

Technical University of Denmark



Improving Convergence of Iterative Feedback Tuning

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

Published in:
Journal of Process Control

Link to article, DOI:
[10.1016/j.jprocont.2008.09.004](https://doi.org/10.1016/j.jprocont.2008.09.004)

Publication date:
2009

Document Version
Early version, also known as pre-print

[Link back to DTU Orbit](#)

Citation (APA):
Huusom, J. K., Poulsen, N. K., & Jørgensen, S. B. (2009). Improving Convergence of Iterative Feedback Tuning. *Journal of Process Control*, 19(4), 570-578. DOI: 10.1016/j.jprocont.2008.09.004

DTU Library

Technical Information Center of Denmark

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Improving Convergence of Iterative Feedback Tuning

Jakob Kjøbsted Huusom ^a, Niels Kjølstad Poulsen ^b,
Sten Bay Jørgensen ^a

^a*Department of Chemical Engineering, Technical University of Denmark, DK-2800 Lyngby*

^b*Informatics and Mathematical Modelling, Technical University of Denmark, DK-2800 Lyngby*

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.

Key words: Iterative Feedback Tuning, Controller tuning, Experimental design

Email addresses: jkh@kt.dtu.dk (Jakob Kjøbsted Huusom), nkp@imm.dtu.dk (Niels Kjølstad Poulsen), sbj@kt.dtu.dk (Sten Bay Jørgensen).

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 [17,1,20].

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 [6] 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 [4], using the more conservative \mathcal{H}_∞ 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 3 shows how Iterative Feedback Tuning fits into this category and how an unbiased gradient estimate is achieved from data. Section 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 5 presents a new algorithm for Per-

turbed 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.

2 Criterion Based Controller Tuning

A description of a scalar closed loop system is depicted in Figure 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:

$$y_t = \frac{GC_r}{1 + GC_y}r_t + \frac{1}{1 + GC_y}v_t = Tr_t + Sv_t \quad (1a)$$

$$u_t = \frac{C_r}{1 + GC_y}r_t - \frac{C_y}{1 + GC_y}v_t = SC_r r_t - SC_y v_t \quad (1b)$$

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

$$\tilde{y}_t = y_t(\boldsymbol{\rho}) - y_t^d \quad (2)$$

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 (3). In case the cost function is convex the minimization will converge to the global minimizer, but this is in general not true.

$$\begin{aligned} \boldsymbol{\rho}_{i+1} &= \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1} \frac{\partial F(\boldsymbol{\rho}_i)}{\partial \boldsymbol{\rho}} \\ &= \boldsymbol{\rho}_i - \gamma_i \mathbf{R}_i^{-1} \mathbf{J}(\boldsymbol{\rho}_i) \end{aligned} \quad (3)$$

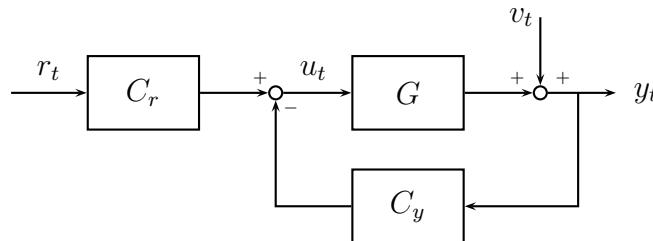


Fig. 1. Feedback loop with a two degree of freedom controller.

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 γ will thus affect the convergence properties of the method [17,32]. The iteration step size γ_i can be determined using e.g. a line search method.

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 (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 [17] and has been analyzed, extended and tested in a number of papers. References [5] and [13] 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 [15]. Using an estimate of the gradient vector in (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 (4) [34].

$$\sum_{i=1}^{\infty} \gamma_i^2 < \infty, \quad \sum_{i=1}^{\infty} \gamma_i = \infty \quad (4)$$

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 [30]. 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

$$F(\boldsymbol{\rho}) = \frac{1}{2N} \mathbb{E} \left[\sum_{t=1}^N \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=1}^N u_t(\boldsymbol{\rho})^2 \right] \quad (5)$$

where the minimization criterion is

$$\mathbf{0} = \mathbf{J}(\boldsymbol{\rho}) = \frac{1}{N} \mathbb{E} \left[\sum_{t=1}^N \tilde{y}_t(\boldsymbol{\rho}) \frac{\partial \tilde{y}_t}{\partial \boldsymbol{\rho}} + \lambda \sum_{t=1}^N u_t(\boldsymbol{\rho}) \frac{\partial u_t}{\partial \boldsymbol{\rho}} \right] \quad (6)$$

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 (1).

