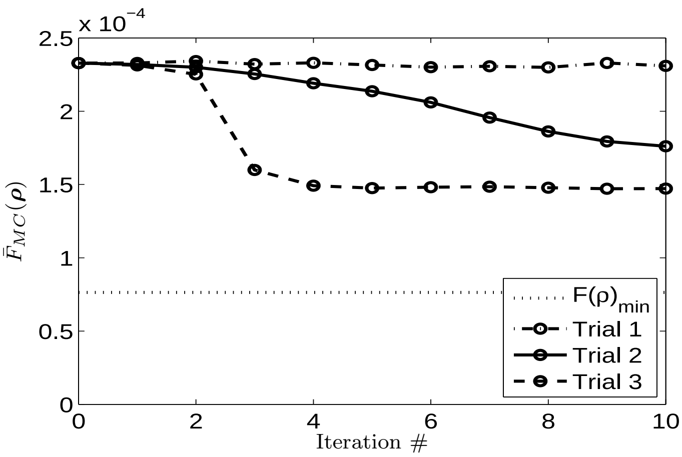


Figure 6.5. Performance of each controller through the iterations for all three series evaluated based on 100 Monte Carlo trials. The cost for the local minimizer is also displayed as the dotted line.

7

A Design Algorithm using External Perturbation to Improve Iterative Feedback Tuning Convergence

Abstract

Iterative Feedback Tuning constitutes an attractive control loop tuning method for processes in the absence of sufficient process insight. It is a purely data driven approach to optimization of the loop performance. The standard formulation ensures an unbiased estimate of the loop performance cost function gradient, which is used in a search algorithm for minimizing the performance cost.

A slow rate of convergence of the tuning method is often experienced when tuning for disturbance rejection. This is due to a poor signal to noise ratio in the process data. A method is proposed for increasing the data information content by introducing an optimal perturbation signal in the tuning algorithm. The perturbation signal design is based on a detailed analysis of the asymptotic accuracy of the tuning method. A formal algorithm for optimization of the perturbation signal spectrum when tuning for disturbance rejection is presented. Special cases where an explicit optimal design are available is discussed. The theoretical analysis is supported by a simulation example.

7.1 Introduction

Control design and tuning for disturbance rejection is one of the classical disciplines in control theory and control engineering science. Design of compensators for disturbance rejections is well documented (Åström, 1970; Box and Jenkins, 1970; Åström and Hägglund, 1995). Given a particular control design, the tuning of the control parameters can be conducted based on tuning rules or by minimization of some loop performance criterion. The performance criterion is typically a quadratic cost function with penalty on the process outputs and the control signals. Given a model of the system, the set of optimal control parameters which minimize the performance cost can be evaluated. In absence of a sufficiently reliable model, the tuning can be performed based on data obtained from the loop, by a data driven optimization. Iterative Feedback Tuning is a method for optimizing control parameters using closed loop data and this algorithm will form the basis for the modifications presented here. The basic algorithm was first presented in Hjalmarsson *et al.* (1994b) and has since then been analyzed, extended and tested in a number of papers. Gevers (2002) and Hjalmarsson (2002) provide extensive overviews of the development of the method and references to applications.

The performance criterion, $F_N(y_t, u_t)$, used in the controller tuning is a function of the output and the control action for the control loop. Hence it is a function of the true system, the controller and external signals acting on the loop. We will use the set-up in Figure 7.1 where G is a causal scalar linear time-invariant system, C is the controller, which also is assumed to be causal scalar linear time-invariant, and where r_t is the reference signal and v_t is the disturbance, respectively. Assuming, as we will, that the disturbance is stochastic implies that the performance cost is itself a random variable. However, as in, e.g., LQG-control, it is natural to minimize the expected cost

$$F(\cdot) \triangleq \mathrm{E}\left[F_N(\cdot)\right] \tag{7.1}$$

where here $\mathrm{E}[\cdot]$ is the mathematical expectation over the random disturbances acting on the closed loop system. This notation will be used throughout this paper. Notice that in the following, when expectation of $F(\cdot)$ is taken, the expectation refers not to the random disturbances acting on the system when assessing the closed loop peformance. It refers to the random variables that have affected the experimental data that has been used to design the controller for which the performance of $F(\cdot)$ is to be assessed. In order words the expectation will be taken over the controller C which will be seen as a random variable.

Our objective is to design a controller such that F is minimized when $r_t \equiv 0$, i.e. we are interested in disturbance rejection. Adding a reference signal during the experimentation phase may however improve the quality of the obtained controller C. In Iterative Feedback Tuning, one tries to minimize F with respect to the controller using noisy closed loop experiments. The accuracy of this very much depends on the shape of the cost function F one tries to minimize. The sharper the optimum of F is, the easier it will be to find a good controller. Now, any change in the spectrum Φ_r of the reference signal, will affect the output spectrum Φ_y and the input spectrum Φ_u. Hence the reference signal spectrum affects the minimum and the shape of the performance cost surface. By designing the spectrum of an external reference it is consequently possible to shape the performance cost function

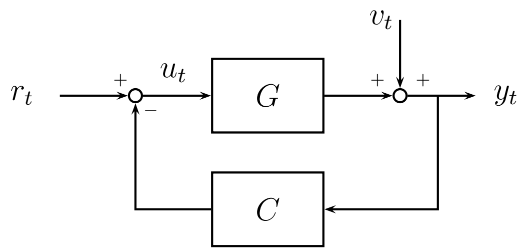


Figure 7.1. A general feedback loop designed for disturbance rejection. The process, G, and the compensator in the feedback loop, C, is given as scalar linear transfer functions.

in order to improve the convergence properties of the search algorithm in the tuning method for the control parameters. However, one has to bear in mind that shaping the cost function will also influence the location of the minimum in the controller parameter space. The cost function evaluated with external perturbation will be different from that of the original design problem when tuning for disturbance rejection. This is illustrated in Figure 7.2 where two examples of a quadratic cost function are shown as function of two control parameters. Let the original design F_0 refer to the disturbance rejection case where the reference signal to the loop is zero. F_1 is then the evaluation of the same cost function for the case with external perturbation where $\Phi_r \neq 0$. Since the contour lines of F_1 are closer together than for F_0, the optimization with the perturbation is less sensitive to the stochastic element in the evaluation of the performance cost. The price to be paid is that the method converges towards a different minimum. Despite this unfortunate consequence, successful simulation studies are reported with respect to convergence using Perturbed Iterative Feedback Tuning when tuning for disturbance rejection (Huusom *et al.*, 2008).

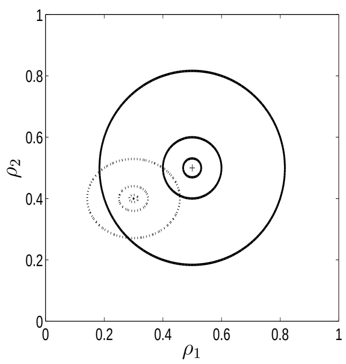


Figure 7.2. Contours and minima for two cost functions with equal levels for the contour lines. ρ_1 and ρ_2 are the control parameters. The full lines and the cross refer to the original design criterion, F_0. The dotted lines and the dot in the center is the cost function when affected by an external perturbation, F_1.

7.1.1 Formulating a design criterion

Let $F(\boldsymbol{\rho}, \boldsymbol{\vartheta})$ denote the cost function that we are interested in minimizing, where $\boldsymbol{\rho}$ and $\boldsymbol{\vartheta}$ represent the free control parameters which are to be tuned and a set of parameters which characterize the reference signal spectrum, respectively. The objective is to find the optimal $\boldsymbol{\rho}$ for a given $\boldsymbol{\vartheta} = \boldsymbol{\vartheta}^0$ which corresponds to $r_t \equiv 0$. We denote the optimum $\boldsymbol{\rho}$ by $\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta})$, indicating its dependence on $\boldsymbol{\vartheta}$. Since the system will be affected by noise it is only possible to obtain a minimizer, $\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta})$, with a certain accuracy; we use subscript n to denote that n iterations are performed in the tuning method. Hence Iterative Feedback Tuning will produce a solution with the following error

$$\boldsymbol{\Sigma}_n(\boldsymbol{\vartheta}) \triangleq \mathrm{E}\left[\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta})\right)\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta})\right)^T\right] \tag{7.2}$$

that has the property that it depends on $\boldsymbol{\vartheta}$. Using a continuity argument it may therefore be advantageous to optimize $\boldsymbol{\rho}$ for a $\boldsymbol{\vartheta} \neq \boldsymbol{\vartheta}^0$, i.e. it may be that the controller corresponding to $\boldsymbol{\vartheta}$ may result in a smaller expected cost for the desired excitation conditions (which correspond to $\boldsymbol{\vartheta}^0$) than the controller tuned with the desired operating conditions $\boldsymbol{\vartheta}^0$. This can be expressed as that it may hold that

$$\mathrm{E}\left[F(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}), \boldsymbol{\vartheta}^0)\right] < \mathrm{E}\left[F(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0)\right] \tag{7.3}$$

Our objective is to determine operating conditions $\boldsymbol{\vartheta}$ such that $\mathrm{E}\left[F(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}), \boldsymbol{\vartheta}^0)\right]$ is minimized. This is a very difficult problem since $F(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}), \boldsymbol{\vartheta}^0)$ is a very complicated and non-linear function of the random disturbances originating from the experiments on which $\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta})$ is based. This in turn means that the expectation with respect to these random variables is very difficult to compute. Our approach to cope with this is to perform a local analysis, assuming $\boldsymbol{\vartheta}$ to be close to $\boldsymbol{\vartheta}^0$. Using Taylor expansion near the optimum we have that

$$\begin{aligned}
F(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}),\boldsymbol{\vartheta}^0) \approx{}& F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0) + \frac{\partial F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}}\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right) + \\
& \frac{1}{2}\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)^T \frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right) \\
={}& F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0) + \frac{1}{2}\mathrm{Tr}\left\{\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)^T\right\}
\end{aligned} \tag{7.4}$$

which means that

$$\begin{aligned}
\mathrm{E}&\left[F(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}), \boldsymbol{\vartheta}^0)\right] - F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0) \\
&\approx \frac{1}{2}\mathrm{Tr}\left\{\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}\mathrm{E}\left[\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)\left(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)^T\right]\right\} \\
&= \frac{1}{2}\mathrm{Tr}\left\{\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}\left(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)\left(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)^T\right\} + \\
&\quad \frac{1}{2}\mathrm{Tr}\left\{\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0), \boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}\boldsymbol{\Sigma}_n(\boldsymbol{\vartheta})\right\} \triangleq \Delta F_n(\boldsymbol{\vartheta})
\end{aligned} \tag{7.5}$$

Now, if $\mathbf{\Sigma}_n(\boldsymbol{\vartheta})$ can be evaluated then $\Delta F_n(\boldsymbol{\vartheta})$ is a quantity that can be minimized with respect to $\boldsymbol{\vartheta}$ in order to find the (approximately) optimal (reference) perturbation signal spectrum to be used in the experiments when tuning the controller parameters $\boldsymbol{\rho}$ using Iterative Feedback Tuning.

The two terms in $\Delta F_n(\boldsymbol{\vartheta})$ can be interpreted as follows: The first term is the bias error due to that $\boldsymbol{\vartheta} \neq \boldsymbol{\vartheta}^0$ is used in the optimization whereas the second term is the variance error incurred on $F(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}), \boldsymbol{\vartheta}^0)$. The bias error will typically increase as $\boldsymbol{\vartheta}$ moves away from $\boldsymbol{\vartheta}^0$. As noted above, it may be possible to decrease the variance error if $\boldsymbol{\vartheta}$ is suitably chosen. The optimal perturbation choice $\boldsymbol{\vartheta} = \bar{\boldsymbol{\vartheta}}$ will balance these two terms. The aim of this study is to construct a systematic and formal algorithm for designing an optimal external perturbation signal for Iterative Feedback Tuning of the disturbance rejection problem. Based on (7.5), this algorithm will minimize a design criterion which explicitly addresses this trade off between bias and variance error in the distribution of the n'th iterate in the tuning algorithm, $\boldsymbol{\rho}_n$.

The paper is organized as follows: Section 7.2 presents the basic Iterative Feedback Tuning algorithm for disturbance rejection. We also review an expression for the error $\mathbf{\Sigma}_n(\boldsymbol{\vartheta})$ of the method derived in Hildebrand *et al.* (2005b) for the disturbance rejection problem. In Section 7.3 the effect of adding an external perturbation signal to the loop in the tuning method is analyzed. This extends the result in Hildebrand *et al.* (2005b) and provides us with an expression for $\mathbf{\Sigma}_n(\boldsymbol{\vartheta})$ required for the computation of the expression on the right in (7.5). Then in Section 7.4, a formal design criterion for the perturbation spectrum is derived and a full algorithm, tuning for disturbance rejection with Perturbed Iterative Feedback Tuning using process insight, is constructed. Finally a simulation example serves to illustrate the advantages of introducing an optimal external perturbation signal in the tuning algorithm for the disturbance rejection case. Derivation of covariance expressions for the derivative of the performance cost function is given in an appendix.

7.2 Iterative Feedback Tuning for disturbance rejection

The algorithm for performing Iterative Feedback Tuning for disturbance rejection is illustrated in the following. The feedback loop in Figure 7.1 depicts the signals and transfer functions which will be used in the algorithm for tuning the parameters $\boldsymbol{\rho}$ in C. The objetive is to tuning the controller such that the effect of the noise, v_t, is rejected in an optimal sense.

The objective is to minimize the cost function:

$$F_N(\boldsymbol{\rho}_i) = \frac{1}{2N}\sum_{t=1}^{N}(y_t(\boldsymbol{\rho}_i) - y_t^d)^2 + \lambda(u_t(\boldsymbol{\rho}_i))^2 \tag{7.6}$$

where N number of data points in the discrete time horizon and y^d is the desired output response. For the disturbance rejection problem $r_t \equiv 0$ and hence $y_t^d = 0$.

The sensitivity of the cost function with respect to the control parameters is

$$\boldsymbol{J}(\boldsymbol{\rho}_i) = \frac{\partial F_N(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} = \frac{1}{N}\sum_{t=1}^{N} y_t(\boldsymbol{\rho}_i)\frac{\partial y_t(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} + \lambda u_t(\boldsymbol{\rho}_i)\frac{\partial u_t(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \tag{7.7}$$

where

$$\frac{\partial y_t}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}) = -\frac{\partial C(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}} G S^2(\boldsymbol{\rho}) v_t \tag{7.8}$$

$$\frac{\partial u_t}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}) = -\frac{\partial C(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}} S^2(\boldsymbol{\rho}) v_t \tag{7.9}$$

The minimization of the cost function is realized by iterating in the gradient scheme

$$\boldsymbol{\rho}_{i+1} = \boldsymbol{\rho}_i - \gamma_i \mathbf{R}^{-1}\boldsymbol{J}(\boldsymbol{\rho}_i) \tag{7.10}$$

where $\mathbf{R}$ is a positive definite matrix. It could be chosen as the Hessian of the cost function with respect to the control parameters $\boldsymbol{\rho}$, or the identity matrix to achieve a Newton or a steepest decent algorithm respectively. If a model for the system is unknown, the gradients of the in- and output and hence the cost function gradient can not be evaluated analytically. An estimate of the performance cost function gradient is

$$\widehat{\boldsymbol{J}(\boldsymbol{\rho}_i)} = \frac{1}{N}\sum_{t=1}^{N} y_t(\boldsymbol{\rho}_i)\frac{\widehat{\partial y_t(\boldsymbol{\rho}_i)}}{\partial \boldsymbol{\rho}} + \lambda u_t(\boldsymbol{\rho}_i)\frac{\widehat{\partial u_t(\boldsymbol{\rho}_i)}}{\partial \boldsymbol{\rho}} \tag{7.11}$$

where $\frac{\widehat{\partial y_t(\boldsymbol{\rho}_i)}}{\partial \boldsymbol{\rho}}$ and $\frac{\widehat{\partial u_t(\boldsymbol{\rho}_i)}}{\partial \boldsymbol{\rho}}$ are estimates of (7.8) and (7.9) respectively. In the traditional Iterative Feedback Tuning framework the minimization of the cost function, (7.6), is based on data from two successive experiments (Hjalmarsson *et al.*, 1998).

- Collect data $\{y_t^1(\boldsymbol{\rho}_i), u_t^1(\boldsymbol{\rho}_i)\}_{t=1,..,N}$ where $r_t^1 = 0$
- Collect data $\{y_t^2(\boldsymbol{\rho}_i), u_t^2(\boldsymbol{\rho}_i)\}_{t=1,..,N}$ where $r_t^2 = -y_t^1$

This data is used to estimate the gradients of the in- and outputs

$$\frac{\widehat{\partial y_t}}{\partial \boldsymbol{\rho}} \triangleq \frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} y_t^2 \tag{7.12}$$

$$= \frac{\partial y_t}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}_i) + \frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}_i) v_t^2 \tag{7.13}$$

$$\frac{\widehat{\partial u_t}}{\partial \boldsymbol{\rho}} \triangleq \frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} u_t^2 \tag{7.14}$$

$$= \frac{\partial u_t}{\partial \boldsymbol{\rho}}(\boldsymbol{\rho}_i) - \frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}_i) C(\boldsymbol{\rho}_i) v_t^2 \tag{7.15}$$

where (7.12) and (7.14), are the estimators for the gradients of the in- and outputs. When these two expressions are used to form the estimate for the performance cost function gradient (7.11), (7.13) and (7.15) imply that the estimate can be split into two terms: An analytic bias term, S_N, and a variance term, E_N. The latter term is due to the noise present in the second experiment.

$$\widehat{\boldsymbol{J}(\boldsymbol{\rho}_i)} = S_N(\boldsymbol{\rho}_i) + E_N(\boldsymbol{\rho}_i) \tag{7.16}$$

where

$$\begin{aligned} S_N(\boldsymbol{\rho}) =& \frac{1}{N}\sum_{t=1}^{N}\left[y_t^1(\boldsymbol{\rho})\frac{\partial y_t(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}} + \lambda u_t^1(\boldsymbol{\rho})\frac{\partial u_t(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}\right] \\ =& \frac{1}{N}\sum_{t=1}^{N}\left[(S(\boldsymbol{\rho})v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}GS(\boldsymbol{\rho})^2v_t^1\right) + \lambda(-S(\boldsymbol{\rho})C(\boldsymbol{\rho})v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})^2v_t^1\right)\right] \end{aligned} \tag{7.17}$$

$$\begin{aligned} E_N(\boldsymbol{\rho}) =& \frac{1}{N}\sum_{t=1}^{N}\left[y_t^1(\boldsymbol{\rho})\left(\widehat{\frac{\partial y_t(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}} - \frac{\partial y_t(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}\right) + \lambda u_t^1(\boldsymbol{\rho})\left(\widehat{\frac{\partial u_t(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}} - \frac{\partial u_t(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}\right)\right] \\ =& \frac{1}{N}\sum_{t=1}^{N}\left[(S(\boldsymbol{\rho})v_t^1)\left(\frac{\partial C(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})v_t^2\right) + \lambda(-S(\boldsymbol{\rho})C(\boldsymbol{\rho})v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho})C(\boldsymbol{\rho})v_t^2\right)\right] \end{aligned} \tag{7.18}$$

The expectation of the variance part is zero, since the noise signals from the first and second experiment are independent. The estimate of the cost function gradient produced by the Iterative Feedback Tuning method is therefore an unbiased realization.

Given that the noise v is a zero mean, weakly stationary random signal, the key contribution in Iterative Feedback Tuning, is that it supplies an unbiased estimate of the cost function gradient, without requiring a plant model estimate, $\hat{G}$, (Hjalmarsson *et al.*, 1998). Let the estimate, (7.11), be an unbiased and monotonically increasing function of $\boldsymbol{\rho}$. Using the estimate (7.11) in the gradient iteration (7.10) instead of the analytical expression (7.7), as a stochastic approximation method, will still make the algorithm converge to the expectation of the local minimizer provided that the sequence of γ_i in (7.10) fulfills (Robbins and Monro, 1951; Hildebrand *et al.*, 2003)

$$\sum_{i=1}^{\infty}\gamma_i^2 < \infty, \quad \sum_{i=1}^{\infty}\gamma_i = \infty. \tag{7.19}$$

This condition is fulfilled e.g. by having $\gamma_i = {}^{a}\!/\!_{i}$ where a is some positive constant.

A Gauss-Newton approximation of the Hessian to the performance cost function with respect to the controller parameters was suggested in Hjalmarsson *et al.* (1994b). This first order approximation can be estimated using the available signals from the tuning method

$$\hat{\mathbf{H}} = \frac{1}{N}\sum_{t=1}^{N}\left[\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\right)^T + \lambda\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\right)^T\right] \tag{7.20}$$

This estimate will not be unbiased due to squared terms of the noise in the two experiments, but it will be positive definite. A modification (Solari and Gevers, 2004) that involves additional experiments in each iteration of the iterative Feedback Tuning algorithm produces an unbiased Hessian estimate.

7.2.1 Asymptotic accuracy of the tuning method

The stochastic contribution in the gradient estimate will affect the asymptotic convergence rate of the tuning method. A quantitative analysis was performed by Hildebrand *et al.* (2005a). The result is as follows: With n being the iteration number and $\bar{\boldsymbol{\rho}}$ the optimal set of parameters, the sequence of random variables, $\sqrt{n}(\boldsymbol{\rho}_n - \bar{\boldsymbol{\rho}})$, converge *in distribution* to a normally distributed random variable with zero mean and covariance matrix $\boldsymbol{\Sigma}$ according to

$$\begin{aligned} &\sqrt{n}(\boldsymbol{\rho}_n - \bar{\boldsymbol{\rho}}) \xrightarrow{D} \mathcal{N}(0, \boldsymbol{\Sigma}) \\ &\boldsymbol{\Sigma} = a^2 \int_0^\infty e^{\boldsymbol{A}t} \boldsymbol{R}^{-1} \mathrm{Cov}\left[\widehat{\boldsymbol{J}(\bar{\boldsymbol{\rho}})}\right] \boldsymbol{R}^{-1} e^{\boldsymbol{A}^T t} dt \end{aligned} \tag{7.21}$$

The result in (7.21) is valid given the following set of conditions hold:

1. The sequence $\boldsymbol{\rho}_n$ converges to a local isolated minimum $\bar{\boldsymbol{\rho}}$ of F
2. $\boldsymbol{H}(\bar{\boldsymbol{\rho}})$ is the true Hessian for $F(\boldsymbol{\rho})$ at $\bar{\boldsymbol{\rho}}$.
3. The gain sequence $\{\boldsymbol{\gamma}_n\}$ in (7.10) is given by $\gamma_n = {}^{a}\!/_{n}$, where a is a positive constant.
4. There exists an index $\bar{n}$ and a matrix $\boldsymbol{R}$ such that $\boldsymbol{R}_n = \boldsymbol{R}$ for all $n > \bar{n}$.
5. The matrix $\boldsymbol{A} = {}^{1}\!/_{2}\boldsymbol{I} - a\boldsymbol{R}^{-1}\boldsymbol{H}(\bar{\boldsymbol{\rho}})$ is stable, i.e. the real part of all the eigenvalues is negative.
6. The covariance matrix $\mathrm{Cov}\left[\widehat{\boldsymbol{J}(\bar{\boldsymbol{\rho}})}\right]$ is positive definite.

The result in (7.21) means that asymptotically the distribution for the deviation between the n'th iterate of the controller parameter and the true optimum is known, and that the method converges to the true local minimizer of the performance cost function. In Hildebrand *et al.* (2005b) it is shown that the covariance expression for the distribution simplifies if $\boldsymbol{H}(\bar{\boldsymbol{\rho}})$, i.e. the true Hessian, is used as the matrix $\boldsymbol{R}$ in (7.10). Hence for a Newton-Raphson optimization

$$\boldsymbol{\Sigma} = \frac{a^2}{2a-1} \boldsymbol{R}^{-1} \mathrm{Cov}\left[\widehat{\boldsymbol{J}(\bar{\boldsymbol{\rho}})}\right] \boldsymbol{R}^{-1} \tag{7.22}$$

As a measure of the quality of the controller for a given iteration, n, in the tuning algorithm Hildebrand *et al.* (2005b) suggest the difference between the expected value of the performance cost with $C(\boldsymbol{\rho}_n)$ in the loop minus the theoretical minimum value. This quantity, ΔF_n, will be referred to as the *control quality index*.

$$\Delta F_n \triangleq \mathrm{E}[F(\boldsymbol{\rho}_n)] - F(\bar{\boldsymbol{\rho}}) \tag{7.23}$$

This index is by definition a positive measure. Expanding it in a Taylor series around the optimum up to second order gives the approximation:

$$\Delta F_n \approx \frac{1}{2} \mathrm{E}\left[\Delta\bar{\boldsymbol{\rho}}_n^T \boldsymbol{H}(\bar{\boldsymbol{\rho}}) \Delta\bar{\boldsymbol{\rho}}_n\right] \tag{7.24}$$

where $\Delta\bar{\boldsymbol{\rho}}_n = \boldsymbol{\rho}_n - \bar{\boldsymbol{\rho}}$. The following asymptotic expression when $\mathbf{H}(\bar{\boldsymbol{\rho}})\mathbf{R}^{-1} = \mathbf{I}$ is given in Hildebrand *et al.* (2005b):

$$\lim_{n\to\infty} n\mathrm{E}\left[\Delta\bar{\boldsymbol{\rho}}_n^T \boldsymbol{H}(\bar{\boldsymbol{\rho}})\Delta\bar{\boldsymbol{\rho}}_n\right] = \frac{a^2}{2a-1}\mathrm{Tr}\left\{\mathrm{Cov}\left[\widehat{\boldsymbol{J}(\bar{\boldsymbol{\rho}})}\right]\left[\boldsymbol{R}^{-1}\right]\right\} \tag{7.25}$$

From this analysis, it is seen that the covariance of the gradient estimate for the performance cost function influences both the asymptotic covariance of the distribution of $\Delta\bar{\boldsymbol{\rho}}_n$ and the control performance quality measure given the parameters $\boldsymbol{\rho}_n$. It is therefore of interest to decompose this covariance expression. Due to the independence of the signals v_t^1 and v_t^2, the covariance of the gradient estimate in Equation (7.16) can be divided into the following contributions.

$$\begin{aligned}\mathrm{Cov}\left[\widehat{\boldsymbol{J}(\boldsymbol{\rho})}\right] &= \mathrm{Cov}[S_N(\boldsymbol{\rho})] + \mathrm{E}\left[E_N(\boldsymbol{\rho})S_N(\boldsymbol{\rho})^T\right] + \mathrm{E}\left[E_N(\boldsymbol{\rho})S_N(\boldsymbol{\rho})^T\right]^T + \mathrm{Cov}[E_N(\boldsymbol{\rho})] \\ &= \mathrm{Cov}[S_N(\boldsymbol{\rho})] + \mathrm{Cov}[E_N(\boldsymbol{\rho})] \end{aligned} \tag{7.26}$$

Assuming that the disturbance $\{v_t\}$ is a Gaussian process, the asymptotic frequency-domain expressions of the two remaining terms are (Hildebrand *et al.*, 2005a):

$$\begin{aligned}\lim_{N\to\infty} N\mathrm{Cov}[S_N(\boldsymbol{\rho})] =& \frac{2}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho})|^4 \Phi_v^2(\omega)\times \\ & \mathcal{R}e\left\{[G(e^{j\omega},\boldsymbol{\rho}) - \lambda\overline{C}(e^{j\omega},\boldsymbol{\rho})]S(e^{j\omega},\boldsymbol{\rho})\frac{\partial C(e^{j\omega},\boldsymbol{\rho})}{\partial\boldsymbol{\rho}}\right\}\times \\ & \mathcal{R}e\left\{[G(e^{j\omega},\boldsymbol{\rho}) - \lambda\overline{C}(e^{j\omega},\boldsymbol{\rho})]S(e^{j\omega},\boldsymbol{\rho})\frac{\partial C(e^{j\omega},\boldsymbol{\rho})}{\partial\boldsymbol{\rho}}\right\}^T d\omega \end{aligned} \tag{7.27}$$

$$\begin{aligned}\lim_{N\to\infty} N\mathrm{Cov}[E_N(\boldsymbol{\rho})] =& \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho})|^4 \left[1 + \lambda|C(e^{j\omega},\boldsymbol{\rho})|^2\right]^2 \times \\ & \frac{\partial C(e^{j\omega},\boldsymbol{\rho})}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho})}{\partial\boldsymbol{\rho}}\Phi_v^2(\omega)d\omega \end{aligned} \tag{7.28}$$

where $\bar{C}$ is the complex conjugate and C^* is the complex conjugate transpose of C. A derivation is presented in Appendix B.

7.3 Introducing external perturbations in the tuning

It is desired to improve the convergence rate and the asymptotic accuracy of the Iterative Feedback Tuning method. To achive this, the signal to noise ratio in data used in the tuning method must be increased. An external perturbation signal will be used as reference in the first of the two experiments used in the tuning algorithm. The experiments are then defined as follows:

- Collect data $\{y_t^1(\boldsymbol{\rho}_i), u_t^1(\boldsymbol{\rho}_i)\}_{t=1,..,N}$ where $r_t^1 = r_t^p$
- Collect data $\{y_t^2(\boldsymbol{\rho}_i), u_t^2(\boldsymbol{\rho}_i)\}_{t=1,..,N}$ where $r_t^2 = -y_t^1$

where the external input r_t^p is characterized by the spectrum Φ_{r^p}. A discussion on using external perturbations in the Iterative Feedback Tuning algorithm and an introduction to Perturbed Iterative Feedback Tuning are given in Huusom *et al.* (2008). The implication of introducing the external perturbation signal on the convergence properties of the method will be elaborated in the following.

The implication on the gradient estimate of the cost function from including this extra signal is

$$\begin{aligned} S_N(\boldsymbol{\rho}_i) =& \frac{1}{N}\sum_{t=1}^{N}\Bigg[\left(S(\boldsymbol{\rho}_i)(Gr_t^p + v_t^1)\right)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} GS(\boldsymbol{\rho}_i)^2(Gr_t^p + v_t^1)\right) + \\ & \lambda S(\boldsymbol{\rho}_i)(r_t^p - C(\boldsymbol{\rho}_i)v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}_i)^2(Gr_t^p + v_t^1)\right)\Bigg] \end{aligned} \tag{7.29}$$

$$\begin{aligned} E_N(\boldsymbol{\rho}_i) =& \frac{1}{N}\sum_{t=1}^{N}\Bigg[\left(S(\boldsymbol{\rho}_i)(Gr_t^p + v_t^1)\right)\left(\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}_i)v_t^2\right) + \\ & \lambda S(\boldsymbol{\rho}_i)(r_t^p - C(\boldsymbol{\rho}_i)v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} C(\boldsymbol{\rho}_i)S(\boldsymbol{\rho}_i)v_t^2\right)\Bigg] \end{aligned} \tag{7.30}$$

Given the following two complex functions

$$\Psi(e^{j\omega}, \boldsymbol{\rho}) = [G(e^{j\omega}, \boldsymbol{\rho}) - \lambda \overline{C}(e^{j\omega}, \boldsymbol{\rho})]S(e^{j\omega}, \boldsymbol{\rho})\frac{\partial C(e^{j\omega}, \boldsymbol{\rho})}{\partial \boldsymbol{\rho}} \tag{7.31}$$

$$\Upsilon(e^{j\omega}, \boldsymbol{\rho}) = [|G(e^{j\omega}, \boldsymbol{\rho})|^2 + \lambda]S(e^{j\omega}, \boldsymbol{\rho})\frac{\partial C(e^{j\omega}, \boldsymbol{\rho})}{\partial \boldsymbol{\rho}} \tag{7.32}$$

and assuming that the disturbance $\{v_t\}$ and the reference signal r_t are Gaussian processes, the asymptotic covariance expressions for $S_N(\boldsymbol{\rho})$ and $E_N(\boldsymbol{\rho})$ are given as (see Appendix C for details)

$$\begin{aligned} \lim_{N\to\infty} N\mathrm{Cov}[S_N(\boldsymbol{\rho})] =& \frac{1}{2\pi}\int_{-\pi}^{\pi}|S(e^{j\omega}, \boldsymbol{\rho})|^4 \times \Bigg[\\ & \mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}^T\Phi_{r^p}^2 + \mathcal{R}e\{\Psi(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{R}e\{\Psi(e^{j\omega}, \boldsymbol{\rho})\}^T\Phi_v^2 + \\ & \Bigg[2\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{R}e\{\Psi(e^{j\omega}, \boldsymbol{\rho})\}^T + \mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{I}m\{\Psi(e^{j\omega}, \boldsymbol{\rho})\}^T - \\ & \mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}^T\mathcal{I}m\{\Psi(e^{j\omega}, \boldsymbol{\rho})\} + \mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega}, \boldsymbol{\rho})\}^T + \\ & \mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega}, \boldsymbol{\rho})\}^T + \mathcal{R}e\{\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{R}e\{\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}^T + \\ & \mathcal{I}m\{\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}\mathcal{I}m\{\Upsilon(e^{j\omega}, \boldsymbol{\rho})\}^T\Bigg]\Phi_{r^p}\Phi_v\Bigg]d\omega \end{aligned} \tag{7.33}$$

$$\begin{aligned} & \lim_{N\to\infty} N\mathrm{Cov}[E_N(\boldsymbol{\rho})] = \\ & \frac{1}{2\pi}\int_{-\pi}^{\pi}|S(e^{j\omega}, \boldsymbol{\rho})|^4\left[1 + \lambda|C(e^{j\omega}, \boldsymbol{\rho})|^2\right]^2 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho})}{\partial \boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho})}{\partial \boldsymbol{\rho}}\right)^T\Phi_v^2 + \\ & \left[\mathcal{R}e\{\Psi(e^{j\omega}, \boldsymbol{\rho}))\}\mathcal{R}e\{\Psi(e^{j\omega}, \boldsymbol{\rho}))\}^T + \mathcal{I}m\{\Psi(e^{j\omega}, \boldsymbol{\rho}))\}\mathcal{I}m\{\Psi(e^{j\omega}, \boldsymbol{\rho}))\}^T\right] \times \\ & |S(e^{j\omega}, \boldsymbol{\rho})|^4\Phi_{r^p}\Phi_v d\omega \end{aligned} \tag{7.34}$$

In relation to experimental design of the perturbation spectrum it is important to know how the Hessian approximation is affected:

$$\hat{\mathbf{H}} = \frac{1}{N}\sum_{t=1}^{N}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}}\right)^T + \lambda\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\left(\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}}\right)^T\right) \tag{7.35}$$

where

$$\widehat{\frac{\partial y_t}{\partial \boldsymbol{\rho}}} = \frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\left(GS(\boldsymbol{\rho}_i)^2(Gr_t^p + v_t^1) + S(\boldsymbol{\rho}_i)v_t^2\right) \tag{7.36}$$

$$\widehat{\frac{\partial u_t}{\partial \boldsymbol{\rho}}} = \frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\left(S(\boldsymbol{\rho}_i)^2(Gr_t^p + v_t^1) + C(\boldsymbol{\rho}_i)S(\boldsymbol{\rho}_i)v_t^2\right) \tag{7.37}$$

hence

$$\begin{aligned}\hat{\mathbf{H}}(e^{j\omega}) =& \frac{1}{2\pi N}\int_{-\pi}^{\pi}\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\times\\ &\left(|G(e^{j\omega})|^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\left(|G(e^{j\omega})|^2\Phi_{r^p}+\Phi_v\right)+|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\right)+\\ &\lambda\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\times\\ &\left(|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\left(|G(e^{j\omega})|^2\Phi_{r^p}+\Phi_v\right)+|C(e^{j\omega},\boldsymbol{\rho}_i)|^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\right)d\omega \end{aligned}\tag{7.38}$$

$$\begin{aligned}=& \frac{1}{2\pi N}\int_{-\pi}^{\pi}\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\times\\ &\Big[(|G(e^{j\omega})|^2+\lambda)|S(e^{j\omega},\boldsymbol{\rho}_i)|^4(|G(e^{j\omega})|^2\Phi_{r^p}+\Phi_v)+\\ &(1+\lambda|C(e^{j\omega},\boldsymbol{\rho}_i)|^2)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\Big]d\omega\end{aligned}\tag{7.39}$$

From the expressions in this section, it can be seen how external perturbation will affect the relevant functions in relation to the covariance of the cost function gradient estimate.

- The asymptotic expressions for S_N and E_N are affine functions in the following variables. $S_N = f(\Phi_{r^p}^2, \Phi_v^2, \Phi_{r^p}\Phi_v)$ and $E_N = f(\Phi_v^2, \Phi_{r^p}\Phi_v)$, hence the asymptotic covariance estimate is also an affine function in $\Phi_{r^p}^2$, Φ_v^2 and $\Phi_{r^p}\Phi_v$.
- The Hessian estimate is an affine function in Φ_{r^p} and Φ_v only.

7.3.1 Unbiased gradient estimation with perturbation

From the general feedback loop, Figure 7.1, it is seen that the closed loop transfer functions are given by

$$y_t = GS(\boldsymbol{\rho}_i)r_t^p + S(\boldsymbol{\rho}_i)v_t \tag{7.40}$$

$$u_t = S(\boldsymbol{\rho}_i)r_t^p - C(\boldsymbol{\rho}_i)S(\boldsymbol{\rho}_i)v_t \tag{7.41}$$

It would be interesting to have a design of r^p which would not change the dynamics in the response of y or u with respect to the inputs, compared to the unperturbed case. If $r_t^p = \sqrt{\alpha}/G v_t$ would be realizable, the output in (7.40) will simplify to

$$y_t = GS(\boldsymbol{\rho}_i)\frac{\sqrt{\alpha}}{G}v_t + S(\boldsymbol{\rho}_i)v_t = (1+\sqrt{\alpha})S(\boldsymbol{\rho}_i)v_t$$

which is only a scaled expression of the output for the unperturbed case. This perturbation signal design will render the gradient estimate unbiased in case $\lambda = 0$ in (7.6), i.e. *minimum variance control*. It is optimal in the sense that this design will contribute to a better signal to noise ratio without driving the optimization of the control parameters to a biased optimum compared to the unperturbed case. In case where $r_p = \sqrt{\alpha}C(e^{j\omega}, \boldsymbol{\rho}_i)v_t$

$$u_t = S(\boldsymbol{\rho}_i)\sqrt{\alpha}C(e^{j\omega}, \boldsymbol{\rho}_i)v_t - C(\boldsymbol{\rho}_i)S(\boldsymbol{\rho}_i)v_t = (1+\sqrt{\alpha})C(\boldsymbol{\rho}_i)S(\boldsymbol{\rho}_i)v_t$$

which means that an equivalent design is possible with an unbiased gradient estimate, if the performance cost function only includes a penalty on the control (i.e. $\lambda \to \infty$). This is of course only of theoretical interest. The functional dependencies in (7.40) and (7.41) means that a perturbation design which will give scaled expressions for both y and u with respect to the unperturbed case does not exist.

In practical applications the actual random disturbance signal is unknown but the spectrum of the disturbance may be known. If the perturbation signal is generated using a signal with spectral properties equal to these of v, i.e. Φ_v, then the expected value of the gradient estimates will still be unbiased. If r_t^p and v_t are independent the spectrum of the output and the input in the two cases are:

$$\Phi_y = |G(e^{j\omega})|^2|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2\Phi_{r^p} + |S(e^{j\omega}, \boldsymbol{\rho}_i)|^2\Phi_v \tag{7.42}$$
$$\Phi_u = |S(e^{j\omega}, \boldsymbol{\rho}_i)|^2\Phi_{r^p} + |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2\Phi_v \tag{7.43}$$

Following the two optimal designs which has just be argued

$$\Phi_{r^p} = \frac{\alpha}{|G(e^{j\omega})|^2}\Phi_v \qquad \Rightarrow \Phi_y = (1+\alpha)|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2\Phi_v$$
$$\Phi_{r^p} = \alpha|C(e^{j\omega}, \boldsymbol{\rho}_i)|^2\Phi_v \Rightarrow \Phi_u = (1+\alpha)C(e^{j\omega}, \boldsymbol{\rho}_i)|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2\Phi_v$$

From these expressions it is seen that the only requirement is knowledge of the noise spectrum and the magnitude functions $|G(e^{j\omega})|^2$ and $|C(e^{j\omega}, \boldsymbol{\rho}_i)|^2$ in order to produce a spectrum of the in- and output which are scaled with $(1+\alpha)$, compared to the unperturbed case. Insuring that the spectrum are scaled, is a less strict requirement than having the signals y and u scaled. E.g. let the true system model contain a time delay such that $G(q) = q^{-k}\bar{G}(q)$. Since $|G(e^{j\omega})|^2 = |\bar{G}(e^{j\omega})|^2$, a perturbation signal generated by $r_t^p = \sqrt{\alpha}/\bar{G} v_t$ would only scale Φ_y up by $(1+\alpha)$ but

$$y_t = GS(\boldsymbol{\rho}_i)\frac{\sqrt{\alpha}}{\bar{G}}v_t + S(\boldsymbol{\rho}_i)v_t = (1+\sqrt{\alpha}q^{-k})S(\boldsymbol{\rho}_i)v_t$$

which will change the dynamic response and hence render the gradient estimate of the minimum variance cost function biased. This result gives some information for generation of the optimal perturbation signal for disturbance rejection tuning of the minimum variance controller. It is desirable to have an input signal with the same spectral properties as the random disturbance acting on the system. Furthermore this signal will have to be filtered through the inverse of the true plant dynamics.