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})} \frac{\partial C_r}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho}) \mathbf{r} - \frac{1}{C_r(\boldsymbol{\rho})} \frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho}) \mathbf{y} \quad (7a)$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \frac{\partial C_r}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}) \mathbf{r} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}) \mathbf{y} \quad (7b)$$

or rewritten into the more favorable form.

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})} \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) T(\boldsymbol{\rho}) \mathbf{r} + \frac{1}{C_r(\boldsymbol{\rho})} \frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho}) (\mathbf{r} - \mathbf{y}) \quad (8a)$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) S(\boldsymbol{\rho}) \mathbf{r} + \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho}) (\mathbf{r} - \mathbf{y}) \quad (8b)$$

where the estimates produced from this expression are expected to have a reduced variance compared to (7) in case $\frac{\partial C_r}{\partial \boldsymbol{\rho}} \approx \frac{\partial C_y}{\partial \boldsymbol{\rho}}$. The derivation of (7) and (8) are given in [15].

3.1 The Tuning algorithm

Estimates of the derivatives (8a) and (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

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

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

$$\tilde{y}_t = y_t^1 - y_t^d \quad (9a)$$

$$u_t = u_t^1 \quad (9b)$$

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) \mathbf{y}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}} \mathbf{y}^2 \right] \quad (9c)$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{1}{C_r(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) \mathbf{u}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}} \mathbf{u}^2 \right] \quad (9d)$$

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

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) \mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}} \mathbf{v}^2 \right] \quad (10a)$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) \mathbf{v}^3 + \frac{\partial C_y}{\partial \boldsymbol{\rho}} \mathbf{v}^2 \right] \quad (10b)$$

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 (8). The Iterative Feedback Tuning method is depicted in Figure 2. The performance check is evaluated by repeating the first experiment with the updated controller.

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 [11,35] states that

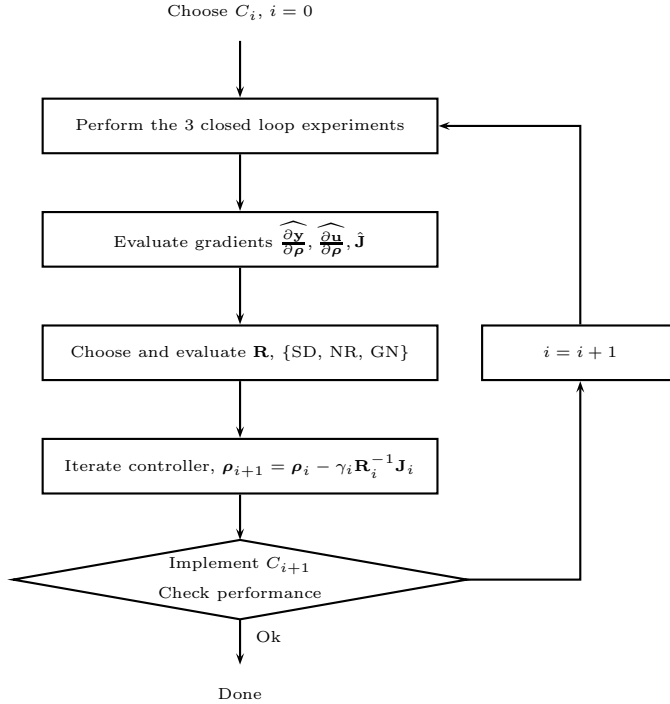


Fig. 2. Schematic representation of the Iterative Feedback Tuning method.

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 [35] and successful tuning of PID loops for industrial processes has been reported in [23,16]. The theory has furthermore been extended to cover optimization of multivariable processes, which implies that more experiments in each iteration are necessary [14,12,19] and to cover non-minimum phase and time delay systems [22]. Finally the only restriction on the control structure is that the closed loop is stable and that the transfer functions in (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 [2] has applied the method with internal model controllers and the reference [18] applies the method on a nonlinear inventory control structure. The filtering in (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 [15].

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 (3) steps in a descent direction, the new controller may render the loop unstable. [3] provides an algorithm which ensures

stability using the generalized stability margin, evaluated by using estimates of the closed loop transfer function. [38,33] go further and define two cost functions, one for performance and another for robustness. The performance cost is of the form (5) and the robustness criterion is minimizing some norm of the closed loop sensitivities, preferably the \mathcal{H}_∞ 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 3.1 and equations (9) and (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 [15] 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 (9c) and (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 [8,10] 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 [9].

Virtual Reference Feedback Tuning [1] and Correlation-based Tuning [20] 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 [21]. 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 [29]. In the present contribution an alternative route for obtaining informative experiments and fast convergence is pursued.

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 [36,26]. 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

$$\mathbf{y} = T(\mathbf{r} + \mathbf{r}_p) + S(\mathbf{v} + G\mathbf{u}_p) \quad (11a)$$

$$\mathbf{u} = SC_r(\mathbf{r} + \mathbf{r}_p) - SC_y\mathbf{v} + \mathbf{u}_p \quad (11b)$$

The derivatives of the input and output in (11) with respect to the control parameters is determined as in (7).

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})} \frac{\partial C_r}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p) - \frac{1}{C_r(\boldsymbol{\rho})} \frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})\mathbf{y} \quad (12a)$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \frac{\partial C_r}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p) - \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})\mathbf{y} \quad (12b)$$

or rewritten into the more favorable form.

$$\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} = \frac{1}{C_r(\boldsymbol{\rho})} \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) T(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p) + \frac{1}{C_r(\boldsymbol{\rho})} \frac{\partial C_y}{\partial \boldsymbol{\rho}} T(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p - \mathbf{y}) \quad (13a)$$

$$\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = \left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) S(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p) + \frac{\partial C_y}{\partial \boldsymbol{\rho}} S(\boldsymbol{\rho})(\mathbf{r} + \mathbf{r}_p - \mathbf{y}) \quad (13b)$$

By conducting the three experiments described in section 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 (9), that not every type of perturbation strategy should be recommended:

$$\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}} + \frac{S(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) (\mathbf{v}^3 + G\mathbf{u}_p^3) + \frac{\partial C_y}{\partial \boldsymbol{\rho}} (\mathbf{v}^2 + G\mathbf{u}_p^2) \right] \quad (14a)$$

$$\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}} = \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} - \frac{S(\boldsymbol{\rho})C_y(\boldsymbol{\rho})}{C_r(\boldsymbol{\rho})} \left[\left(\frac{\partial C_r}{\partial \boldsymbol{\rho}} - \frac{\partial C_y}{\partial \boldsymbol{\rho}} \right) (\mathbf{v}^3 + G\mathbf{u}_p^3) + \frac{\partial C_y}{\partial \boldsymbol{\rho}} (\mathbf{v}^2 + G\mathbf{u}_p^2) \right] \quad (14b)$$

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 $\widehat{\frac{\partial \mathbf{y}}{\partial \boldsymbol{\rho}}}$ and $\widehat{\frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}}$, but note that it must be required that $\mathbf{r}_p^2 = \mathbf{r}_p^3$ in order to construct equation (13) from (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 $E[\boldsymbol{\rho}_n - \boldsymbol{\rho}^{opt}]$ 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 $E[\boldsymbol{\rho}_n(\mathbf{r}^p) - \boldsymbol{\rho}^{opt}]$ 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

$$E[F(\boldsymbol{\rho}_n(\mathbf{r}_p))] < E[F(\boldsymbol{\rho}_n)] \quad (15)$$

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 [13].

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 [7]. 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

$$\begin{aligned} \boldsymbol{\vartheta}^{opt} &= \arg \max_{\boldsymbol{\vartheta}} \det(\mathbf{H}) \\ s.t. \quad & F(\boldsymbol{\rho}, \boldsymbol{\vartheta}) \leq F^{max} \end{aligned} \quad (16)$$

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 course 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 (16) does not restrict neither the system nor the parameterization of the perturbation. Convexity is therefore not guaranteed.