In practice it is not possible to generate an optimal perturbation signal since the plant dynamics is unknown. On the other hand, the analysis in this section offers an optimal design strategy for the perturbation signal in case a plant and noise covariance estimates are available.

7.3.2 Influence of the perturbation power

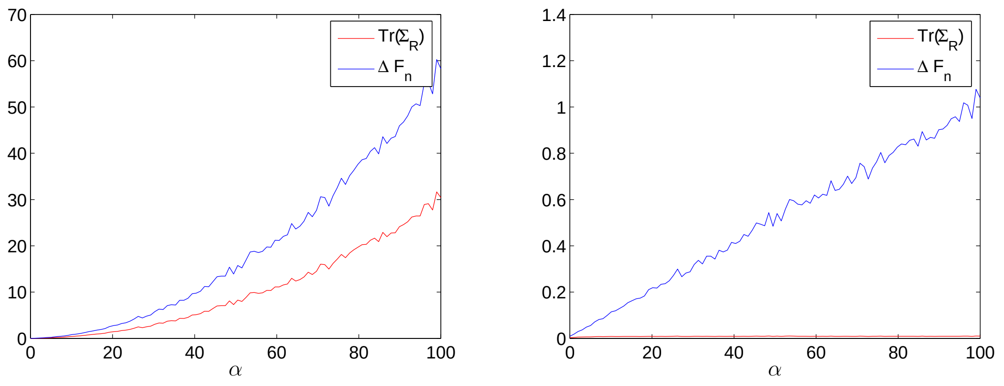
In this section it will be assumed that $\lambda = 0$ in the performance cost function which will therefore only depend on Φ_y. The perturbation signal spectrum will be chosen as $\Phi_{r^p} = (\alpha/|G(e^{j\omega})|^2)\Phi_v$ such that the only free parameter is α which will determine the power of the signal.

Using perturbations in the tuning algorithm will influence the covariance matrix of the performance cost function gradient estimate and hence the expected performance of the n'th iteration. Since expressions (7.33) and (7.34) show that the covariance matrix is proportional to the squared spectrum of the perturbation signal, it will be proportional to α^2. The true Hessian of the performance cost function, used in evaluation of Σ and ΔF_n, is independent of the perturbation, since this Hessian is evaluated at the optimum for the unperturbed problem. For $\alpha \to \infty$ it will therefore be expected that Σ and the control quality index will grow with α^2. In practice the true Hessian is not known and has to be estimated from the same perturbed data. Equation (7.39) shows that such a Hessian estimate is proportional to the perturbation spectrum and hence α. By substitution of the true Hessian with this perturbed Hessian estimate in the expressions for Σ and ΔF_n, it will be expected that Σ will approach a constant value when $\alpha \to \infty$ while the control quality index will grow linearly. These results are verified by simulation in Figure 7.3.

In case the perturbation signal is *kept constant* between iterations, the covariance expression for the performance cost will change. Since the perturbation signal does not change between iterations it will be regarded as a deterministic signal. Hence the multiplication between signals driven by the perturbation signal r_p will not contribute to the covariance. That implies that the term in S_N in (7.33) with the squared spectrum of the perturbation signal will be zero. Hence for a deterministic r_p

$$
\begin{aligned}
\lim_{N\to\infty} N\mathrm{Cov}[S_N(\boldsymbol{\rho})] = \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho})|^4 \times \Bigg[\left[\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho})\}\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho})\}^T\right]\Phi_v^2 + \\
\Bigg[2\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho})\}\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho})\}^T + \mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho})\}\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho})\}^T - \\
\mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho})\}^T\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho})\} + \mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho})\}\mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho})\}^T + \\
\mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho})\}\mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho})\}^T + \mathcal{R}e\{\Upsilon(e^{j\omega},\boldsymbol{\rho})\}\mathcal{R}e\{\Upsilon(e^{j\omega},\boldsymbol{\rho})\}^T + \\
\mathcal{I}m\{\Upsilon(e^{j\omega},\boldsymbol{\rho})\}\mathcal{I}m\{\Upsilon(e^{j\omega},\boldsymbol{\rho})\}^T \Bigg] \Phi_{r^p}\Phi_v \Bigg] d\omega \qquad (7.44)
\end{aligned}
$$

The covariance expression for E_N in (7.34) remains unchanged. Having the same realization for the perturbation signal will give a covariance expression for the per-



(a) True Hessian, $\mathbf{R} = \mathbf{H}$, for the unperturbed problem.

(b) Estimated Hessian, $\mathbf{R} = \hat{\mathbf{H}}$, using perturbations.

Figure 7.3. Simulations of $\mathrm{Tr}(\Sigma)$ and the control quality index, ΔF_n, for increasing power of the perturbation signal $\Phi_{r^p} = (\alpha/|G(e^{j\omega})|^2)\Phi_v$. 1000 data points are used in the simulation and 1000 repeated simulations are used to evaluate the covariance of the gradient estimate. The perturbation signal is *changed* between subsequent simulations.

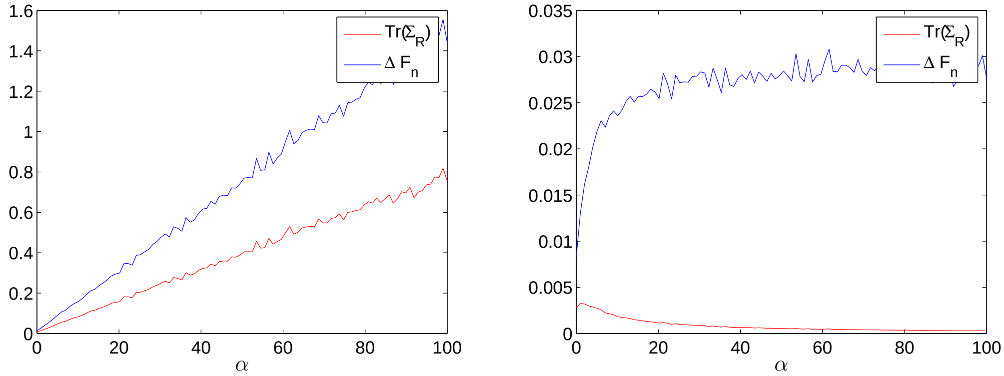
formance cost gradient estimate which is proportional to the perturbation signal spectrum and not the spectrum squared. The influence of the deterministic perturbation signal on Σ and ΔF_n is shown in Figure 7.4. *It is seen that using a constant perturbation signal while the Hessian is estimated from data, produces a covariance matrix Σ which approaches zero as the power of the perturbation signal is increased.*

7.4 A formal design criterion for the perturbation spectrum

The previous section has shown that introducing an external perturbation signal in the first of the experiments in the Iterative Feedback Tuning algorithm, can improve the convergence and decrease the necessary number of iterations when the objective is disturbance rejection. In this section we summarize the formal design criterion, outlined in Section 7.1.1, and discuss practical issues.

Denoting the design variables of the reference spectrum by $\boldsymbol{\vartheta}$ (with $\boldsymbol{\vartheta}^0$ corresponding to a zero reference signal), we have from (7.5) that a suitable design criterion is

$$\begin{aligned}\Delta F_n(\boldsymbol{\vartheta}) \triangleq & \frac{1}{2}\mathrm{Tr}\left\{\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0),\boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}\left(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta})-\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)\left(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta})-\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0)\right)^T\right\}+ \\ & \frac{1}{2}\mathrm{Tr}\left\{\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0),\boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}\boldsymbol{\Sigma}_n(\boldsymbol{\vartheta})\right\} \qquad (7.45)\end{aligned}$$



(a) True Hessian, $\mathbf{R} = \mathbf{H}$, for the unperturbed problem.

(b) Estimated Hessian, $\mathbf{R} = \hat{\mathbf{H}}$, using perturbations.

Figure 7.4. Simulations of $\mathrm{Tr}(\Sigma)$ and the control quality index, ΔF_n, for increasing power of the perturbation signal $\Phi_{r^p} = (\alpha/|G(e^{j\omega})|^2)\Phi_v$. 1000 data points are used in the simulation and 1000 repeated simulations are used to evaluate the covariance of the gradient estimate. The perturbation signal *remains unchanged* for all the subsequent simulations.

where

$$\boldsymbol{\Sigma}_n(\boldsymbol{\vartheta}) \triangleq \mathrm{E}\left[(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}))\,(\hat{\boldsymbol{\rho}}_n(\boldsymbol{\vartheta}) - \bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}))^T\right]$$

Recall that the first term in (7.45) is the bias, or the displacement of the optimal performance due to the external perturbation, and that the second term is the variance error. Under Conditions 1–6 in Section 7.2.1, (7.21) gives that $\boldsymbol{\Sigma}_n(\boldsymbol{\vartheta})$ can be expressed as

$$\boldsymbol{\Sigma}_n(\boldsymbol{\vartheta}) \approx \frac{1}{n}\boldsymbol{\Sigma}(\boldsymbol{\vartheta})$$

where n is the number of iterations that are going to be performed, and where

$$\boldsymbol{\Sigma}(\boldsymbol{\vartheta}) = a^2 \int_0^\infty e^{\boldsymbol{A}t}\boldsymbol{R}^{-1}\mathrm{Cov}\left[\widehat{\boldsymbol{J}(\bar{\boldsymbol{\rho}}, \boldsymbol{\vartheta})}\right]\boldsymbol{R}^{-1}e^{\boldsymbol{A}^T t}dt \tag{7.46}$$

(recall that a is the gain in the step-size, i.e. at iteration n, the step-size is $\gamma_n = a/n$). In (7.46), $\mathrm{Cov}\left[\widehat{\boldsymbol{J}(\bar{\boldsymbol{\rho}}, \boldsymbol{\vartheta})}\right]$ is given by (7.26)

$$\mathrm{Cov}\left[\widehat{\boldsymbol{J}(\boldsymbol{\rho}, \boldsymbol{\vartheta})}\right] = \mathrm{Cov}[S_N(\boldsymbol{\rho}, \boldsymbol{\vartheta})] + \mathrm{Cov}[E_N(\boldsymbol{\rho}, \boldsymbol{\vartheta})] \tag{7.47}$$

where asymptotic (in the experiment length N) expressions for $\mathrm{Cov}[S_N(\boldsymbol{\rho}, \boldsymbol{\vartheta})]$ and $\mathrm{Cov}[E_N(\boldsymbol{\rho}), \boldsymbol{\vartheta}]$ are given in (7.33)–(7.34) with Φ_{r^p} being the reference signal spectrum that corresponds to the parameter $\boldsymbol{\vartheta}$. Observe that these expressions hold when the disturbance v_t and the reference signal r_t are Gaussian distributed.

In case the gain direction $\mathbf{R}$ in the Iterative Feedback Tuning algorithm (7.10) is taken as $\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})}{\partial \boldsymbol{\rho}^2}$, the simplified expression (7.22) can be used resulting in that

$$\boldsymbol{\Sigma}_n(\boldsymbol{\vartheta}) \approx \frac{a^2}{n(2a-1)} \left[\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})}{\partial \boldsymbol{\rho}^2}\right]^{-1} \operatorname{Cov}\left[\widehat{\boldsymbol{J}(\bar{\boldsymbol{\rho}},\boldsymbol{\vartheta})}\right] \left[\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})}{\partial \boldsymbol{\rho}^2}\right]^{-1} \tag{7.48}$$

When full process knowledge is available all quantities in (7.45) can be computed from the equations above and thus one can optimize $\Delta F_n(\boldsymbol{\vartheta})$ in order to obtain a reference signal spectrum suitable for when using Iterative Feedback Tuning to tune a controller that is to be used for disturbance rejection. Since the design criterion $\Delta F_n(\boldsymbol{\vartheta})$ is based on a Taylor expansion it is recommended to introduce a constraint on the reference signal power in the optimization. There are many possibilities for parametrizing the reference spectrum. In the next section a straightforward method where filter coefficients are used as design variables $\boldsymbol{\vartheta}$. It is also possible to use a linear parametrization of the spectrum itself, we refer to Jansson and Hjalmarsson (2005) for details.

As in general experimental design algorithms, the evaluation of the optimal solution relies on knowledge of the true system which is not available (Goodwin and Payne, 1977; Gevers and Ljung, 1986; Bombois *et al.*, 2004). Therefore, practical use of the method will have to rely on an initial plant model. However, since the cost function appears to be smooth in many problems (see for example the next section), the accuracy of this model does not seem to be critical. The model may also be updated using the experimental data that is generated throughout the Iterative Feedback Tuning-experiments in order to successively improve the design.

7.5 An example

A simulation study is preformed in order to illustrate the ideas and advantages of introducing external perturbations in the Iterative Feedback Tuning method when tuning for disturbance rejection. For simplicity the control loop used is a discrete-time linear time-invariant transfer function model, and the controller has only two adjustable parameters. The random disturbance acting on the system is Gaussian white noise i.e. $e_t \in \mathcal{N}_{iid}(0,\sigma^2)$ where $\sigma = 1$. The nomenclature refers to the block diagram in Figure 7.1 where $v_t = H(q)e_t$.

$$\begin{aligned} &\text{Plant model:} && G(q) = \frac{q^{-1} - 0.5q^{-2}}{1 - 0.3q^{-1} - 0.28q^{-2}} \\ &\text{Noise model:} && H(q) = \frac{1}{1 + 0.9q^{-1}} \\ &\text{Controller:} && C(q) = \rho_1 + \rho_2 q^{-1} \end{aligned} \tag{7.49}$$

This system was used in Hildebrand *et al.* (2005b) to test the advantages of optimal pre-filters in Iterative Feedback Tuning for disturbance rejection. The simulation study is divided into two cases. In the first case a minimum variance control design is used, hence $\lambda = 0$ in (7.6). In this case the optimal design of the perturbation signal is known analytically. The second case treats the more general case where penalty on both in- and outputs are included in the quadratic performance cost function. In the second case $\lambda = 0.25$ is chosen, and the optimal perturbation signal is designed by optimizing the parameters in a data filter.

Before we proceed, we remark that in this example we have replaced $\frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}^0),\boldsymbol{\vartheta}^0)}{\partial \boldsymbol{\rho}^2}$ in the second term of (7.45) by

$$\frac{F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta}^0)}{F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})} \frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})}{\partial \boldsymbol{\rho}^2} \tag{7.50}$$

This approximation is accurate when

$$\nu \triangleq \frac{d}{d\boldsymbol{\rho}} \frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})}{\partial \boldsymbol{\rho}^2}\bigg|_{\boldsymbol{\vartheta}=\boldsymbol{\vartheta}^0}$$

is small since $g(\boldsymbol{\vartheta})$ has first order derivative at $\boldsymbol{\vartheta} = \boldsymbol{\vartheta}^0$ given by $g'(\boldsymbol{\vartheta}^0) = \nu$.

$$g(\boldsymbol{\vartheta}) \triangleq \frac{F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta}^0)}{F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})} \frac{\partial^2 F(\bar{\boldsymbol{\rho}}(\boldsymbol{\vartheta}),\boldsymbol{\vartheta})}{\partial \boldsymbol{\rho}^2}$$

7.5.1 Case one: Minimum variance control

In this section the external perturbation signal is given by $r_t^p = \sqrt{\alpha}/G v_t$ where the plant model G and the noise model H are assumed known. Hence α is the only free parameter which will give the input power of the perturbation signal. The tuning of the controller is performed for the minimum variance design where $\lambda = 0$ in the performance cost function (7.6). An external perturbation increases the value of the performance cost when applied. Figure 7.5 shows the normalized cost function as a surface on a grid of controller parameter values. These surface plots are smooth functions since the same noise realization has been used for each grid point and in both the perturbation signal design and in the evaluation of the cost function. The cost function is of course only a smooth function when the number of samples, $N \to \infty$, which is not practically realizable. In this simulation $N = 512$.

The two surfaces have the same minimum in Figure 7.5, since for this idealized case

$$F(\boldsymbol{\rho}, \Phi_{r^p}) = (\sqrt{\alpha} + 1)^2 F(\boldsymbol{\rho}, 0)$$

This property means that the perturbation gives the desired change in the curvature of the performance cost function to yield a faster convergence. In order to illustrate this result further a series of Monte Carlo experiments are performed using Perturbed Iterative Feedback Tuning. Initially the control parameters has the optimal value, but due to the stochastic nature of the data the tuning will move the parameters away from this value for repeated iterations. In 1000 experiments, 10 iterations have been performed from the same optimal starting point, and the values of the resulting set of parameters has been saved. 1000 data points has been collected and used in each iteration of the tuning. For four different values of α in the perturbation signal, the results are presented in Figure 7.6. The variance and the cross-covariance of the resulting control parameters are reported in Table 7.1. From the results of the Monte Carlo simulations in Figure 7.6 and Table 7.1 it is obvious that increasing the value of α in the perturbation signal, produces an optimization problem with a statistically better defined optimum.

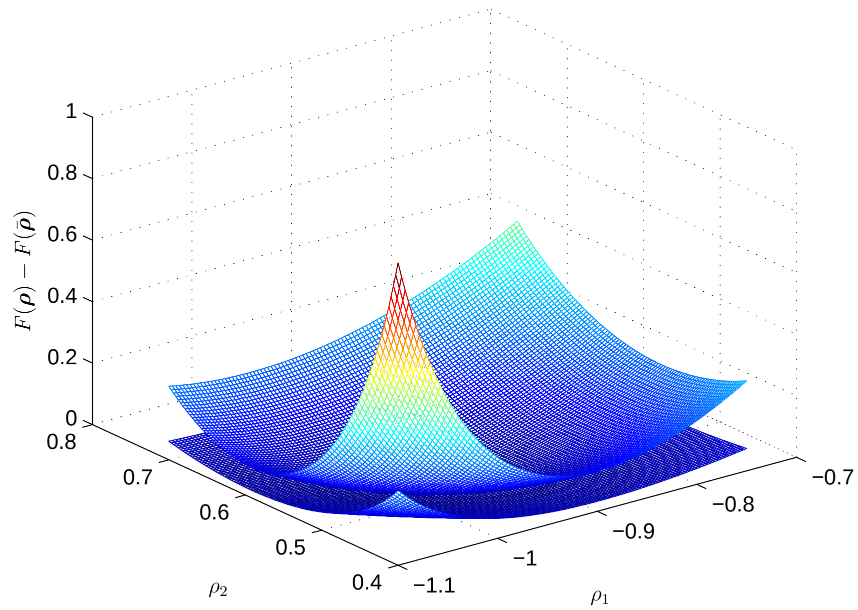


Figure 7.5. Surface plot of the normalized performance cost function on a control parameter grid. The lower surface is the performance cost when $\alpha = 0$ and the upper surface is for $\alpha = 1$. The same noise realization v_t has been used at each grid point and in the perturbation signal design in order to obtain a smooth surface.