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 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.

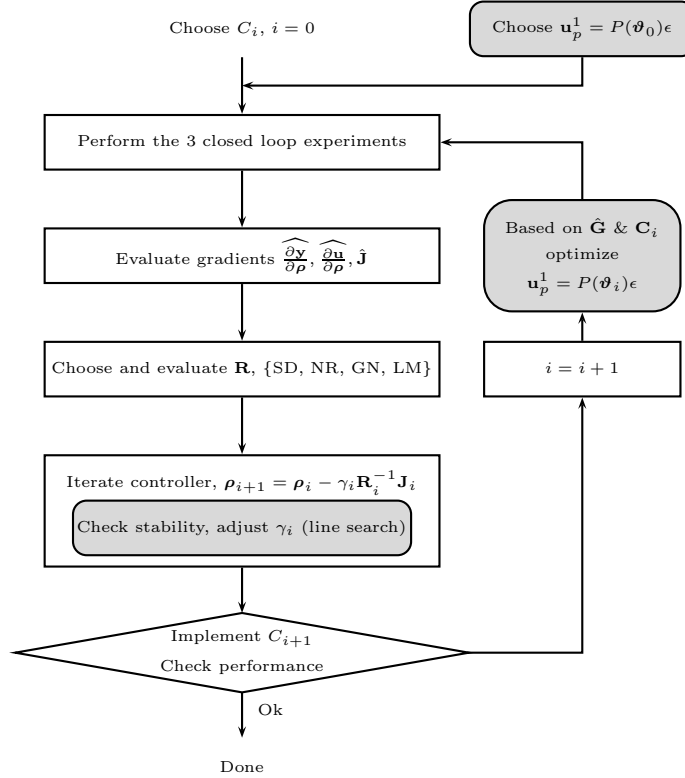


Fig. 3. Workflow in the novel Perturbed Iterative Feedback Tuning. The new elements are shown with a gray shaded background. The parameters, $\boldsymbol{\vartheta}_i$, are the filter coefficients.

The workflow in Figure 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 [36,26].

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 [17] equation (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 [37] 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 (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.

$$\hat{\mathbf{H}} = \frac{1}{N} \sum_{t=1}^N \left[\frac{\widehat{\partial y_t}}{\partial \boldsymbol{\rho}} \left(\frac{\widehat{\partial y_t}}{\partial \boldsymbol{\rho}} \right)^T + \lambda \frac{\widehat{\partial u_t}}{\partial \boldsymbol{\rho}} \left(\frac{\widehat{\partial u_t}}{\partial \boldsymbol{\rho}} \right)^T \right] \quad (17)$$

The Gauss-Newton optimization method is known to perform very well in the vicinity of the optimal solution [32]. 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 [25].

$$\mathbf{R}_i = \hat{\mathbf{H}}_i + \mu_i \mathbf{I} \quad (18)$$

A starting value for the damping coefficient is

$$\mu_0 = \tau \max(\text{diag}(\hat{\mathbf{H}}_0)) \quad (19)$$

where τ is 10^{-6} for a good initial guess and 10^{-3} to 1 if the guess is expected to be poor [27]. 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:

$$\varrho = \frac{F_{i-1} - F_i}{\frac{1}{2} \mathbf{h}_i^T (\mu_i \mathbf{h}_i - \mathbf{J}_i)} \quad (20)$$

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 [28] is

$$\mu_{i+1} = \begin{cases} 2\mu_i, & \varrho \in]-\infty; 0.25] \\ \mu_i, & \varrho \in]0.25; 0.75[\\ \mu_i/3, & \varrho \in [0.75; \infty[\end{cases} \quad (21)$$

In [31] the damping coefficient is updated by

$$\varrho < 0 \begin{cases} \mu_{i+1} = \mu_i \nu_i \\ \nu_{i+1} = 2\nu_i \end{cases} \quad (22)$$

$$\varrho \geq 0 \begin{cases} \mu_{i+1} = \mu_i \max(\frac{1}{3}, 1 - (2\varrho - 1)^3) \\ \nu_{i+1} = 2 \end{cases}$$

where $\nu_0 = 2$. This scheme is a continuous version of the strategy suggested by [28] but converges generally faster. The two updating strategies are illustrated in Figure 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

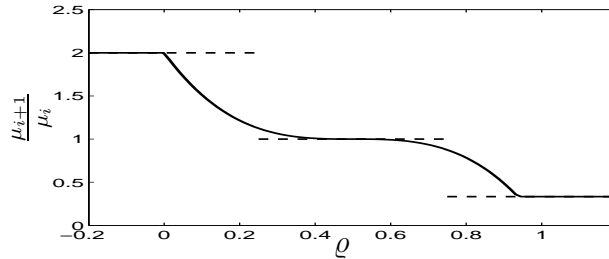


Fig. 4. Updating of the damping coefficient as function of the gain ratio. The straight dashed curves is the strategy proposed in [28] cf. equation (21) and the full line is the strategy proposed by [31] cf. equation (22).

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 [24] 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 [24] was

$$F(\boldsymbol{\rho}) = \frac{1}{2N} \mathbb{E} \left[\sum_{t=t_0}^N \tilde{y}_t(\boldsymbol{\rho})^2 + \lambda \sum_{t=1}^N u_t(\boldsymbol{\rho})^2 \right] \quad (23)$$

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 optimizes 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 Simulation Examples

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

$$C_r : \frac{U(s)}{R(s)} = K_c \left[1 + \frac{1}{\tau_I s} \right] \quad (24a)$$

$$C_y : \frac{U(s)}{Y(s)} = K_c \left[1 + \frac{1}{\tau_I s} + \frac{\tau_D s}{0.1\tau_D s + 1} \right] \quad (24b)$$

working in closed loop on the linear time invariant second order process model, (25a), affected by Gaussian white noise $v_t \in \mathcal{N}(0, 0.05^2)$ filtered through the first order noise model (25b). See Figure 1.

$$G(s) = \frac{1}{s^2 + 0.1s + 1} \quad (25a)$$

$$H(s) = \frac{1}{s + 1} \quad (25b)$$

This process model was used in [24] 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 6.2, 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 used 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 [24], which gives a very poorly tuned loop, but helps to illustrate some of the inherent problems

in Iterative Feedback Tuning.

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

6.1 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 (5). When the classical Iterative Feedback Tuning method is used with the simple Gauss-Newton optimization, the biased Hessian estimate (17) and $\gamma = \mathbf{1}$, 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, [24] uses the cost function (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 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.

Table 1

Control parameters and the value of the performance cost function for each iteration with the corresponding mask as suggested by [24]. 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

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

In this paper the above problem is solved using Levenberg-Marquardt opti-

mization, with $\tau = 10^{-4}$ in (19) and the update strategy for μ is based on (22). The results are presented in Table 2. Comparing the results in Tables 1 and

Table 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

Iteration	K_c	τ_I	τ_D	Mask	$F(\rho_i) \cdot 10^3$
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

2, it is seen that the performance after 5 iterations is very close despite the different development of the control parameters through the iterations. The main difference between two methods is, that the method of [24] 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 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.

6.2 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 (19) and the update strategy for μ is based on (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

$$u_p = P(\boldsymbol{\vartheta})\epsilon, \quad \epsilon_t \in \mathcal{N}(0, 1), \quad t \in \{1, 2, \dots, N\} \quad (26)$$

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

$$P(\boldsymbol{\vartheta}) = \frac{\vartheta_1}{\vartheta_2 s + 1}, \quad \boldsymbol{\vartheta} \in \mathbb{R}_+^2 \quad (27)$$

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 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 3 and 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 %.

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 improvements can be observed be-

Table 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, $\bar{F}_{MC}(\rho_i)$.