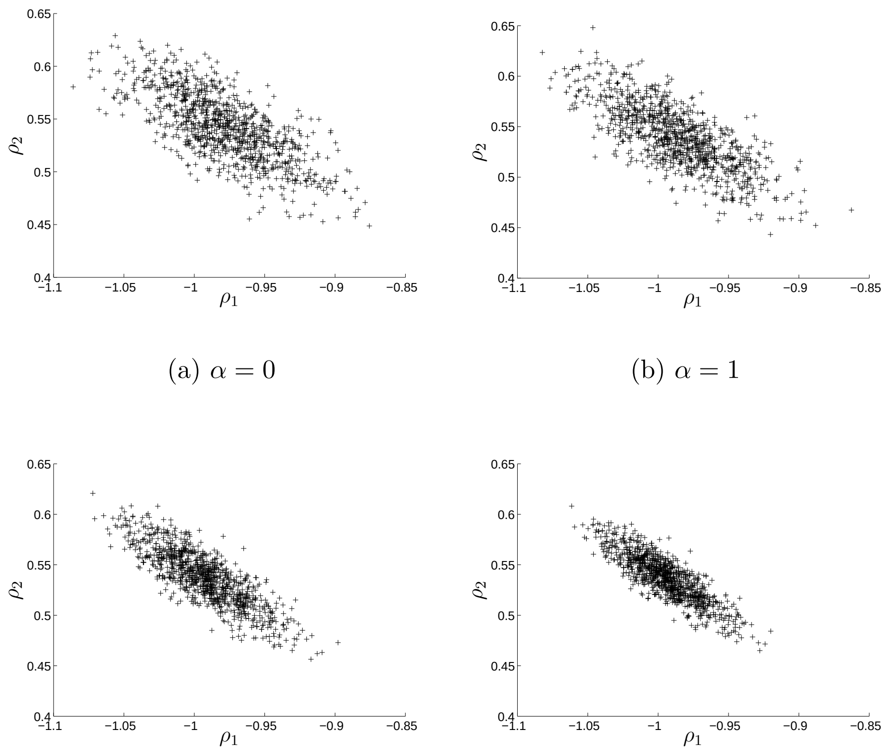
7.5.2 Case two: The general performance cost function

In this example the same system is used but the performance cost function is changed so that $\lambda = 0.25$. Initially the perturbation signal is formed in the same way as in the previous example, i.e. $r_t^p = \sqrt{\alpha}/G v_t$. Figure 7.7 show how the optimum of the cost function depends on the perturbation power when $\lambda \neq 0$. This figure also shows 30 contour lines for the cost functions, hence the contour lines in the two plots do not represent equal levels. The optimal control parameters for each surface and the corresponding value of the performance cost function are given in Table 7.2.

In the general case where $\lambda \neq 0$ in the performance cost function, the ratio $\frac{F(\boldsymbol{\rho}(\Phi_{r^p}),0)}{F(\boldsymbol{\rho}(\Phi_{r^p}),\Phi_{r^p})}$ in (7.50) is not constant. This is evident from Figure 7.8 which also shows that the approximation is reasonable close to $\bar{\boldsymbol{\rho}}$. The figure also show that the curvature of this surface is small close to $\bar{\boldsymbol{\rho}}(\Phi_{r^p})$.

7.5.2.1 Optimizing the perturbation signal

Since the value for λ is not very large, it is possible that the optimal design for $\lambda = 0$ yields a reasonable filter choice. From Figure 7.7 it is seen that the optimum for the perturbed problem has not been moved very far from the unperturbed problem in the control parameter space. Therefore the structure of the filter used to generate the perturbation signal is chosen identical to the inverted system model. The initial



(a) $\alpha = 0$

(b) $\alpha = 1$

(c) $\alpha = 5$

(d) $\alpha = 10$

Figure 7.6. The final control parameters from 1000 Monte Carlo experiments each with 10 iterations in the Perturbed Iterative Feedback Tuning method. All iterations are initiated at the optimal value for the control parameters. The value of α in scaling the perturbation signal has been changed in four steps from zero up to 10.

values for the filter parameters are selected as the model parameters $\boldsymbol{\theta}$.

$$r_t^p = \frac{1}{G(q)} H(q) e_t = G_{r^p}(q) H(q) e_t, \quad e_t \in \mathcal{N}_{iid}(0, \sigma^2) \tag{7.51}$$

where

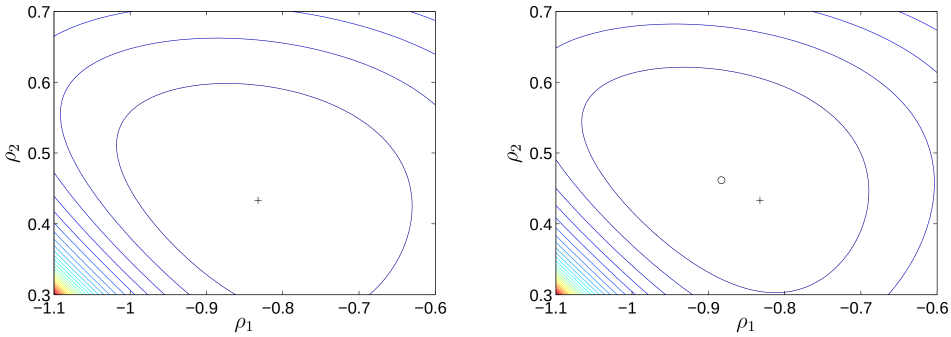
$$G_{r^p}(q) = \frac{1 + \vartheta_1 q^{-1} + \vartheta_2 q^{-2}}{q^{-1} + \vartheta_3 q^{-2}}, \quad \boldsymbol{\vartheta}_0^T = \boldsymbol{\theta}^T = [-0.3 \ -0.28 \ -0.5] \tag{7.52}$$

hence non causal filtering is required. In this filter design the parameter α which adjusts the gain will not be included in $\boldsymbol{\vartheta}$ as a free parameter. Hence variance of the perturbation signal will be determined by the remaining free parameters. The optimal set of filter parameters can be detirmined by the minimization of the control quality index, $\Delta F_n(\Phi_{r^p})$ in (7.45) as an unconstrained problem. The reason being that the optimal perturbation power will be a trade off between the displacement of the optimal control parameters due to perturbation, and the distance between the expected and optimal performance. The optimal solution based on full process insight where computed as

$$\boldsymbol{\vartheta}_{opt}^T = [-8.115 \ -10.21 \ 0.5434]$$

Variance	$\sigma^2_{\rho_1} \cdot 10^3$	$\sigma^2_{\rho_2} \cdot 10^3$	$\sigma_{\rho_1,\rho_2} \cdot 10^3$
$\alpha = 0$	1.24	1.03	-0.817
$\alpha = 1$	1.16	1.02	-0.820
$\alpha = 5$	0.757	0.743	-0.623
$\alpha = 10$	0.522	0.531	-0.451

Table 7.1. The variance and the cross-covariance for the resulting set of control parameters from 1000 Monte Carlo experiments each with 10 iterations in the Perturbed Iterative Feedback Tuning method. All iterations are initiated a t the optimal value for the control parameters. Results are given for different values of α.



(a) $\alpha = 0$ (b) $\alpha = 1$

Figure 7.7. Contour plots with each 30 conture lines of the performance cost function, with $\lambda = 0.25$, on a control parameter grid. The perturbation signal $r_t^p = \sqrt{\alpha}/G v_t$ is applied and the optimal set of control parameters is marked with a + for $\alpha = 0$ and with ∘ for $\alpha = 1$. The same noise realization v_t has been used at each grid point and in the perturbation signal design in order to obtain a smooth surface.

and the control quality index were improved from $5.123 \cdot 10^{-3}$ to $0.2658 \cdot 10^{-3}$.

7.5.2.2 Perturbed Iterative Feedback Tuning

In the following four series of 1000 Monte Carlo experiments are performed each containing $n = 10$ iterations with Perturbed Iterative Feedback Tuning. Initially the loop starts with the optimal set of control parameters for the unperturbed operation.

- The first series is classical Iterative Feedback Tuning without a perturbation signal.

- In the second series the optimal designed perturbation filter for $\lambda = 0$ is used, hence $r_t^p = {H(q)}/{G(q)} e_t$. When λ is equal 0.25, this design is expected to produce a cloud of Monte Carlo solutions which is more dense but biased compeared to the first series.

	$\bar{\rho}_1$	$\bar{\rho}_2$	$F(\bar{\boldsymbol{\rho}})$	$F_{corr}(\bar{\boldsymbol{\rho}})$
$\alpha = 0$	-0.8323	0.4333	1.5008	-
$\alpha = 1$	-0.8828	0.4616	5.2697	1.5051

Table 7.2. The optimal set of control parameters for the two experiments with $\lambda = 0.25$ where the perturbation signal is given by $r_t^p = \sqrt{\alpha}/G v_t$. The value of the performance cost function for the optimal set is given together with the corrected value which compensates for the effect of perturbation.

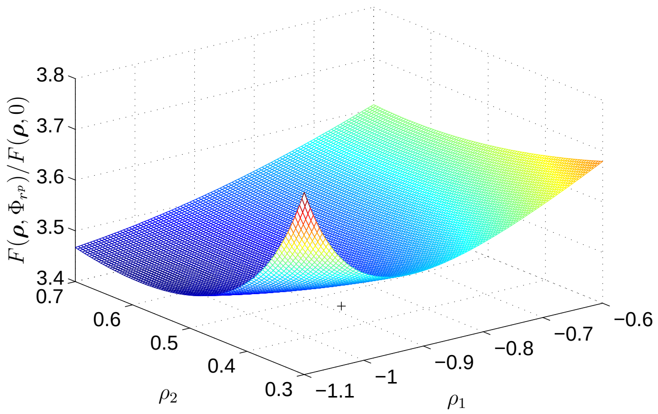


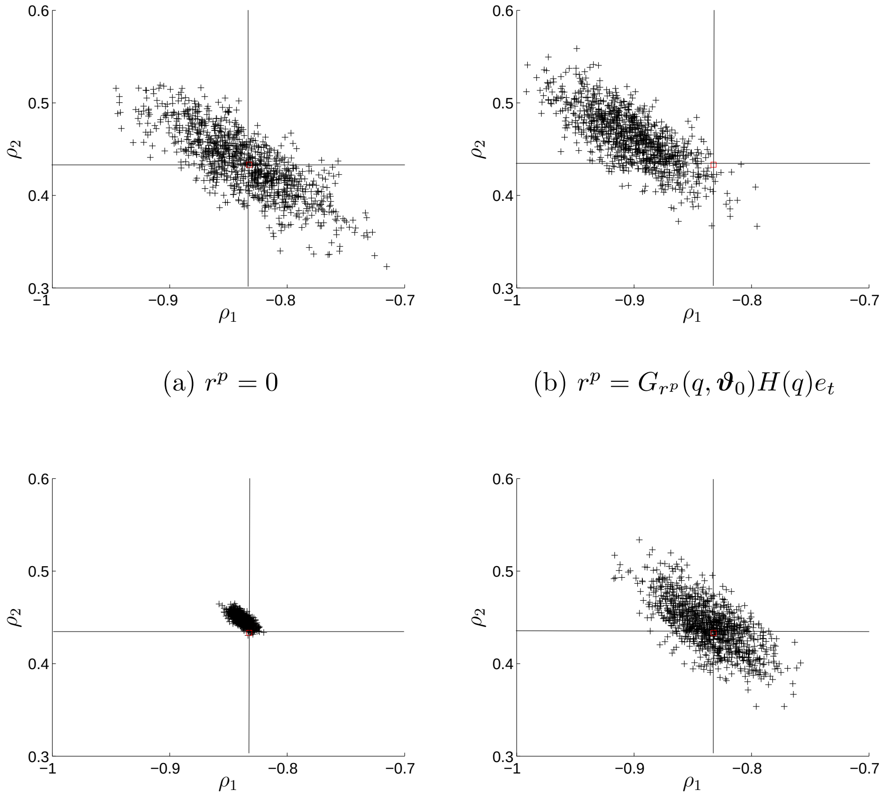
Figure 7.8. Surface plot of the performance cost function ratio $\frac{F(\boldsymbol{\rho}(\Phi_{r^p}),0)}{F(\boldsymbol{\rho}(\Phi_{r^p}),\Phi_{r^p})}$, with $\lambda = 0.25$, on a control parameter grid. The two evaluations of the performance cost are affected by perturbation signals $r_t^p = \sqrt{\alpha}/G v_t$ with $\alpha = 0$ and $\alpha = 1$ respectively. The same noise realization v_t has been used at each grid point and in the perturbation signal design in order to obtain a smooth surface. The point marked with + is the optimal set of control parameters for the perturbed problem.

- The third series uses the perturbation signal with the optimal parameters which was presented in Section 7.5.2.1. Since the variance of the perturbation signal is unconstrained in the optimization the variance for $r^p = G_{r^p}(q, \boldsymbol{\vartheta}_0)H(q)e_t$ is 2.842 while it is 165.6 for $r^p = G_{r^p}(q, \boldsymbol{\vartheta}_{opt})H(q)e_t$

- The fourth and last series, the third experiment is repeated but such a strong signal will not be allowed during the tuning and the filter is scaled accordingly. Using $\alpha G_{r^p}(q, \boldsymbol{\vartheta}_{opt})$ where $\alpha = \sqrt{2.842}/\sqrt{165.6}$, will give a variance of the optimal perturbation signal which is the same as for $r^p = G_{r^p}(q, \boldsymbol{\vartheta}_{opt})H(q)e_t$.

The results of these four trials are shown in Figure 7.9 as scatter plot of $\boldsymbol{\rho}_n$ together with the optimal solution for the unperturbed problem as a red square. Table 7.3 presents the mean value, the variance and the cross-covariance for the Monte Carlo solutions together with the optimal control parameter values for the unperturbed case.

The results in Figure 7.9 and Table 7.3 clearly illustrates the advantage of introducing external perturbations when tuning for disturbance rejection. The optimal set of



(a) $r^p = 0$

(b) $r^p = G_{r^p}(q, \boldsymbol{\vartheta}_0)H(q)e_t$

(c) $r^p = G_{r^p}(q, \boldsymbol{\vartheta}_{opt})H(q)e_t$

(d) $r^p = \alpha G_{r^p}(q, \boldsymbol{\vartheta}_{opt})H(q)e_t$

Figure 7.9. The final control parameters from 1000 Monte Carlo experiments each with 10 iterations in the Perturbed Iterative Feedback Tuning method. All iterations are initiated at the optimal value for the control parameters for the unperturbed problem which is marked as a red square and with the straight lines. The perturbations signals used in series one to four corresponds to subfigure a, b, c, and d respectively.

perturbation filter parameters both significantly reduce the variance of final control parameters from the 1000 Monte Carlo experiments, and reduce the displacement of the optimal control parameter solutions for the perturbed and unperturbed problem. It is possible to evaluate the optimal filter parameters for generating the perturbation signal as an unconstrained optimization. Constrains can then be included by a scaling the gain of the filter, which has been done for Figure 7.9d.

7.6 Conclusions

The convergence properties of the Perturbed Iterative Feedback Tuning algorithm for optimizing control parameters for disturbance rejection problems, have been investigated. Asymptotic expressions for the covariance of the cost function gradient have been derived and a control quality index for Perturbed Iterative Feedback Tuning is proposed. It is shown that using a deterministic external perturbation

Statistic	$mean(\boldsymbol{\rho}_1)$	$mean(\boldsymbol{\rho}_2)$	$\sigma^2_{\rho_1} \cdot 10^3$	$\sigma^2_{\rho_2} \cdot 10^3$	$\sigma_{\rho_1,\rho_2} \cdot 10^3$
$r^p = 0$ (optimal)	-0.8323	0.4333	-	-	-
$r^p = 0$	-0.8369	0.4353	1.57	1.22	-1.09
$r^p = G_{r^p}(q, \boldsymbol{\vartheta}_0)H(q)e_t$	-0.9019	0.4628	1.07	0.899	-0.726
$r^p = G_{r^p}(q, \boldsymbol{\vartheta}_{opt})H(q)e_t$	-0.8382	0.4477	0.0275	0.0311	-0.0204
$r^p = \alpha G_{r^p}(q, \boldsymbol{\vartheta}_{opt})H(q)e_t$	-0.8371	0.4418	0.698	0.718	-0.488

Table 7.3. The mean, variance and the cross-covariance for the resulting set of control parameters from 1000 Monte Carlo experiments each with 10 iterations in the Perturbed Iterative Feedback Tuning method. All iterations are initiated at the optimal value for the control parameters for the unperturbed problem.

signal in the tuning will affect the control quality index. The magnitude of the improvement depends on the power and the frequency content of the perturbation signal.

For minimal variance control design an analytical expression is derived for a parameterization of the perturbation signal which is optimal in the sense that it converges to the same set of control parameters as the unperturbed case. This optimal design is illustrated in a simulation example where it is shown that increasing the power of the perturbation signal improves the control quality index. For a general cost function with quadratic penalty on both the output and the input, there does not exist such an unbiased optimal parameterization of the perturbation signal. An algorithm for minimizing the control quality index for this general case is proposed based on process insight. This algorithm is shown to be able to produce a perturbation signal which significantly improves the control quality index. Hence Perturbed Iterative Feedback Tuning performs better than classical Iterative Feedback Tuning when tuning for disturbance rejection.

Investigation on optimal parameterization of the perturbation signal and the performance of Perturbed Iterative Feedback Tuning in case of very limited process insight remain for future work.

8
Conclusions

In this thesis data driven tuning of control parameters for optimization of closed loop performance have been investigated with a special focus on the Iterative Feedback Tuning method. A short review of iterative methods for performance enhancement has been provided with a clear classification in direct and indirect methods. The class of indirect methods is model based and relies on subsequent model identification and certainty equivalence control design. The direct methods are the class of purely data driven algorithms which optimize the performance, excluding the model identification step. Iterative Feedback Tuning has been related to two recent direct tuning methods, the Correlation-based Tuning and Virtual Reference Feedback Tuning. A more detailed review of the developments in the Iterative Feedback Tuning method is presented separately, discussing the perspectives and pinpointing some shortcomings of the method. The main advantages of the method are that:

- The tuning method is easy to apply - also for operators without a higher technical degree.
- It is flexible with respect to optimization criterion.
- The method offers very few restrictions on the system and the loop. It is only demanded that the data filters used in the evaluation of gradient estimated of the in- and outputs are proper and stable. Use of pre-filters may ensure this property.
- All experiments are conducted in closed loop, and only one experiment in each iteration perturbs the process.

Disadvantages of the method are:

- The tuning relies on a large number of plant experiments with is proportional to the number of iterations. The number of required experiments in one iteration grows with the dimension of the number of pairings for multivariable system. Approximations do exist which solves the dimension problem for multivariable systems, at the expense of introducing bias in the estimate in the gradient of the performance cost function.
- The algorithm may show a slow rate of convergence due to the stochastic nature of the produced gradient estimate of the cost function. This is especially a problem when tuning a control loop for disturbance rejection.
- The basic algorithm does neither ensure stability nor robustness of the achieved loop. Results have been published which handle this inconvenience at the expense of making the tuning method computationally more complicated.

Based on the investigation of the Iterative Feedback Tuning method two different types of novel contributions have been developed and are documented in this thesis. The first type of contribution is related to data driven tuning on control structures where the application of Iterative Feedback Tuning is novel but thought to have a significant potential. Two such control structures which have been investigated are nonlinear inventory control and state space control structures with state estimation. The second type of contribution is related to speeding up the convergence rate of the data driven tuning for disturbance rejection, and hence reduce the total number of required plant experiments. This improvement is achieved by introducing external perturbation in order to introduce excitation and a higher information content in data.

The application of Iterative Feedback Tuning on a system with an inventory controller in the feedback loop is a straight forward application of the basic algorithm. In the control law, the process model is imbedded with e.g. proportional and integral action in order to compensate for model mismatch and ensure offset free tracking. Tuning of the gain and the integral time in the PI part is possible despite the general nonlinear nature of the controller, since the data filters required by the Iterative Feedback Tuning fulfill the basic requirement. Tuning of these parameters by classical tuning rules would in general not be possible. This work has successfully been tested on a multivariable implementation of inventory control of liquid levels of a pilot scale plant of the classical four tank system.

Extending the application of Iterative Feedback Tuning to adjust the feedback gain and possibly the Kalman filter gain in a state space control structure with observer is possible by realizing the feedback control representation in transfer function form. This controller will then depend on both these two gains, but also on the parameters in the estimated system model which is used by the observer. It is shown that the data driven tuning method converge to known analytical solutions of both the feedback and observer gain, when the same cost function is used by the tuning and in solving the stationary Riccati equations for optimal design of LQG control.

When tuning a set of control parameters by Iterative Feedback Tuning for a disturbance rejection problem the rate of convergence is often very slow even when using long data series. This problem is caused by the stochastic nature of how the algorithm constructs the gradient estimate of the performance cost function, with respect to the control parameters by using new data series in each iteration. For the disturbance rejection problem only the noise in the first experiment is driving the optimization, while the noise in the gradient experiments is producing a contribution to the estimation variance. If the variance is large compared to the mean, which is often the case for disturbance rejection, it has a negative effect on the convergence rate. This problem has previously been addressed by designing prefilters, which are used for the reference signal in the gradient experiments. The contribution presented here has applied external perturbation as a mean to achieve the same end. When a deterministic external signal is applied in the first experiment it affects the optimization objective since the control is rejecting both noise and the probing. The aim is to shape the curvature of the cost function such that the optimization problem is less sensitive to the noise in the gradient experiments without changing the optimum away from the original problem. A detailed analysis of the effect of such external perturbation signals is given. It is seen that in the general case it is not possible to apply a perturbation signal without introducing bias in the optimization objective. For the special case of tuning a minimum variance controller a design for unbiased perturbations exists. This analysis is extended to give a design algorithm for optimizing perturbation signals for the Iterative Feedback Tuning method, which balance the negative effect of bias against the positive effect of reducing the expected distance, after a fixed number of iterations, to the biased optimum. Data driven tuning of a control loop using this algorithm has been named Perturbed Iterative Feedback Tuning. This algorithm has shown good improvements when the perturbation design is based on knowledge of the true system.

8.1 Future work

Through the work which has been conducted for writing this thesis several open problems and potentials have become apparent which have not been pursued due to time limitations. One general limitation is the extent where it has been possible to apply the proposed methods on real systems and not only rely on simulation results. The pilot scale, four tank test facility at the Department of Chemical and Biochemical Engineering, Technical University of Denmark would serve as a convenient test bench for both tuning of state space control systems and also Perturbed Iterative Feedback Tuning. It is expected that test of these and further developments will be conducted on this or other test facilities in order to, not only virtually, validate the proposed methods.

In relation to the tuning of state space control structures, a next step would be a more profound analysis of tuning the feedback and the observer gains given parametric uncertainty in the model estimate. The effect caused by under-modelled dynamics is another interesting extension. A step further would be tuning of the model parameters and subsequent recalculation of the feedback and observer gain. In which case the gradient calculation would be troubled by the optimization involved in the calculation of these gains. If a solution to this problem exists further extension to the constrained case and Model Predictive Control would provide a great potential for optimizing process operation. This might provide a systematically tuning of the key parameters in model based controllers for processes where limited process information is available.

Future work in relation to Perturbed Iterative Feedback Tuning will be to show the potential of the method, even for systems where limited process knowledge is available. Furthermore it would be interesting to look at the parameterizations of the perturbation signal. Especially would guidelines for utilizing Perturbed Iterative Feedback Tuning, based on no process information, be very useful. A systematic comparison with competing techniques would be obvious. Since the perturbations are applied in the first experiment and related problems are focusing on the gradient experiments, combinations may be possible which would limit the total level of excitation on each experiment and still achieve the same benefits with respect to convergence.

Appendices

A

Data Filters for the Iterative Feedback Tuning Method

Computation of the estimate of the gradient, $\hat{\mathbf{J}}(\boldsymbol{\rho})$, in the Iterative Feedback Tuning method requires calculation of the data filters used in:

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho}_i)} \frac{\partial C_r}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})\mathbf{r} - \frac{1}{C_r(\boldsymbol{\rho}_i)} \frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})\mathbf{y} \tag{3.6a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \frac{\partial C_r}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})\mathbf{r} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})\mathbf{y} \tag{3.6b}$$

or in

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho}_i)} \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) T(\boldsymbol{\rho})\mathbf{r} - \frac{1}{C_r(\boldsymbol{\rho}_i)} \frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})(\mathbf{r} - \mathbf{y}) \tag{3.12a}$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) S(\boldsymbol{\rho})\mathbf{r} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})(\mathbf{r} - \mathbf{y}) \tag{3.12b}$$

depended on whether a one or two degree of freedom controller are in operation. These filters $\frac{1}{C_r}\frac{\partial C_r}{\partial \boldsymbol{\rho}}$ and $\frac{1}{C_r}\frac{\partial C_y}{\partial \boldsymbol{\rho}}$ or $\frac{1}{C}\frac{\partial C}{\partial \boldsymbol{\rho}}$ are calculated based on the particular controller implemented in the loop.

A.1 PID Control

In case where a PID controller is used in the feedback loop, the controller is given by the transfer function from the error between the measurement and the reference to the control signal as:

$$C : \frac{\mathbf{U}(s)}{\mathbf{E}(s)} = K_c \left[1 + \frac{1}{\tau_I s} + \tau_D s \right] \tag{A.3}$$

In practical application a derivative filter is supplied to the derivative term and a two degree of freedom control is often used in order to avoid derivative action on the reference signal.

$$C_r : \frac{\mathbf{U}(s)}{\mathbf{R}(s)} = K_c \left[1 + \frac{1}{\tau_I s} \right] \tag{A.4a}$$

$$C_y : \frac{\mathbf{U}(s)}{\mathbf{Y}(s)} = -K_c \left[1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{(\alpha \tau_D s + 1)^n} \right] \tag{A.4b}$$

where n is the order of the derivative filter and α becomes an additional parameter.

A.1.1 First Order Derivative Filter

For a first order filter the controller transfer functions are

$$C_r : \frac{\mathbf{U}(s)}{\mathbf{R}(s)} = K_c \left[1 + \frac{1}{\tau_I s}\right] \tag{A.5a}$$

$$C_y : \frac{\mathbf{U}(s)}{\mathbf{Y}(s)} = -K_c \left[1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{(\alpha \tau_D s + 1)}\right] \tag{A.5b}$$

Derivatives $\frac{\partial C_y}{\partial \boldsymbol{\rho}}$

$$\frac{\partial C_y}{\partial K_c} = \left[1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{(\alpha \tau_D s + 1)}\right]$$

$$\frac{\partial C_y}{\partial \tau_I} = \frac{-K_c}{\tau_I^2 s}$$

$$\frac{\partial C_y}{\partial \tau_D} = \frac{K_c s}{(\alpha \tau_D s + 1)^2}$$

$$\frac{\partial C_y}{\partial \alpha} = \frac{-K_c \tau_D^2 s^2}{(\alpha \tau_D s + 1)^2}$$

Filters $\frac{1}{C_y}\frac{\partial C_y}{\partial \boldsymbol{\rho}}$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial K_c} = \frac{1}{K_c}$$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial \tau_I} = \frac{-(\alpha \tau_D s + 1)}{(1+\alpha)\tau_I^2 \tau_D s^2 + (\tau_I^2 + \alpha \tau_I \tau_D)s + \tau_I}$$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial \tau_D} = \frac{\tau_I s^2}{\alpha(1+\alpha)\tau_I \tau_D^2 s^3 + ((1+2\alpha)\tau_I \tau_D + \alpha^2 \tau_D^2)s^2 + (\tau_I + 2\alpha \tau_D)s + 1}$$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial \alpha} = \frac{-\tau_I \tau_D^2 s^3}{\alpha(1+\alpha)\tau_I \tau_D^2 s^3 + ((1+2\alpha)\tau_I \tau_D + \alpha^2 \tau_D^2)s^2 + (\tau_I + 2\alpha \tau_D)s + 1}$$

Filters $\frac{1}{C_r}\frac{\partial C_y}{\partial \boldsymbol{\rho}}$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial K_c} = \frac{(1+\alpha)\tau_I \tau_D s^2 + (\tau_I + \alpha \tau_D)s + 1}{K_c(\alpha \tau_I \tau_D s^2 + (\tau_I + \alpha \tau_D)s + 1)}$$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial \tau_I} = \frac{-1}{\tau_I^2 s + \tau_I}$$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial \tau_D} = \frac{\tau_I s^2}{\alpha^2 \tau_I \tau_D^2 s^3 + (2\alpha \tau_I \tau_D + \alpha^2 \tau_D^2)s^2 + (\tau_I + 2\alpha \tau_D)s + 1}$$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial \alpha} = \frac{-\tau_I \tau_D^2 s^3}{\alpha^2 \tau_I \tau_D^2 s^3 + (2\alpha \tau_I \tau_D + \alpha^2 \tau_D^2)s^2 + (\tau_I + 2\alpha \tau_D)s + 1}$$

A.1.2 Second Order Derivative Filter

For a second order filter the controller are

$$C_r : \frac{\mathbf{U}(s)}{\mathbf{R}(s)} = K_c \left[1 + \frac{1}{\tau_I s}\right] \tag{A.6a}$$

$$C_y : \frac{\mathbf{U}(s)}{\mathbf{Y}(s)} = -K_c \left[1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{(\alpha\tau_D s + 1)^2}\right] \tag{A.6b}$$

Derivatives $\frac{\partial C_y}{\partial \boldsymbol{\rho}}$

$$\frac{\partial C_y}{\partial K_c} = \left[1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{(\alpha\tau_D s + 1)^2}\right]$$

$$\frac{\partial C_y}{\partial \tau_I} = \frac{-K_c}{\tau_I^2 s}$$

$$\frac{\partial C_y}{\partial \tau_D} = \frac{-K_c s(\alpha\tau_D s - 1)}{(\alpha\tau_D s + 1)^3}$$

$$\frac{\partial C_y}{\partial \alpha} = \frac{2K_c\tau_D^2 s^2}{(\alpha\tau_D s + 1)^3}$$

Filters $\frac{1}{C_y}\frac{\partial C_y}{\partial \boldsymbol{\rho}}$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial K_c} = \frac{1}{K_c}$$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial \tau_I} = \frac{-(\alpha\tau_D s + 1)^2}{\alpha^2\tau_I^2\tau_D^2 s^3 + ((1+2\alpha)\tau_I^2\tau_D + \alpha^2\tau_I\tau_D^2)s^2 + (\tau_I^2 + 2\alpha\tau_I\tau_D)s + \tau_I}$$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial \tau_D} =$$

$$\frac{(\tau_I - \alpha\tau_I\tau_D s)s^2}{\alpha^3\tau_I\tau_D^3 s^4 + (\alpha(1+3\alpha)\tau_I\tau_D^2 + \alpha^3\tau_D^3)s^3 + ((1+3\alpha)\tau_I\tau_D + 3\alpha^2\tau_D^2)s^2 + (\tau_I + 3\alpha\tau_D)s + 1}$$

$$\frac{1}{C_y}\frac{\partial C_y}{\partial \alpha} =$$

$$\frac{-2\tau_I\tau_D^2 s^3}{\alpha^3\tau_I\tau_D^3 s^4 + (\alpha(1+3\alpha)\tau_I\tau_D^2 + \alpha^3\tau_D^3)s^3 + ((1+3\alpha)\tau_I\tau_D + 3\alpha^2\tau_D^2)s^2 + (\tau_I + 3\alpha\tau_D)s + 1}$$

Filters $\frac{1}{C_r}\frac{\partial C_y}{\partial \boldsymbol{\rho}}$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial K_c} = \frac{\alpha^2\tau_I\tau_D^2 s^3 + ((1+2\alpha)\tau_I\tau_D + \alpha^2\tau_D^2)s^2 + (\tau_I + 2\alpha\tau_D)s + 1}{K_c(\alpha^2\tau_I\tau_D^2 s^3 + (2\alpha\tau_I\tau_D + \alpha^2\tau_D^2)s^2 + (\tau_I + 2\alpha\tau_D)s + 1)}$$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial \tau_I} = \frac{-1}{\tau_I^2 s + \tau_I}$$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial \tau_D} = \frac{(\tau_I - \alpha\tau_I\tau_D s)s^2}{\alpha^3\tau_I\tau_D^3 s^4 + (3\alpha^2\tau_I\tau_D^2 + \alpha^3\tau_D^3)s^3 + (3\alpha\tau_I\tau_D + 3\alpha^2\tau_D^2)s^2 + (\tau_I + 3\alpha\tau_D)s + 1}$$

$$\frac{1}{C_r}\frac{\partial C_y}{\partial \alpha} = \frac{-2\tau_I\tau_D^2 s^3}{\alpha^3\tau_I\tau_D^3 s^4 + (3\alpha^2\tau_I\tau_D^2 + \alpha^3\tau_D^3)s^3 + (3\alpha\tau_I\tau_D + 3\alpha^2\tau_D^2)s^2 + (\tau_I + 3\alpha\tau_D)s + 1}$$

B

Derivation of covariance expression for the cost function estimate $\hat{F}(\boldsymbol{\rho})$ in standard Iterative Feedback Tuning.

In this section the asymptotic covariance expression for the performance cost function estimate will be derived. This covariance expression has been shown in Section 7.2.1 to consist of the sum of the asymptotic covariance for the sums S_N and E_N reflecting the deterministic and the variance part of the gradient estimate respectively. These two sums are given by:

$$\begin{aligned}
S_N(\boldsymbol{\rho}_i) =& \frac{1}{N}\sum_{t=1}^{N}\left[(S(\boldsymbol{\rho}_i)v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} G S(\boldsymbol{\rho}_i)^2 v_t^1\right) + \right.\\
&\left.\lambda(-S(\boldsymbol{\rho}_i)C(\boldsymbol{\rho}_i)v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}_i)^2 v_t^1\right)\right]\\
E_N(\boldsymbol{\rho}_i) =& \frac{1}{N}\sum_{t=1}^{N}\left[(S(\boldsymbol{\rho}_i)v_t^1)\left(\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}_i) v_t^2\right) + \right.\\
&\left.\lambda(-S(\boldsymbol{\rho}_i)C(\boldsymbol{\rho}_i)v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}_i) C(\boldsymbol{\rho}_i) v_t^2\right)\right]
\end{aligned}$$

In the following derivations it will be used that S_N consists of four signals that are driven by the same noise realization. Hence all signals are correlated. E_N also consists of four signals but these are driven by two different realizations v_t^1 and v_t^2 from the same distribution.

Derivation B.0.1 (Covariance expressions for S_N) *Given the sum Q_N which is a generalization of the structure of S_N.*

$$Q_N = \frac{1}{N}\sum_{t=1}^{N}[a(t)b(t) + c(t)d(t)] \tag{B.1}$$

where $a(t), b(t), c(t)$ and $d(t)$ are signals generated by filtering the white noise signal $e(t)$ through the stable scalar transfer functions A and C and the vectors of stable transfer functions B and D. Hence

$$a(t) = Ae(t), \quad b(t) = Be(t), \quad c(t) = Ce(t), \quad d(t) = De(t)$$

Since all signals from $a(t)$ to $d(t)$ are correlated due to $e(t)$, one obtains:

$$Cov[Q_N] = E[Q_N Q_N^T] - E[Q_N]E[Q_N]^T \tag{B.2}$$

Evaluation of the first term gives

$$\begin{aligned}
E[Q_N Q_N^T] =& E\left[\frac{1}{N^2}\sum_{t=1}^{N}[a(t)b(t)+c(t)d(t)]\left(\sum_{t=1}^{N}a(t)b(t)+c(t)d(t)\right)^T\right]\\
=&\frac{1}{N^2}E\left[\sum_{t=1}^{N}[a(t)b(t)+c(t)d(t)]\left(\sum_{s=1}^{N}a(s)b(s)+c(s)d(s)\right)^T\right]\\
=&\frac{1}{N^2}E\left[\sum_{t,s=1}^{N}a(t)b(t)a(s)b(s)^T+a(t)b(t)c(s)d(s)^T+\right.\\
&\left.c(t)d(t)a(s)b(s)^T+c(t)d(t)c(s)d(s)^T\right]\\
=&\frac{1}{N^2}\left(E[\sum_{t,s=1}^{N}a(t)b(t)a(s)b(s)^T]+E[\sum_{t,s=1}^{N}a(t)b(t)c(s)d(s)^T]+\right.\\
&\left.E[\sum_{t,s=1}^{N}c(t)d(t)a(s)b(s)^T]+E[\sum_{t,s=1}^{N}c(t)d(t)c(s)d(s)^T]\right)
\end{aligned}$$

Using the following formula which is correct for the given properties of $a(t), b(t), c(t)$ and $d(t)$ and where α, β, δ and γ are fixed delays.

$$\begin{aligned}
\overline{E}[a(t-\alpha)b(t-\beta)c(t-\gamma)d(t-\delta)] =& R_{ab}(\beta-\alpha)R_{cd^T}(\delta-\gamma)+R_{ac}(\gamma-\alpha)R_{bd^T}(\delta-\beta)+\\
&R_{bc}(\delta-\alpha)R_{ad^T}(\gamma-\beta)
\end{aligned}$$

the expression can be written as

$$\begin{aligned}
E[Q_N Q_N^T] =& \frac{1}{N^2}\left(\sum_{t,s=1}^{N}(R_{ab}(0)R_{ab^T}(0)+R_{aa}(t-s)R_{bb^T}(t-s)+R_{ba}(t-s)R_{ab^T}(t-s))+\right.\\
&\sum_{t,s=1}^{N}(R_{ab}(0)R_{cd^T}(0)+R_{ac}(t-s)R_{bd^T}(t-s)+R_{bc}(t-s)R_{ad^T}(t-s))+\\
&\sum_{t,s=1}^{N}(R_{cd}(0)R_{ab^T}(0)+R_{ca}(t-s)R_{db^T}(t-s)+R_{da}(t-s)R_{cb^T}(t-s))+\\
&\left.\sum_{t,s=1}^{N}(R_{cd}(0)R_{cd^T}(0)+R_{cc}(t-s)R_{dd^T}(t-s)+R_{dc}(t-s)R_{cd^T}(t-s))\right)
\end{aligned}$$

The second term in (B.2) *will take the same form but will only have a contribution different from zero when the lag $t-s=0$. Hence*

$$E[Q_N]E[Q_N]^T = R_{ab}(0)R_{ab^T}(0)+R_{ab}(0)R_{cd^T}(0)+R_{cd}(0)R_{ab^T}(0)+R_{cd}(0)R_{cd^T}(0)$$

which means that the covariance of Q_N simplifies to

$$\begin{aligned}
Cov[Q_N] =& \frac{1}{N^2}\sum_{t,s=1}^{N}R_{aa}(t-s)R_{bb^T}(t-s)+R_{ba}(t-s)R_{ab^T}(t-s)+\\
&R_{ac}(t-s)R_{bd^T}(t-s)+R_{bc}(t-s)R_{ad^T}(t-s)+R_{ca}(t-s)R_{db^T}(t-s)+\\
&R_{da}(t-s)R_{cb^T}(t-s)+R_{cc}(t-s)R_{dd^T}(t-s)+R_{dc}(t-s)R_{cd^T}(t-s)
\end{aligned}$$

or

$$
\begin{aligned}
Cov[Q_N] =&\frac{1}{N}\sum_{\tau=-(N-1)}^{N-1}\frac{N-|\tau|}{N}\Big(R_{aa}(\tau)R_{bb^T}(\tau)+R_{ba}(\tau)R_{ab^T}(\tau)+\\
&R_{ac}(\tau)R_{bd^T}(\tau)+R_{bc}(\tau)R_{ad^T}(\tau)+R_{ca}(\tau)R_{db^T}(\tau)+\\
&R_{da}(\tau)R_{cb^T}(\tau)+R_{cc}(\tau)R_{dd^T}(\tau)+R_{dc}(\tau)R_{cd^T}(\tau)\Big)
\end{aligned}
$$

by letting $N \to \infty$, using Kronecker's lemma and applying the formula

$$
\sum_{\tau=-\infty}^{\infty} R_{ab}(\tau)R_{cd^T}(\tau)=\frac{1}{2\pi}\int_{-\pi}^{\pi}\Phi_{ab}(\omega)\overline{\Phi}_{cd^T}(\omega)d\omega \tag{B.3}
$$

the following asymptotic expression appears

$$
\begin{aligned}
\lim_{N\to\infty} N\,Cov[Q_N] =&\frac{1}{2\pi}\int_{-\pi}^{\pi}\Phi_{aa}(\omega)\overline{\Phi}_{bb^T}(\omega)+\Phi_{ba}(\omega)\overline{\Phi}_{ab^T}(\omega)+\\
&\Phi_{ac}(\omega)\overline{\Phi}_{bd^T}(\omega)+\Phi_{bc}(\omega)\overline{\Phi}_{ad^T}(\omega)+\Phi_{ca}(\omega)\overline{\Phi}_{db^T}(\omega)+\\
&\Phi_{da}(\omega)\overline{\Phi}_{cb^T}(\omega)+\Phi_{cc}(\omega)\overline{\Phi}_{dd^T}(\omega)+\Phi_{dc}(\omega)\overline{\Phi}_{cd^T}(\omega)d\omega
\end{aligned} \tag{B.4}
$$

where $\overline{x}$ is the complex conjugate of x.