Iteration	K_c	τ_I	τ_D	$\bar{F}_{MC}(\rho_i) \cdot 10^3$
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

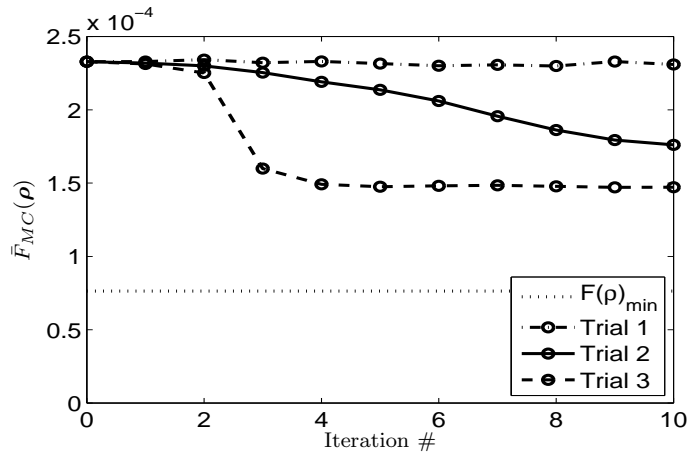


Fig. 5. Performance of each controller through the iterations for all three series evaluated based on 100 Monte Carlo trials. The cost for the local minimizer is also displayed as the dotted line.

tween 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.

Table 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, $\bar{F}_{MC}(\boldsymbol{\rho}_i)$.

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

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.

References

- [1] M. C. Campi, A. Lecchini, and S. M. Savaresi. Virtual reference feedback tuning: a direct method for the design of feedback controllers. *Automatica*, 28:1337 – 1346, 2002.
- [2] Franky De Bruyne. Iterative feedback tuning for internal model controllers. *Control engineering practice*, 11:1043–1048, 2003.
- [3] Franky De Bruyne, Brian Anderson, and Natasha Linard. Implementation issues for a nonlinear version of the hansen scheme. In *Proceedings of Information Decision and Control*, pages 295 – 300, 1999.
- [4] Raymond Arnoud de Callafon. *Feedback Oriented identification for Enhanced and Robust Control - a fractional approach applied to a wafer stage*. PhD thesis, Technical University of Delft, The Netherlands, October 1998.
- [5] Michel Gevers. A decade of progress in iterative process control design: from theory to practice. *Journal of process control*, 12(4):519–531, 2002.
- [6] Michel Gevers, Xavier Bombois, Benoit Codrons, Gerard Scorletti, and Brian D. O. Anderson. Model validation for control and controller validation in a prediction error identification framework - part i: Theory. *Automatica*, 39:403–415, 2003.
- [7] Graham C. Goodwin and Robert L. Payne. *Dynamic System Identification: Experiment Design and Data Analysis*, volume 136 of *Mathematics in science and engineering*. Academic Press, 1977.
- [8] R. Hildebrand, A. Lecchini, G. Solari, and M. Gevers. Prefiltering in iterative feedback tuning: Optimization of the prefilter for accuracy. *IEEE Transactions on Automatic Control*, 49(10):1801 – 1805, 2004.
- [9] R. Hildebrand, A. Lecchini, G. Solari, and M. Gevers. Asymptotic accuracy of iterative feedback tuning. *IEEE Transactions on Automatic Control*, 50(8):1182 – 1185, 2005.
- [10] R. Hildebrand, A. Lecchini, G. Solari, and M. Gevers. Optimal prefiltering in iterative feedback tuning. *IEEE Transactions on Automatic Control*, 50(8):1196 – 1200, 2005.
- [11] Håkan Hjalmarsson. Control on nonlinear systems using iterative feedback tuning. In *Proceedings of the American Control Conference*, pages 2083 – 2087, 1998.
- [12] Håkan Hjalmarsson. Efficient tuning of linear multivariable controllers using iterative feedback tuning. *International journal of adaptive control and signal processing*, 13:553–572, 1999.
- [13] Håkan Hjalmarsson. Iterative feedback tuning - an overview. *International journal of adaptive control and signal processing*, 16:373–395, 2002.