Letting A, B, C and D refer to the transfer functions in Equation (7.18)*, the cross spectra can be evaluated using Equation* (B.5) *where x^* is the complex conjugated transpose of x.*

$$
\Phi_{pq} = PQ^*\Phi_e \tag{B.5}
$$

where $p(t) = Pe(t)$ and $q(t) = Qe(t)$ is a set of signals produced by filtering the same noise signal, $e(t)$ through the stable transfer functions P and Q.

$$
\begin{aligned}
\Phi_{aa} =&|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\\
\Phi_{bb^T} =&|G(e^{j\omega})|^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{cc} =&\lambda|C(e^{j\omega},\boldsymbol{\rho}_i)|^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\\
\Phi_{dd^T} =&\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{ba} =&-G(e^{j\omega})S(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{ab^T} =&-G^*(e^{j\omega})S^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{ac} =&-\sqrt{\lambda}C^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\\
\Phi_{bd^T} =&\sqrt{\lambda}G(e^{j\omega})|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{bc} =&\sqrt{\lambda}G(e^{j\omega})S(e^{j\omega},\boldsymbol{\rho}_i)C^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{ad^T} =&-\sqrt{\lambda}S^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v
\end{aligned}
$$

$$\begin{aligned}
\Phi_{ca} =& -\sqrt{\lambda}C(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v \\
\Phi_{db^T} =& \sqrt{\lambda}G^*(e^{j\omega})|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\
\Phi_{da} =& -\sqrt{\lambda}S(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\
\Phi_{cb^T} =& \sqrt{\lambda}C(e^{j\omega},\boldsymbol{\rho}_i)G^*(e^{j\omega})S^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\
\Phi_{dc} =& \lambda S(e^{j\omega},\boldsymbol{\rho}_i)C^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\
\Phi_{cd^T} =& \lambda C(e^{j\omega},\boldsymbol{\rho}_i)S^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v
\end{aligned}$$

Hence

$$\begin{aligned}
\Phi_{aa}\overline{\Phi}_{bb^T} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^2|G(e^{j\omega})|^2\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2 \\
\Phi_{ba}\overline{\Phi}_{ab^T} =& S^2(e^{j\omega},\boldsymbol{\rho}_i)G^2(e^{j\omega})\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2 \\
\Phi_{cc}\overline{\Phi}_{dd^T} =& \lambda^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^2|C(e^{j\omega},\boldsymbol{\rho}_i)|^2\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2 \\
\Phi_{dc}\overline{\Phi}_{cd^T} =& \lambda^2S^2(e^{j\omega},\boldsymbol{\rho}_i)\overline{C}^2(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2 \\
\Phi_{ac}\overline{\Phi}_{bd^T} =& -\lambda\overline{G}(e^{j\omega})\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2 \\
\Phi_{bc}\overline{\Phi}_{ad^T} =& -\lambda G(e^{j\omega})\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)S^2(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2 \\
\Phi_{ca}\overline{\Phi}_{db^T} =& -\lambda G(e^{j\omega})\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2 \\
\Phi_{da}\overline{\Phi}_{cb^T} =& -\lambda G(e^{j\omega})\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)S^2(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2
\end{aligned}$$

Inserting these expressions in (B.4) *one obtain after a few manipulations:*

$$\begin{aligned}
\lim_{N\to\infty}N\,Cov[S_N(\boldsymbol{\rho}_i)] =& \frac{1}{2\pi}\int_{-\pi}^{\pi}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2\left[\Psi(e^{j\omega},\boldsymbol{\rho}_i)\Psi(e^{j\omega},\boldsymbol{\rho}_i)^T+\overline{\Psi}(e^{j\omega},\boldsymbol{\rho}_i)\Psi(e^{j\omega},\boldsymbol{\rho}_i)^T\right]d\omega \\
=& \frac{2}{2\pi}\int_{-\pi}^{\pi}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_v^2\left[\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T+\right. \\
& \left. j\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T\right]d\omega
\end{aligned}$$

where

$$\Psi(e^{j\omega},\boldsymbol{\rho}_i) = [G(e^{j\omega},\boldsymbol{\rho}_i)-\lambda\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)]S(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}$$

Which is the same as in (7.28) *since the integration of* $2j\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T$ *from* $-\pi$ *to* π *is zero for all* $\Psi(e^{j\omega},\boldsymbol{\rho}_i)\in\mathbb{C}$. **q.e.d.**

Derivation B.0.2 (Covariance expressions for E_N) *For the derivation of the covariance of E_N, let the sum Q_N be a generalization of the structure of E_N.*

$$Q_N = \frac{1}{N}\sum_{t=1}^{N}[a(t)b(t) + c(t)d(t)] \tag{B.6}$$

where $a(t), b(t), c(t)$ and $d(t)$ are signals generated by filtering the two uncorrelated white noise signals $e(t)$ and $f(t)$ through the stable scalar transfer functions A and C and the vectors of stable transfer functions B and D. Hence

$$a(t) = Ae(t), \quad b(t) = Bf(t), \quad c(t) = Ce(t), \quad d(t) = Df(t)$$

Using the same derivation as above, but realizing that the cross correlation terms between signals driven by uncorrelated noise realizations will be zero, the covariance can be written as

$$\begin{aligned} Cov[Q_N] =& \frac{1}{N^2}\sum_{t,s=1}^{N} R_{aa}(t-s)R_{bb^T}(t-s) + R_{ac}(t-s)R_{bd^T}(t-s) + \\ & R_{ca}(t-s)R_{db^T}(t-s) + R_{cc}(t-s)R_{dd^T}(t-s) \end{aligned}$$

hence

$$\begin{aligned} \lim_{N\to\infty} N\,Cov[Q_N] =& \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_{aa}(\omega)\overline{\Phi}_{bb^T}(\omega) + \Phi_{ac}(\omega)\overline{\Phi}_{bd^T}(\omega) + \\ & \Phi_{ca}(\omega)\overline{\Phi}_{db^T}(\omega) + \Phi_{cc}(\omega)\overline{\Phi}_{dd^T}(\omega)d\omega \end{aligned} \tag{B.7}$$

When A, B, C and D refers to transfer functions in Equation (7.18) the cross spectra are

$$\begin{aligned} \Phi_{aa} =& |S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \Phi_v \\ \Phi_{cc} =& \lambda |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \Phi_v \\ \Phi_{ac} =& -\sqrt{\lambda}\overline{C}(e^{j\omega}, \boldsymbol{\rho}_i)|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \Phi_v \\ \Phi_{ca} =& -\sqrt{\lambda}C(e^{j\omega}, \boldsymbol{\rho}_i)|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \Phi_v \\ \Phi_{bb^T} =& |S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \frac{\partial C^*(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \Phi_v \\ \Phi_{dd^T} =& \lambda |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \frac{\partial C^*(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \Phi_v \\ \Phi_{bd^T} =& -\sqrt{\lambda}\overline{C}(e^{j\omega}, \boldsymbol{\rho}_i)|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \frac{\partial C^*(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \Phi_v \\ \Phi_{db^T} =& -\sqrt{\lambda}C(e^{j\omega}, \boldsymbol{\rho}_i)|S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \frac{\partial C^*(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \Phi_v \end{aligned}$$

Hence

$$\begin{aligned} \Phi_{aa}\overline{\Phi}_{bb^T} =& |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\ \Phi_{cc}\overline{\Phi}_{dd^T} =& \lambda^2 |C(e^{j\omega}, \boldsymbol{\rho}_i)|^4 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\ \Phi_{ac}\overline{\Phi}_{bd^T} =& \lambda |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\ \Phi_{ca}\overline{\Phi}_{db^T} =& \Phi_{ac}\overline{\Phi}_{bd^T} \end{aligned}$$

Inserting these expressions in (B.7) *gives:*

$$\lim_{N\to\infty} N\,Cov[E_N(\boldsymbol{\rho}_i)] = \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \left[1+\lambda|C(e^{j\omega},\boldsymbol{\rho}_i)|^2\right]^2 \times \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T \Phi_v^2 d\omega$$

Since the result is a real number, this expression is the same as (7.28) *where the integrand is the complex conjugate of the expression above.* **q.e.d.**

C

Derivation of covariance expression for the cost function estimate $\hat{F}(\boldsymbol{\rho})$ for Iterative Feedback Tuning with external perturbation.

In this section the asymptotic covariance expression for the performance cost function estimate will be derived for the case where the system is perturbed. As argued in Section 7.3 the covariance expression still consists only of the sum of the asymptotic covariances for the sums S_N and E_N which reflect the deterministic and the variance part of the gradient estimate respectively. These two sums are given by:

$$\begin{aligned}
S_N(\boldsymbol{\rho}_i) =& \frac{1}{N}\sum_{t=1}^{N}\Bigg[\big(S(\boldsymbol{\rho}_i)(Gr_t^p+v_t^1)\big)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}GS(\boldsymbol{\rho}_i)^2(Gr_t^p+v_t^1)\right)+\\
&\lambda S(\boldsymbol{\rho}_i)(r_t^p-C(\boldsymbol{\rho}_i)v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho}_i)^2(Gr_t^p+v_t^1)\right)\Bigg]\\
E_N(\boldsymbol{\rho}_i) =& \frac{1}{N}\sum_{t=1}^{N}\Bigg[\big(S(\boldsymbol{\rho}_i)(Gr_t^p+v_t^1)\big)\left(\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}S(\boldsymbol{\rho}_i)v_t^2\right)+\\
&\lambda S(\boldsymbol{\rho}_i)(r_t^p-C(\boldsymbol{\rho}_i)v_t^1)\left(-\frac{\partial C(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}C(\boldsymbol{\rho}_i)S(\boldsymbol{\rho}_i)v_t^2\right)\Bigg]
\end{aligned}$$

In the following derivations r_t^p will be regarded as a signal driven by a white noise process with the same distribution as v_t^1 and v_t^2. r_t^p can be regarded as a deterministic signal which will be reused in every iteration of the Iterative Feedback tuning or alternatively a new signal could be generated for each iteration. The latter option will be assumed in the following. Hence S_N consists of eight signals arranged in two sets of four driven by r_t^p and v_t^1. E_N consists of six signals collected in three pairs each driven by r_t^p, v_t^1 or v_t^2.

Derivation C.0.3 (Covariance expressions for S_N) *For the derivation of the covariance of S_N, let the sum Q_N be a generalization of the structure of S_N.*

$$\begin{aligned}
Q_N =& \frac{1}{N}\sum_{t=1}^{N}[(a1(t)+b1(t))(a2(t)+b2(t))+(a3(t)+b3(t))(a4(t)+b4(t))] \\
=& \frac{1}{N}\sum_{t=1}^{N}[a1(t)a2(t)+b1(t)b2(t)+a1(t)b2(t)+b1(t)a2(t)+ \\
& a3(t)a4(t)+b3(t)b4(t)+a3(t)b4(t)+b3(t)a4(t)]
\end{aligned}$$

where $ai(t)$ and $bi(t)$, $i \in \{1,2,3,4\}$ are signals generated by filtering the two uncorrelated white noise signals $e(t)$ and $f(t)$ through the stable scalar filters $A1, B1, A3, B3$ and the vectors of stable filters $A2, B2, A4, B4$.

$$\begin{array}{llll}
a1(t) = A1e(t), & a2(t) = A2e(t), & a3(t) = A3e(t), & a4(t) = A4e(t) \\
b1(t) = B1f(t), & b2(t) = B2f(t), & b3(t) = B3f(t), & b4(t) = B4f(t)
\end{array}$$

Using (B.2) *and evaluating the terms yields the following sum of cross correlation functions:*

$$\begin{aligned}
Cov[Q_N] =& \frac{1}{N^2}\sum_{t,s=1}^{N} R_{a1a1}(t-s)R_{a2a2^T}(t-s)+R_{a2a1}(t-s)R_{a1a2^T}(t-s)+ \\
& R_{a1a3}(t-s)R_{a2a4^T}(t-s)+R_{a2a3}(t-s)R_{a1a4^T}(t-s)+R_{b1b1}(t-s)R_{b2b2^T}(t-s)+ \\
& R_{b2b1}(t-s)R_{b1b2^T}(t-s)+R_{b1b3}(t-s)R_{b2b4^T}(t-s)+R_{b2b3}(t-s)R_{b1b4^T}(t-s)+ \\
& R_{a3a1}(t-s)R_{a4a2^T}(t-s)+R_{a4a1}(t-s)R_{a3a2^T}(t-s)+R_{a3a3}(t-s)R_{a4a4^T}(t-s)+ \\
& R_{a4a3}(t-s)R_{a3a4^T}(t-s)+R_{b3b1}(t-s)R_{b4b2^T}(t-s)+R_{b4b1}(t-s)R_{b3b2^T}(t-s)+ \\
& R_{b3b3}(t-s)R_{b4b4^T}(t-s)+R_{b4b3}(t-s)R_{b3b4^T}(t-s)+R_{a1a1}(t-s)R_{b2b2^T}(t-s)+ \\
& R_{a1a2}(t-s)R_{b2b1^T}(t-s)+R_{a1a3}(t-s)R_{b2b4^T}(t-s)+R_{a1a4}(t-s)R_{b2b3^T}(t-s)+ \\
& R_{a2a1}(t-s)R_{b1b2^T}(t-s)+R_{a2a2}(t-s)R_{b1b1^T}(t-s)+R_{a2a3}(t-s)R_{b1b4^T}(t-s)+ \\
& R_{a2a4}(t-s)R_{b1b3^T}(t-s)+R_{a3a1}(t-s)R_{b4b2^T}(t-s)+R_{a3a2}(t-s)R_{b4b1^T}(t-s)+ \\
& R_{a3a3}(t-s)R_{b4b4^T}(t-s)+R_{a3a4}(t-s)R_{b4b3^T}(t-s)+R_{a4a1}(t-s)R_{b3b2^T}(t-s)+ \\
& R_{a4a2}(t-s)R_{b3b1^T}(t-s)+R_{a4a3}(t-s)R_{b3b4^T}(t-s)+R_{a4a4}(t-s)R_{b3b3^T}(t-s)
\end{aligned}$$

Hence

$$\begin{aligned}
\lim_{N\to\infty} N\,Cov[Q_N] =& \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_{a1a1}(\omega)\overline{\Phi}_{a2a2^T}(\omega)+\Phi_{a2a1}(\omega)\overline{\Phi}_{a1a2^T}(\omega)+ \\
& \Phi_{a1a3}(\omega)\overline{\Phi}_{a2a4^T}(\omega)+\Phi_{a2a3}(\omega)\overline{\Phi}_{a1a4^T}(\omega)+\Phi_{b1b1}(\omega)\overline{\Phi}_{b2b2^T}(\omega)+ \\
& \Phi_{b2b1}(\omega)\overline{\Phi}_{b1b2^T}(\omega)+\Phi_{b1b3}(\omega)\overline{\Phi}_{b2b4^T}(\omega)+\Phi_{b2b3}(\omega)\overline{\Phi}_{b1b4^T}(\omega)+ \\
& \Phi_{a3a1}(\omega)\overline{\Phi}_{a4a2^T}(\omega)+\Phi_{a4a1}(\omega)\overline{\Phi}_{a3a2^T}(\omega)+\Phi_{a3a3}(\omega)\overline{\Phi}_{a4a4^T}(\omega)+ \\
& \Phi_{a4a3}(\omega)\overline{\Phi}_{a3a4^T}(\omega)+\Phi_{b3b1}(\omega)\overline{\Phi}_{b4b2^T}(\omega)+\Phi_{b4b1}(\omega)\overline{\Phi}_{b3b2^T}(\omega)+ \\
& \Phi_{b3b3}(\omega)\overline{\Phi}_{b4b4^T}(\omega)+\Phi_{b4b3}(\omega)\overline{\Phi}_{b3b4^T}(\omega)+\Phi_{a1a1}(\omega)\overline{\Phi}_{b2b2^T}(\omega)+ \\
& \Phi_{a1a2}(\omega)\overline{\Phi}_{b2b1^T}(\omega)+\Phi_{a1a3}(\omega)\overline{\Phi}_{b2b4^T}(\omega)+\Phi_{a1a4}(\omega)\overline{\Phi}_{b2b3^T}(\omega)+ \\
& \Phi_{a2a1}(\omega)\overline{\Phi}_{b1b2^T}(\omega)+\Phi_{a2a2}(\omega)\overline{\Phi}_{b1b1^T}(\omega)+\Phi_{a2a3}(\omega)\overline{\Phi}_{b1b4^T}(\omega)+ \\
& \Phi_{a2a4}(\omega)\overline{\Phi}_{b1b3^T}(\omega)+\Phi_{a3a1}(\omega)\overline{\Phi}_{b4b2^T}(\omega)+\Phi_{a3a2}(\omega)\overline{\Phi}_{b4b1^T}(\omega)+ \\
& \Phi_{a3a3}(\omega)\overline{\Phi}_{b4b4^T}(\omega)+\Phi_{a3a4}(\omega)\overline{\Phi}_{b4b3^T}(\omega)+\Phi_{a4a1}(\omega)\overline{\Phi}_{b3b2^T}(\omega)+ \\
& \Phi_{a4a2}(\omega)\overline{\Phi}_{b3b1^T}(\omega)+\Phi_{a4a3}(\omega)\overline{\Phi}_{b3b4^T}(\omega)+\Phi_{a4a4}(\omega)\overline{\Phi}_{b3b3^T}(\omega)d\omega \quad \text{(C.1)}
\end{aligned}$$

When Ai, Bi, $i \in \{1,2,3,4\}$ refers to transfer functions in Equation (7.29) the cross spectra are

$$
\begin{aligned}
\Phi_{a1a1} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^2|G(e^{j\omega})|^2\Phi_{r^p}\\
\Phi_{a2a2} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^4\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\\
\Phi_{a3a3} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_{r^p}\\
\Phi_{a4a4} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\\
\Phi_{a1a2} =& -|S(e^{j\omega},\boldsymbol{\rho}_i)|^2|G(e^{j\omega})|^2\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega})\right)^*\Phi_{r^p}\\
\Phi_{a2a1} =& -|S(e^{j\omega},\boldsymbol{\rho}_i)|^2|G(e^{j\omega})|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega})\Phi_{r^p}\\
\Phi_{a1a3} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2G(e^{j\omega})\Phi_{r^p}\\
\Phi_{a3a1} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\overline{G}(e^{j\omega})\Phi_{r^p}\\
\Phi_{a1a4} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2|G(e^{j\omega})|^2\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)\right)^*\Phi_{r^p}\\
\Phi_{a4a1} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2|G(e^{j\omega})|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)\Phi_{r^p}\\
\Phi_{a2a3} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}G(e^{j\omega})^2S(e^{j\omega},\boldsymbol{\rho}_i)\Phi_{r^p}\\
\Phi_{a3a2} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}G(e^{j\omega})^2S(e^{j\omega},\boldsymbol{\rho}_i)\right)^*\Phi_{r^p}\\
\Phi_{a2a4} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2G(e^{j\omega})\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\\
\Phi_{a4a2} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2\overline{G}(e^{j\omega})\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\\
\Phi_{a3a4} =& -\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega})\right)^*\Phi_{r^p}\\
\Phi_{a4a3} =& -\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega})\Phi_{r^p}\\
\Phi_{b1b1} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\\
\Phi_{b2b2} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{b3b3} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2|C(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v\\
\Phi_{b4b4} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v\\
\Phi_{b1b2} =& -|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega})\right)^*\Phi_v\\
\Phi_{b2b1} =& -|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega})\Phi_v\\
\Phi_{b1b3} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)\Phi_v\\
\Phi_{b3b1} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2C(e^{j\omega},\boldsymbol{\rho}_i)\Phi_v
\end{aligned}
$$

$$
\begin{aligned}
\Phi_{b1b4} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)\right)^*\Phi_v \\
\Phi_{b4b1} =& -\sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)\Phi_v \\
\Phi_{b2b3} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}G(e^{j\omega})^2S(e^{j\omega},\boldsymbol{\rho}_i)\Phi_v \\
\Phi_{b3b2} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^2C(e^{j\omega},\boldsymbol{\rho}_i)\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}G(e^{j\omega})^2S(e^{j\omega},\boldsymbol{\rho}_i)\right)^*\Phi_v \\
\Phi_{b2b4} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4G(e^{j\omega})\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\
\Phi_{b4b2} =& \sqrt{\lambda}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\overline{G}(e^{j\omega})\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\
\Phi_{b3b4} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2C(e^{j\omega},\boldsymbol{\rho}_i)\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)\right)^*\Phi_v \\
\Phi_{b4b3} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}S(e^{j\omega},\boldsymbol{\rho}_i)\Phi_v
\end{aligned}
$$

Evaluating the multiplication of the cross spectra in (C.1)*, it is evident that the terms can be divided into four groups. The terms in these four sub-groups are evaluated separately and summed.*

In the following two complex functions are utilized

$$
\Psi(e^{j\omega},\boldsymbol{\rho}_i) = [G(e^{j\omega},\boldsymbol{\rho}_i)-\lambda\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)]S(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}} \tag{C.2}
$$

$$
\Upsilon(e^{j\omega},\boldsymbol{\rho}_i) = [|G(e^{j\omega},\boldsymbol{\rho}_i)|^2+\lambda]S(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}} \tag{C.3}
$$

$$
\begin{aligned}
\Phi_{a1a1}\overline{\Phi}_{a2a2^T} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^6\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2 \\
\Phi_{a2a1}\overline{\Phi}_{a1a2^T} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^4S(e^{j\omega},\boldsymbol{\rho}_i)^2G(e^{j\omega})^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2 \\
\Phi_{a1a3}\overline{\Phi}_{a2a4^T} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^4\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2 \\
\Phi_{a2a3}\overline{\Phi}_{a1a4^T} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2S(e^{j\omega},\boldsymbol{\rho}_i)^2G(e^{j\omega})^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2 \\
\Phi_{a3a1}\overline{\Phi}_{a4a2^T} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^4\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2 \\
\Phi_{a4a1}\overline{\Phi}_{a3a2^T} =& \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2S(e^{j\omega},\boldsymbol{\rho}_i)^2G(e^{j\omega})^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2 \\
\Phi_{a3a3}\overline{\Phi}_{a4a4^T} =& \lambda^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^2\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2 \\
\Phi_{a4a3}\overline{\Phi}_{a3a4^T} =& \lambda^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^4S(e^{j\omega},\boldsymbol{\rho}_i)^2G(e^{j\omega})^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{rp}^2
\end{aligned}
$$

the sum of which yields

$$\begin{aligned}\lim_{N\to\infty} N\,Cov[Q_N^1] =& \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \Phi_{r^p}^2 \times \\ & \left[G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)(G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))^T + \overline{(G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))}(G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))^T\right] d\omega \\ =& \frac{2}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \left[\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T\right] \Phi_{r^p}^2 d\omega\end{aligned}$$

$$\begin{aligned}
\Phi_{b1b1}\overline{\Phi}_{b2b2^T} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^6 |G(e^{j\omega})|^2 \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\
\Phi_{b2b1}\overline{\Phi}_{b1b2^T} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 S(e^{j\omega},\boldsymbol{\rho}_i)^2 G(e^{j\omega})^2 \frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\
\Phi_{b1b3}\overline{\Phi}_{b2b4^T} =& -\lambda |S(e^{j\omega},\boldsymbol{\rho}_i)|^6 \overline{G}(e^{j\omega})\overline{C}(e^{j\omega},\boldsymbol{\rho}_i) \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\
\Phi_{b2b3}\overline{\Phi}_{b1b4^T} =& -\lambda |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 G(e^{j\omega}) S(e^{j\omega},\boldsymbol{\rho}_i)^2 \overline{C}(e^{j\omega},\boldsymbol{\rho}_i) \frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\
\Phi_{b3b1}\overline{\Phi}_{b4b2^T} =& -\lambda |S(e^{j\omega},\boldsymbol{\rho}_i)|^6 G(e^{j\omega}) C(e^{j\omega},\boldsymbol{\rho}_i) \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\
\Phi_{b4b1}\overline{\Phi}_{b3b2^T} =& -\lambda |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 G(e^{j\omega}) S(e^{j\omega},\boldsymbol{\rho}_i)^2 \overline{C}(e^{j\omega},\boldsymbol{\rho}_i) \frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\
\Phi_{b3b3}\overline{\Phi}_{b4b4^T} =& \lambda^2 |S(e^{j\omega},\boldsymbol{\rho}_i)|^6 |C(e^{j\omega},\boldsymbol{\rho}_i)|^2 \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2 \\
\Phi_{b4b3}\overline{\Phi}_{b3b4^T} =& \lambda^2 |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 S(e^{j\omega},\boldsymbol{\rho}_i)^2 \overline{C}(e^{j\omega},\boldsymbol{\rho}_i)^2 \frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_v^2
\end{aligned}$$

the sum of which yields

$$\begin{aligned}\lim_{N\to\infty} N\,Cov[Q_N^2] =& \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \Phi_v^2 \times \\ & \left[\Psi(e^{j\omega},\boldsymbol{\rho}_i)\Psi(e^{j\omega},\boldsymbol{\rho}_i)^T + \overline{\Psi}(e^{j\omega},\boldsymbol{\rho}_i)\Psi(e^{j\omega},\boldsymbol{\rho}_i)^T\right] d\omega \\ =& \frac{2}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \left[\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T\right] \Phi_v^2 d\omega\end{aligned}$$

$$\begin{aligned}
\Phi_{a1a1}\overline{\Phi}_{b2b2^T} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^6 |G(e^{j\omega})|^4 \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_{r^p}\Phi_v \\
\Phi_{a1a2}\overline{\Phi}_{b2b1^T} =& |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 |G(e^{j\omega})|^2 (\overline{S}(e^{j\omega},\boldsymbol{\rho}_i)\overline{G}(e^{j\omega}))^2 \frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \Phi_{r^p}\Phi_v \\
\Phi_{a1a3}\overline{\Phi}_{b2b4^T} =& \lambda |S(e^{j\omega},\boldsymbol{\rho}_i)|^6 |G(e^{j\omega})|^2 \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}\right)^T \Phi_{r^p}\Phi_v \\
\Phi_{a1a4}\overline{\Phi}_{b2b3^T} =& -\lambda |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 |G(e^{j\omega})|^2 \overline{S}(e^{j\omega},\boldsymbol{\rho}_i)^2 \overline{G}(e^{j\omega}) C(e^{j\omega},\boldsymbol{\rho}_i) \frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \Phi_{r^p}\Phi_v
\end{aligned}$$

$$
\begin{aligned}
\Phi_{a3a1}\overline{\Phi}_{b4b2^T} =&\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^2\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{r^p}\Phi_v\\
\Phi_{a3a2}\overline{\Phi}_{b4b1^T} =&\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4(\overline{S}(e^{j\omega},\boldsymbol{\rho}_i)\overline{G}(e^{j\omega}))^2\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\Phi_{r^p}\Phi_v\\
\Phi_{a3a3}\overline{\Phi}_{b4b4^T} =&\lambda^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^6\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{r^p}\Phi_v\\
\Phi_{a3a4}\overline{\Phi}_{b4b3^T} =&-\lambda^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\overline{S}(e^{j\omega},\boldsymbol{\rho}_i)^2\overline{G}(e^{j\omega})C(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\overline{\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}}\Phi_{r^p}\Phi_v
\end{aligned}
$$

the sum of which yields

$$
\begin{aligned}
\lim_{N\to\infty} N\,Cov[Q_N^3] =&\frac{1}{2\pi}\int_{-\pi}^{\pi}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\Phi_{r^p}\Phi_v\times\\
&\left[G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)(\Psi(e^{j\omega},\boldsymbol{\rho}_i))^T+\overline{\Upsilon}(e^{j\omega},\boldsymbol{\rho}_i)\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)^T\right]d\omega\\
=&\frac{1}{2\pi}\int_{-\pi}^{\pi}|S(e^{j\omega},\boldsymbol{\rho}_i)|^4\times\\
&\Big[\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T+\\
&\mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T+\\
&\mathcal{R}e\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T+\\
&\mathcal{I}m\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T\Big]\Phi_{r^p}\Phi_v d\omega
\end{aligned}
$$

$$
\begin{aligned}
\Phi_{a2a1}\overline{\Phi}_{b1b2^T} =&|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2(S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega}))^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{r^p}\Phi_v\\
\Phi_{a2a2}\overline{\Phi}_{b1b1^T} =&|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^2(\overline{S}(e^{j\omega},\boldsymbol{\rho}_i)\overline{G}(e^{j\omega}))^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\Phi_v\\
\Phi_{a2a3}\overline{\Phi}_{b1b4^T} =&\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2(S(e^{j\omega},\boldsymbol{\rho}_i)G(e^{j\omega}))^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{r^p}\Phi_v\\
\Phi_{a2a4}\overline{\Phi}_{b1b3^T} =&-\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^2S(e^{j\omega},\boldsymbol{\rho}_i)C(e^{j\omega,\boldsymbol{\rho}_i})\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\Phi_v\\
\Phi_{a4a1}\overline{\Phi}_{b3b2^T} =&-\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^4|G(e^{j\omega})|^2S(e^{j\omega},\boldsymbol{\rho}_i)^2G(e^{j\omega})\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{r^p}\Phi_v\\
\Phi_{a4a2}\overline{\Phi}_{b3b1^T} =&-\lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^2\overline{G}(e^{j\omega}))^2\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\Phi_v\\
\Phi_{a4a3}\overline{\Phi}_{b3b4^T} =&-\lambda^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^4S(e^{j\omega},\boldsymbol{\rho}_i)^2G(e^{j\omega})\overline{C}(e^{j\omega},\boldsymbol{\rho}_i)\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\left(\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\right)^T\Phi_{r^p}\Phi_v\\
\Phi_{a4a4}\overline{\Phi}_{b3b3^T} =&\lambda^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^6|G(e^{j\omega})|^2|C(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_{r^p}\Phi_v
\end{aligned}
$$

the sum of which yields

$$\begin{aligned}
\lim_{N\to\infty} N\,Cov[Q_N^4] =& \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \Phi_{r^p}\Phi_v \times \\
& \left[G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)(G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i))^* + (G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))^*\overline{\Psi}(e^{j\omega},\boldsymbol{\rho}_i)\right] d\omega \\
=& \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \times \\
& \Big[\mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T + \\
& \mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T + \\
& \mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\} - \\
& \mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\Big] \Phi_{r^p}\Phi_v d\omega
\end{aligned}$$

Combining these four terms gives the covariance expression for S_N.

$$\begin{aligned}
\lim_{N\to\infty} N\,Cov[S_N(\boldsymbol{\rho}_i)] = & \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \times [\\
& \left[G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)(G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))^T + \overline{(G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))}(G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))^T\right] \Phi_{r^p}^2 + \\
& \left[\Psi(e^{j\omega},\boldsymbol{\rho}_i)\Psi(e^{j\omega},\boldsymbol{\rho}_i)^T + \overline{\Psi}(e^{j\omega},\boldsymbol{\rho}_i)\Psi(e^{j\omega},\boldsymbol{\rho}_i)^T\right] \Phi_v^2 + \\
& \left[G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)(\Psi(e^{j\omega},\boldsymbol{\rho}_i))^T + \overline{\Upsilon}(e^{j\omega},\boldsymbol{\rho}_i)\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)^T\right] \Phi_{r^p}\Phi_v + \\
& \left[G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)(G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i))^* + (G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i))^*\overline{\Psi}(e^{j\omega},\boldsymbol{\rho}_i)\right] \Phi_{r^p}\Phi_v \;] \, d\omega
\end{aligned}$$

or

$$\begin{aligned}
\lim_{N\to\infty} N\,Cov[S_N(\boldsymbol{\rho}_i)] = & \frac{1}{2\pi}\int_{-\pi}^{\pi} |S(e^{j\omega},\boldsymbol{\rho}_i)|^4 \times \Bigg[\\
& \left[\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T\right] \Phi_{r^p}^2 + \\
& \left[\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T\right] \Phi_v^2 + \\
& \Big[2\mathcal{R}e\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T + \mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T - \\
& \mathcal{I}m\{G(e^{j\omega})\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T\mathcal{I}m\{\Psi(e^{j\omega},\boldsymbol{\rho}_i)\} + \mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T + \\
& \mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{G(e^{j\omega})\Psi(e^{j\omega},\boldsymbol{\rho}_i)\}^T + \mathcal{R}e\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{R}e\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T + \\
& \mathcal{I}m\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}\mathcal{I}m\{\Upsilon(e^{j\omega},\boldsymbol{\rho}_i)\}^T\Big] \Phi_{r^p}\Phi_v \Bigg] d\omega
\end{aligned}$$

Which is the shortest possible representation of the asymptotic covariance of S_N from (7.29). **q.e.d.**

Derivation C.0.4 (Covariance expressions for E_N) *For the derivation of the covariance of E_N, let the sum Q_N be a generalization of the structure of E_N.*

$$\begin{aligned}
Q_N =& \frac{1}{N}\sum_{t=1}^{N} [(a1(t) + b1(t))c1(t) + (a2(t) + b2(t))c2(t)] \\
=& \frac{1}{N}\sum_{t=1}^{N} [a1(t)c1(t) + b1(t)c1(t) + a2(t)c2(t) + b2(t)c2(t)]
\end{aligned}$$

where $ai(t)$, $bi(t)$ and $ci(t)$, $i \in \{1,2\}$ are signals generated by filtering the three uncorrelated white noise signals $e(t)$, $f(t)$ and $g(t)$ through the stable scalar filters $A1, A2, B1, B2$ and the vectors of stable filters $C1, C2$.

$$\begin{aligned} a1(t) &= A1e(t), & a2(t) &= A2e(t) \\ b1(t) &= B1f(t), & b2(t) &= B2f(t) \\ c1(t) &= C1g(t), & c2(t) &= C2g(t) \end{aligned}$$

Using (B.2)*, evaluating the terms and realizing that the cross correlation function between uncorrelated signals is zero yields:*

$$\begin{aligned} Cov[Q_N] =& \frac{1}{N^2}\sum_{t,s=1}^{N} R_{a1a1}(t-s)R_{c1c1^T}(t-s) + R_{a1a2}(t-s)R_{c1c2^T}(t-s) + \\ & R_{b1b1}(t-s)R_{c1c1^T}(t-s) + R_{b1b2}(t-s)R_{c1c2^T}(t-s) + R_{a2a1}(t-s)R_{c2c1^T}(t-s) + \\ & R_{a2a2}(t-s)R_{c2c2^T}(t-s) + R_{b2b1}(t-s)R_{c2c1^T}(t-s) + R_{b2b2}(t-s)R_{c2c2^T}(t-s) \end{aligned}$$

Hence

$$\begin{aligned} \lim_{N\to\infty} N\,Cov[Q_N] =& \frac{1}{2\pi}\int_{-\pi}^{\pi} \Phi_{a1a1}(\omega)\overline{\Phi}_{c1c1^T}(\omega) + \Phi_{a1a2}(\omega)\overline{\Phi}_{c1c2^T}(\omega) + \\ & \Phi_{b1b1}(\omega)\overline{\Phi}_{c1c1^T}(\omega) + \Phi_{b1b2}(\omega)\overline{\Phi}_{c1c2^T}(\omega) + \Phi_{a2a1}(\omega)\overline{\Phi}_{c2c1^T}(\omega) + \\ & \Phi_{a2a2}(\omega)\overline{\Phi}_{c2c2^T}(\omega) + \Phi_{b2b1}(\omega)\overline{\Phi}_{c2c1^T}(\omega) + \Phi_{b2b2}(\omega)\overline{\Phi}_{c2c2^T}(\omega)d\omega \end{aligned} \tag{C.4}$$

When $A1, A2, B1, B2, C1$ and $C2$ refer to transfer functions in Equation (7.30) *the cross spectra are*

$$\begin{aligned} \Phi_{a1a1} &= |S(e^{j\omega},\boldsymbol{\rho}_i)|^2|G(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_{r^p} \\ \Phi_{a2a2} &= \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_{r^p} \\ \Phi_{a1a2} &= \sqrt{\lambda}G(e^{j\omega})|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_{r^p} \\ \Phi_{a2a1} &= \sqrt{\lambda}G^*(e^{j\omega})|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_{r^p} \\ \Phi_{b1b1} &= |S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v \\ \Phi_{b1b2} &= -\sqrt{\lambda}C^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v \\ \Phi_{b2b2} &= \lambda|S(e^{j\omega},\boldsymbol{\rho}_i)|^2|C(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v \\ \Phi_{b2b1} &= -\sqrt{\lambda}C(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\Phi_v \\ \Phi_{c1c1^T} &= |S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\ \Phi_{c2c2^T} &= \lambda|C(e^{j\omega},\boldsymbol{\rho}_i)|^2|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\ \Phi_{c1c2^T} &= -\sqrt{\lambda}C^*(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \\ \Phi_{c2c1^T} &= -\sqrt{\lambda}C(e^{j\omega},\boldsymbol{\rho}_i)|S(e^{j\omega},\boldsymbol{\rho}_i)|^2\frac{\partial C(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\frac{\partial C^*(e^{j\omega},\boldsymbol{\rho}_i)}{\partial\boldsymbol{\rho}}\Phi_v \end{aligned}$$

Hence

$$\Phi_{a1a1}\overline{\Phi}_{c1c1^T} = |G(e^{j\omega})|^2 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_{r^p} \Phi_v$$

$$\Phi_{a2a2}\overline{\Phi}_{c2c2^T} = \lambda^2 |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^2 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_{r^p} \Phi_v$$

$$\Phi_{a1a2}\overline{\Phi}_{c2c1^T} = -\lambda G(e^{j\omega}) C(e^{j\omega}, \boldsymbol{\rho}_i) |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_{r^p} \Phi_v$$

$$\Phi_{a2a1}\overline{\Phi}_{c1c2^T} = -\lambda G^*(e^{j\omega}) \overline{C}(e^{j\omega}, \boldsymbol{\rho}_i) |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_{r^p} \Phi_v$$

$$\Phi_{b1b1}\overline{\Phi}_{c1c1^T} = |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_v^2$$

$$\Phi_{b2b2}\overline{\Phi}_{c2c2^T} = \lambda^2 |C(e^{j\omega}, \boldsymbol{\rho}_i)|^4 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_v^2$$

$$\Phi_{b1b2}\overline{\Phi}_{c2c1^T} = \lambda |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2 |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_v^2$$

$$\Phi_{b2b1}\overline{\Phi}_{c1c2^T} = \Phi_{b1b2}\overline{\Phi}_{c2c1^T}$$

Inserting these expressions in (C.4) *gives:*

$$\begin{aligned}\lim_{N\to\infty} N\,Cov[E_N(\boldsymbol{\rho}_i)] =& \frac{1}{2\pi} \int_{-\pi}^{\pi} |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \left[1 + \lambda |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2\right]^2 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_v^2 + \\ & |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \left[\overline{\Psi(e^{j\omega}, \boldsymbol{\rho}_i))} \Psi(e^{j\omega}, \boldsymbol{\rho}_i))^T \right] \Phi_{r^p} \Phi_v d\omega\end{aligned}$$

$$\begin{aligned}\lim_{N\to\infty} N\,Cov[E_N(\boldsymbol{\rho}_i)] =& \frac{1}{2\pi} \int_{-\pi}^{\pi} |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \left[1 + \lambda |C(e^{j\omega}, \boldsymbol{\rho}_i)|^2\right]^2 \overline{\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}}} \left(\frac{\partial C(e^{j\omega}, \boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \right)^T \Phi_v^2 + \\ & \left[\mathcal{R}e\{\Psi(e^{j\omega}, \boldsymbol{\rho}_i))\} \mathcal{R}e\{\Psi(e^{j\omega}, \boldsymbol{\rho}_i))\}^T + \mathcal{I}m\{\Psi(e^{j\omega}, \boldsymbol{\rho}_i))\} \mathcal{I}m\{\Psi(e^{j\omega}, \boldsymbol{\rho}_i))\}^T \right] \times \\ & |S(e^{j\omega}, \boldsymbol{\rho}_i)|^4 \Phi_{r^p} \Phi_v d\omega\end{aligned}$$

This is the shortest possible representation of the asymptotic covariance of E_N *from* (7.30).
q.e.d.

References

Åkeson, B. M.; Jørgensen, J. B.; Poulsen, N. K. and Jørgensen, S. B. (2008). A generalized autocovariance least-squares method for Kalman filter tuning. *Journal of Process Control*, **18**, 769 – 779.

Andersen, H. W.; Rasmussen, K. H. and Jørgensen, S. B. (1991). Advances in process identification. In Y. Arkun and W. H. Ray, editors, *Proceeding of the Fourth International Conference on Chemical Process Control*, pages 237 – 269.

Anderson, B. D. O. and Moore, J. B. (1989). *Optimal Control. Linear Quadratic Methods*. Prentice-Hall Int. Ed.

Andrade, P. S. (2007). *Optimizing Plantwide Control*. Master's thesis, Dept. of Chemical Eng.; Technical University of Denmark.

Ansay, P.; Gevers, M. and Wertz, V. (1999). Identification with the Youla parameterization in identification for control. In *Proceedings of the American Control Conference*, pages 827–831.

Åström, K. J. (1970). *Introduction to Stochastic Control Theory*. Academic Press.