- [14] Håkan Hjalmarsson and Thom Birkeland. Iterative feedback tuning of linear time-invariant mimo systems. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 3893–3898, 1998.
- [15] Håkan Hjalmarsson, Michel Gevers, Svante Gunnarsson, and Olivier Lequin. Iterative feedback tuning: Theory and applications. *IEEE Control Systems Magazine*, 18(4):26–41, 1998.
- [16] Håkan Hjalmarsson, Michel Gevers, and Olivier Lequin. Iterative feedback tuning: Theory and applications in chemical process control. *Journal A*, 38(1):16 – 25, 1997.
- [17] Håkan Hjalmarsson, Svante Gunnarsson, and Michel Gevers. A convergent iterative restricted complexity control design scheme. In *Proceedings of the 33rd IEEE Conference on Decision and Control*, volume 2, pages 1735 – 1740, 1994.
- [18] Jakob Kjøbsted Huusom, Paloma Andrade Santacoloma, Niels Kjølstad Poulsen, and Sten Bay Jørgensen. Data driven tuning of inventory controllers. In *Proceedings for 46th IEEE Conference on Decision and Control*, pages 4191 – 4196, 2007.
- [19] Henrik Jansson and Håkan Hjalmarsson. Gradient approximations in iterative feedback tuning for multivariable processes. *International journal of adaptive control and signal processing*, 18:665–681, 2004.
- [20] A. Karimi, L. Mišković, and D. Bonvin. Iterative correlation-based controller tuning. *Int. journal of adaptive control and signal processing*, 18(8):645 – 664, 2004.
- [21] Andrea Lecchini. *Virtual Reference Feedback Tuning (VRFT): A New Direct Data-Based Method for the Design of Feedback Controllers*. PhD thesis, Universit di Brescia, Italy, 2001.
- [22] Andrea Lecchini and Michel Gevers. On iterative feedback tuning for non-minimum phase plants. In *Proceedings of the 41st IEEE Conference on Decision and Control*, pages 4658 – 4663, 2002.
- [23] Olivier Lequin. Optimal closed-loop pid tuning in the process industry with the ”iterative feedback tuning” scheme. In *4th European Control Conference*, 1997. paper TH-A-H6.
- [24] Olivier Lequin, Michel Gevers, and Lionel Triest. Optimizing the settling time with iterative feedback tuning. In *Proceedings of 14th IFAC world congress*, pages 433–437, 1999. paper I-3b-08-3.
- [25] K. Levenberg. A method for the solution of certain problems in least squares. *Quarterly of Applied Mathematics*, 2:164–168, 1944.
- [26] Lennart Ljung. *System identification - Theory for the user*. Prentice Hall information and system sciences series. Prentice hall, 2 edition, 1999.

- [27] K. Madsen, H. B. Nielsen, and O. Tingleff. *Method For Non-Linear Least Squares Problems*. IMM, Technical University of Denmark, 2 ed edition, 2004.
- [28] Donald W. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.
- [29] L. Mišković, A. Karimi, and D. Bonvin. Correlation-based tuning of a restricted-complexity controller for an active suspension system. *European Journal of Control*, 9(1):77 – 83, 2003.
- [30] Ljubisa Mišković. *Data-Driven Controller Tuning using the Correlation Approach*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2006.
- [31] Hans Brun Nielsen. Damping parameter in marquardt’s method. Technical Report IMM-REP-1999-05, IMM, DTU, 1999. Available as <http://www.imm.dtu.dk/hbn/publ/TR9905.ps>.
- [32] Jorge Nocedal and Stephen J. Wright. *Numerical optimization*. Springer series in operations research. Springer, 1999.
- [33] Hynek Procházka, Michel Gevers, Brian D. O. Anderson, and Christel Ferrera. 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, 2005.
- [34] Herbert Robbins and Sutton Monro. A stochastic approximation method. *Annals of mathematical statistics*, 22(3):400–407, 1951.
- [35] J. Sjöberg, F De Bruyne, M. Agarwal, B. D. O. Anderson, M. Gevers, F. J. Kraus, and N. Linard. Iterative controller optimization for nonlinear systems. *Control Engineering Practice*, 11:1079–1086, 2003.
- [36] Torsten Söderström and Petre Stoica. *System identification*. Prentice Hall, 1989.
- [37] Gabriel Solari and Michel Gevers. 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, 2004.
- [38] Sandor Veres and Håkan Hjalmarsson. Tuning for robustness and performance using iterative feedback tuning. In *Proceedings of the 14th IEEE Conference on Decision and Control*, pages 4682–4687, 2002.