Åström, K. J. and Hägglund, T. (1995). *PID Controllers: Theory, Design and Tuning*. Instrument Society of America, Research Triangle Park, North Carolina, second edition.

Åström, K. J. and Nilsson, J. (1994). Analysis of a scheme for iterated identification and control. In *Proceedings of SYSID94*.

Bombois, X.; Scorletti, G.; Gevers, M.; Hildebrand, R. and van den Hof, P. (2004). Cheapest open-loop identification for control. In *Proceedings of the 34rd IEEE Conference on Decision and Control*, pages 382 – 387.

Box, G. E. P. and Jenkins, G. M. (1970). *Time Series Analysis forecasting and control.*. Holden-Day.

Byrnes, C.; Isidori, A. and Willems, J. (1991). Passivity, Feedback Equivalence, and the Global Stabilization of Minimum Phase Nonlinear System. *IEEE Transactions on Automatic Control*, **36**(11), 1228–1240.

Campi, M. C. and Savareci, S. M. (2006). Direct Nonlinear Control Design: The Virtual Reference Feedback Tuning (VRFT) Approach. *IEEE transactions on automatic control*, **51**(1), 14 – 27.

Campi, M. C.; Lecchini, A. and Savaresi, S. M. (2000). Virtual Reference Feedback Tuning (VRFT): a new direct approach to the design of feedback controllers. In *Proceedings for 39'th IEEE Conference on Decision and Control*, pages 623 – 628.

Campi, M. C.; Lecchini, A. and Savaresi, S. M. (2002). Virtual reference feedback tuning: a direct method for the design of feedback controllers. *Automatica*, **28**, 1337 – 1346.

Chen, C.-T. (1993). *Analog & Digital Control System Design: Transfer-Function, State Space, and Algebraic Methods*. Saunders College Publishing.

Chen, D. and Seborg, D. E. (2002). PI/PID Controller Design Based on Direct Synthesis and Disturbance Rejection. *Ind. Eng. Chem. Res.*, **41**(19), 4807–4822.

Chien, I.-L. and Fruehauf, P. S. (1990). Consider IMC Tuning to Improve Controller Performance. *Chemical Engineering Progress*, **86**(10), 33.

Clarke, D.; Mohtadi, C. and Tuffs, P. S. (1987a). Generalized Predictive Control - Part I. The Basic Algorithm. *Automatica*, **23**(2), 137 – 148.

Clarke, D.; Mohtadi, C. and Tuffs, P. S. (1987b). Generalized Predictive Control - Part II. Extensions and Interpretations. *Automatica*, **23**(2), 149 – 160.

Cohen, G. H. and Coon, G. A. (1953). Theoretical Consideration of Retarded Control. *Trans. ASME*, **75**, 827–834.

Cutler, C. and Ramaker, B. (1980). Dynamic Matrix Control - a Computer Control Algorithm. In *Proceedings of the Joint Automatic Control Conference*.

De Bruyne, F. (2003). Iterative feedback tuning for internal model controllers. *Control engineering practice*, **11**, 1043–1048.

De Bruyne, F.; Anderson, B. and Linard, N. (1998). The Hansen Scheme Revisited. In *Proceedings of the 37th IEEE Conference on Decision and Control*, volume 1, pages 706 – 711.

De Bruyne, F.; Anderson, B. and Linard, N. (1999). Implementation Issues for a Nonlinear Version of the Hansen Scheme. In *Proceedings of Information Decision and Control*, pages 295 – 300.

de Callafon, R. A. (1998). *Feedback Oriented identification for Enhanced and Robust Control - a fractional approach applied to a wafer stage*. Ph.D. thesis, Technical University of Delft, The Netherlands.

Desoer, C. A.; R.-W-Liu; Murray, J. and Saeks, R. (1980). Feedback System Design: The Fractional Representation Approach to Analysis and Synthesis. *IEEE Transactions on Automatic Control*, **25**(3), 399 – 412.

Dueñas, M. (2004). *Population balance modeling and passivity-based control of particulate processes, applied to the Silgrain®process*. Ph.D. thesis, Norwegian University of Science and Technology. Telemark University College.

Farschman, C. A.; Viswanath, K. and Ydstie, B. (1998). Process Systems and Inventory Control. *AIChE Journal*, **44**(8), 1841–1857.

Fel'dbaum, A. A. (1965). *Optimal Control Systems*. Mathematics in Science and Engineering. Academic Press.

Forssell, U. and Ljung, L. (2000). Some results on optimal experiment design. *Automatica*, **36**(5), 749 – 756.

Garcia, C. E. and Morari, M. (1982). Internal Model Control. 1. A Unifying Review and Some New Results. *Ind. Eng. Chem. Process Des. Dev.*, **21**(2), 308–323.

Gevers, M. (1996). Identification for control. *Annual Reviews in Control*, **20**, 95–106.

Gevers, M. (2002). A decade of progress in iterative process control design: from theory to practice. *Journal of process control*, **12**(4), 519–531.

Gevers, M. and Ljung, L. (1986). Optimal Experiment Design with Respect to the Intended Model Application. *Automatica*, **22**(5), 543–554.

Gevers, M.; Bombois, X.; Codrons, B.; Scorletti, G. and Anderson, B. D. O. (2003a). Model validation for control and controller validation in a prediction error identification framework - Part I: Theory. *Automatica*, **39**, 403–415.

Gevers, M.; Bombois, X.; Codrons, B.; Scorletti, G. and Anderson, B. D. O. (2003b). Model validation for control and controller validation in a prediction error identification framework - Part II: Illustrations. *Automatica*, **39**, 417–427.

Goodwin, G. C. and Payne, R. L. (1977). *Dynamic System Identification: Experiment Design and Data Analysis*, volume 136 of *Mathematics in science and engineering*. Academic Press.

Gopaluni, R. B.; Patwardhan, R. S. and Shah, S. L. (2004). MPC relevant identification - tuning the noise model. *Journal of process Control*, **14**(6), 699 – 714.

Grewal, M. S. and Andrews, A. P. (1993). *Kalman Filtering: Theory and Practice*. Prentice Hall Information and system sciences series. Prentice Hall.

Gunnarsson, S.; Collignon, V. and Rousseaux, O. (2003). Tuning of a decoupling controller for a 2x2 system using iterative feedback tuning. *Control Engineering Practice*, **11**, 1035–1041.

Hallager, L.; Goldschmidt, L. and Jørgensen, S. B. (1984). Multivariable adaptive identification and control of a distributed chemical reactor. *Large Scale Systems*, **6**, 323–336.

Hansen, F.; Franklin, G. and Kosut, R. (1989). Closed-Loop Identification via Fractional Representation: Experiment Design. In *Proceedings American Control Conference 1989, Pittsburgh, USA*, pages 1422–1427.

Hildebrand, R.; Lecchini, A.; Solari, G. and Gevers, M. (2003). A Convergence Theorem for Iterative Feedback Tuning. Technical report, CESAME, Universit Catholique de Louvain, Louvain-la-Neuve, Belgium.

Hildebrand, R.; Lecchini, A.; Solari, G. and Gevers, M. (2004). Prefiltering in Iterative Feedback Tuning: Optimization of the Prefilter for Accuracy. *IEEE Transactions on Automatic Control*, **49**(10), 1801 – 1805.

Hildebrand, R.; Lecchini, A.; Solari, G. and Gevers, M. (2005a). Asymptotic Accuracy of Iterative Feedback Tuning. *IEEE Transactions on Automatic Control*, **50**(8), 1182 – 1185.

Hildebrand, R.; Lecchini, A.; Solari, G. and Gevers, M. (2005b). Optimal Prefiltering in Iterative Feedback Tuning. *IEEE Transactions on Automatic Control*, **50**(8), 1196 – 1200.

Hjalmarsson, H. (1998). Control on Nonlinear Systems using Iterative Feedback Tuning. In *Proceedings of the American Control Conference*, pages 2083 – 2087.

Hjalmarsson, H. (1999). Efficient Tuning of Linear Multivariable Controllers using Iterative Feedback Tuning. *International journal of adaptive control and signal processing*, **13**, 553–572.

Hjalmarsson, H. (2002). Iterative feedback tuning - an overview. *International journal of adaptive control and signal processing*, **16**, 373–395.

Hjalmarsson, H. (2005). From experiment design to closed loop control. *Automatica*, **41**(3), 393 – 438.

Hjalmarsson, H. and Birkeland, T. (1998). Iterative Feedback Tuning of Linear Time-Invariant MIMO Systems. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 3893–3898.

Hjalmarsson, H.; Gevers, M.; Bruyne, F. D. and Leblond, J. (1994a). Identification for control: Closing the loop gives more accurate controllers. In *IEEE Proceedings of the 33rd Conference on Decision and Control*, pages 4150–4455.

Hjalmarsson, H.; Gunnarsson, S. and Gevers, M. (1994b). A Convergent Iterative Restricted Complexity Control Design Scheme. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, volume 2, pages 1735 – 1740.

Hjalmarsson, H.; Gevers, M. and de Bruyne, F. (1996). For Model-based Control Design, Closed-loop Identification Gives Better Performance. *Automatica*, **32**(12), 1659 – 1673.

Hjalmarsson, H.; Gevers, M. and Lequin, O. (1997). Iterative Feedback Tuning: Theory and applications in chemical process control. *Journal A*, **38**(1), 16 – 25.

Hjalmarsson, H.; Gevers, M.; Gunnarsson, S. and Lequin, O. (1998). Iterative Feedback Tuning: Theory and Applications. *IEEE Control Systems Magazine*, **18**(4), 26–41.

Huusom, J. K.; Poulsen, N. K. and Jørgensen, S. B. (2007a). Iterative Controller Tuning for Processes with Fold Bifurcations. In *Proceedings for ESCAPE 17*, pages 835 – 840.

Huusom, J. K.; Santacoloma, P. A.; Poulsen, N. K. and Jørgensen, S. B. (2007b). Data Driven Tuning of Inventory Controllers. In *Proceedings for 46'th IEEE Conference on Decision and Control*, pages 4191 – 4196.

Huusom, J. K.; Poulsen, N. K. and Jørgensen, S. B. (2008). Improving Convergence of Iterative Feedback Tuning. *In Press for Journal of Process Control.*. DOI information: 10.1016/j.jprocont.2008.09.004.

Jansson, H. and Hjalmarsson, H. (2004). Gradient approximations in iterative feedback tuning for multivariable processes. *International journal of adaptive control and signal processing*, **18**, 665–681.

Jansson, H. and Hjalmarsson, H. (2005). Input Design via LMIs Admitting Frequency-wise Model Specifications in Confidence Regions. *IEEE Transactions on Automatic Control*, **50**(10), 1534–1549.

Jørgensen, S. B. and Lee, J. H. (2002). Recent Advances and Challenges in Process Control. In *Chemical Process Control 6*, pages 55 – 74.

Johasson, K. H. (2000). The Quadruple-Tank Process: A Multivariable Laboratory Process with an Adjustable Zero. *IEEE Transactions on Control Systems Technology*, **8**(3), 456–465.

Kailath, T.; Sayed, A. H. and Hassibi, B. (2000). *Linear Estimation*. Prentice Hall information and system sciences series. Prentice Hall.

Karimi, A.; Mišković, L. and Bonvin, D. (2003). Iterative correlation-based controller tuning with application to a magnetic suspension system. *Control Engineering Practice*, **11**(9), 1069 – 1078.

Karimi, A.; Mišković, L. and Bonvin, D. (2004). Iterative correlation-based controller tuning. *Int. journal of adaptive control and signal processing*, **18**(8), 645 – 664.

Kučera, V. (1991). *Analysis and Design of Discrete Linear Control Systems*. Prentice Hall International Series in Systems and Control Engineering. Prentice Hall.

Kwakernaak, H. and Sivan, R. (1972). *Linear Optimal Control Systems*. John Wiley & Sons Inc.

Laachi, N. and Rawlings, J. B. (2005). A Candidate to replace PID Control: SISO-Constrained LQ Control. *AIChE Journal*, **51**(4), 1178 – 1189.

Lecchini, A. (2001). *Virtual Reference Feedback Tuning (VRFT): A New Direct Data-Based Method for the Design of Feedback Controllers*. Ph.D. thesis, Universit di Brescia, Italy.

Lecchini, A. and Gevers, M. (2002). On Iterative Feedback Tuning for non-minimum phase plants. In *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 4658 – 4663.

Lecchini, A.; Campi, M. C. and Savaresi, S. (2002). Virtual reference feedback tuning for two degree of freedom controllers. *International Journal of Adaptive Control and Signal Processing*, **16**, 355–371.

Lequin, O. (1997). Optimal Closed-Loop PID Tuning in the Process Industry with the "Iterative Feedback Tuning" Scheme. In *4th European Control Conference*. paper TH-A-H6.

Lequin, O.; Gevers, M. and Triest, L. (1999). Optimizing the Settling Time with Iterative Feedback tuning. In *Proceedings of 14th IFAC world congress*, pages 433–437. paper I-3b-08-3.

Lequin, O.; Gevers, M.; Mossberg, M.; Bosmans, E. and Triest, L. (2003). Iterative feedback tuning of PID parameters: Comparison with classical tuning rules. *Control Engineering Practice*, **11**, 1023–1033.

Levenberg, K. (1944). A method for the solution of certain problems in least squares. *Quarterly of Applied Mathematics*, **2**, 164–168.

Li, D.; Qian, F. and Fu, P. (2008). Optimal nominal dual control for discrete-time linear-quadratic Gaussian problem with unknown parameters. *Automatica*, **44**(1), 119 – 127.

Lindoff, B.; Holst, J. and Wittenmark, B. (1999). Analysis of Approximations of Dual Control. *International Journal of Adaptive Control Signal Processing*, **13**(7), 593 – 620.

Ljung, L. (1999). *System identification - Theory for the user*. Prentice Hall information and system sciences series. Prentice hall, second edition.

Luyben, W. L.; Tyréus, B. D. and Luyben, M. L. (1998). *Plantwide Process Control*. McGraw Hill.

Maciejowski, J. M. (2002). *Predictive Control with Constraints*. Prentice Hall.

Madsen, K.; Nielsen, H. B. and Tingleff, O. (2004). *Method For Non-Linear Least Squares Problems*. IMM, Technical University of Denmark, second edition.

Marquardt, D. W. (1963). An Algorithm for Least-Squares Estimation of Nonlinear Parameters. *Journal of the Society for Industrial and Applied Mathematics*, **11**(2), 431–441.

Milanese, M. and Taragna, M. (2005). H_∞ set membership identification: A survey. *automatica*, **41**(12), 2019–2032.

Mišković, L. (2006). *Data-Driven Controller Tuning using the Correlation Approach*. Ph.D. thesis, Éole Polytechnique Fédérale de Lausanne.

Mišković, L.; Karimi, A. and Bonvin, D. (2003). Correlation-Based Tuning of a Restricted-Complexity Controller for an Active Suspension System. *European Journal of Control*, **9**(1), 77 – 83.

Muske, K. R. and Rawlings, J. B. (1993). Model Predictive Control with Linear Models. *AIChE Journal*, **39**(2), 262 – 287.

Nielsen, H. B. (1999). Damping Parameter in Marquardt's Method. Technical Report IMM-REP-1999-05, IMM, DTU. Available as http://www.imm.dtu.dk/ hbn/publ/TR9905.ps.

Nocedal, J. and Wright, S. J. (1999). *Numerical optimization*. Springer series in operations research. Springer.

Packard, A. and Doyle, J. (1993). The Complex Structured Singular Value. *Automatica*, **29**(1), 71 – 109.

Poulsen, N. K. (1997). *Stokastisk Adaptiv Regulering - Stokastiske systemer, Filtrering og Regulering*, volume 2. IMM - DTU. Danish.

Procházka, H.; Gevers, M.; Anderson, B. D. O. and Ferrera, C. (2005). Iterative Feedback Tuning for robust controller design and optimization. In *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, pages 3602–3607.

Qin, S. J.; Lin, W. and Ljung, L. (2005). A novel subspace identification approach with enforced causal models. *Automatica*, **41**(12), 2043 – 2053.

Richalet, J.; Rault, A.; Testud, J. L. and Papon, J. (1978). Model predictive Heuristic Control: Application to Industrial Processes. *Automatica*, **14**(5), 413 – 428.

Rivera, D. E.; Morari, M. and Skogestad, S. (1986). Internal Model Control. 4. PID Controller Design. *Ind. Eng. Chem. Process Des. Dev.*, **25**(1), 252–265.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *Annals of mathematical statistics*, **22**(3), 400–407.

Schrama, R. (1992a). *Approximate Identification and control design*. Ph.D. thesis, Delft University of Technology.

Schrama, R. J. P. (1992b). Accurate Identification for Control: The Necessity of an Iterative Scheme. *IEEE Transactions on automatic control*, **37**(7), 991–994.

Sira, H. and Angulo, M. (1997). Passivity-based control of nonlinear chemical processes. *INT. J. CONTROL*, **68**(5), 971–996.

Sjöberg, J.; Zhang, Q.; Ljung, L.; Benveniste, A.; Delyon, B.; Glorennec, P. Y.; Hjalmarsson, H. and Judisky, A. (1995). Nonlinear Black-box Modeling in Systems Identification: a Unified Overview. *Automatica*, **31**(12), 1691 – 1724.

Sjöberg, J.; De Bruyne, F.; Agarwal, M.; Anderson, B. D. O.; Gevers, M.; Kraus, F. J. and Linard, N. (2003). Iterative controller optimization for nonlinear systems. *Control Engineering Practice*, **11**, 1079–1086.

Skogestad, S. (2003). Simple analytic rules for model reduction and PID controller tuning. *Journal of Process Control*, **13**(4), 291–309.

Skogestad, S. and Postlethwaite, I. (1996). *Multivariable Feedback Control - Analysis and Design*. John Wiley & sons.

Söderström, T. and Stoica, P. (1989). *System identification*. Prentice Hall.

Solari, G. and Gevers, M. (2004). Unbiased Estimation of the Hessian for Iterative Feedback Tuning (IFT). In *Proceedings of 43rd IEEE Conference on Decision and Control*, volume 2, pages 1759–1760.

Van den Hof, P. M. J. and Schrama, R. J. P. (1993). An indirect method for transfer function estimation from closed loop data. *Automatica*, **29**(6), 1523–1527.

Van den Hof, P. M. J. and Schrama, R. J. P. (1995). Identification and Control - Closed Loop Issues. *Automatica*, **31**(12), 1751 – 1770.

Van den Hof, P. M. J.; Schrama, R. J. P.; de Callafon, R. A. and Bosgra, O. H. (1995). Identification of normalized coprime plant factors from closed loop experimental data. *European Journal of Control*, **1**(1), 62–74.

Van Overschee, P. and De Moor, B. (1994). N4SID: Subspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems. *Automatica*, **30**(1), 75 – 93.

Veres, S. and Hjalmarsson, H. (2002). Tuning for robustness and performance using iterative feedback tuning. In *Proceedings of the 14st IEEE Conference on Decision and Control*, pages 4682–4687.

Verhaegen, M. (1994). Identification of the Deterministic Part of MIMO State Space Models given in Innovations Form from Input-Output Data. *Automatica*, **30**(1), 61 – 74.

Vinnicombe, G. (1993). Frequency Domain Uncertainty and the Graph Topology. *IEEE Transactions on Automatic control*, **38**(9), 1371–1383.

Willems, J. C. (2007). In Control, Almost from the Beginning Until the Day After Tomorrow. *European Journal of Control*, **13**(1), 71–81.

Wittenmark, B.; Åstrøm, K. J. and Jørgensen, S. B. (1995). *Process Control*.

Ydstie, B. and Alonso, A. (1997). Process system and passivity via the Clausius-Planck inequality. *Systems & Control Letters*, **30**, 253–264.

Ydstie, B. E. (1987). Adaptive Process Control. *Chemical Engineering Research and Design*, **65**(6), 480 – 489.

Youla, D. C.; Bongiorno, J. J. and Jabr, H. A. (1976). Modern Wiener-Hopf Design of Optimal Controllers Part 1: The single-Input-Output Case. *IEEE Transactions on Automatic Control*, **21**(1), 3 – 13.

Zang, Z.; Bitmead, R. R. and Gevers, M. (1995). Iterative Weighted Least-squares Identification and Weighted LQG Control Design. *Automatica*, **31**(11), 1577–1594.

Zhou, K. and Doyle, J. C. (1998). *Essentials of Robust Control*. Prentice Hall.

Ziegler, J. G. and Nichols, N. B. (1942). Optimum Settings for Automatic Controllers. *Trans. ASME*, **64**(8), 759.