Iterative controller optimization for nonlinear systems [1]

J. Sjöberg [a], F. De Bruyne [b], M. Agarwal [c], B. D. O. Anderson [b],
M. Gevers [d], F. J. Kraus [e], and N. Linard [b]

[a] *Department of Machine and Vehicle Systems, Chalmers University of Technology,
412 96 Gothenburg, Sweden*

[b] *Department of Systems Engineering and Cooperative Research Center for Robust
and Adaptive Systems, RSISE, The Australian National University, Canberra
ACT 0200, Australia*

[c] *Corporate Technology, Buhler AG, CH-9240 Uzwil, Switzerland*

[d] *CESAME, Centre for Systems Engineering and Applied Mechanics, Université
Catholique de Louvain, Bâtiment Euler, Avenue Georges Lemaître 4-6, 1348
Louvain-La-Neuve, Belgium*

[e] *Automatic Control Laboratory, Swiss Federal Institute of Technology, CH-8092
Zurich, Switzerland*

Abstract

Recently, a data-driven model-free control design method has been proposed in
(7; 6) for linear systems. It is based on the minimization of a control criterion
with respect to the controller parameters using an iterative gradient technique.
In this paper, we extend this method to the case where both the plant and the
controller can be nonlinear. It is shown that an estimate of the gradient of the
control criterion can be constructed using only signal-based information obtained
from closed loop experiments. The obtained estimate contains a bias which depends
on the local nonlinearity of the noise description of the closed loop system which
can be expected to be small in many practical situations. As a side-effect the linear
model-free control design method is re-obtained in a new way.

Key words: Controller optimization; nonlinear control; optimal control; estimation

1 Introduction

The data-driven model-free control tuning method for linear systems, called Iterative Feedback Tuning (IFT), described in (7; 6) has been shown to give good results from both an experimental and industrial point of view: see e.g. (1; 4; 6). This scheme is based on an iterative tuning of the controller parameter vector along the gradient direction of a control performance criterion. The key contribution of (7; 6) was to show that an unbiased estimate of this gradient can be constructed from filtered versions of the signals measured on the closed-loop system. The construction of this gradient requires a "special experiment" in which a finite record of the output of the closed-loop system is recycled at the reference input of that system. The main advantage of this iterative procedure is that it is model-free. Indeed, tuning algorithms that use a model of the plant invariably give biased controllers, where the extent of the bias depends on the flexibility of the model. Another benefit of this tuning mechanism is that it only uses closed-loop "operating" data, i.e. there is no need to open the control loop.

In this paper, we extend these results to the case where both the controller and the plant can be nonlinear. Our main contribution is to show that it is possible to obtain an estimate of the gradient by performing a series of experiments on the closed loop system with reference signals close to one another. The gradient estimate is formed as the difference between the output signals of the different experiments. In the nonlinear case it is not possible to obtain a totally unbiased estimate, a small bias term due to nonlinear noise contribution cannot be removed. However, if the noise level is low and if the closed loop performance is fairly good then this bias term will be small.

The total number of experiments per iteration is equal to the number of controller parameters, n, plus two, i.e. one experiment for the generation of the error signal and $n+1$ experiments for the estimation of the n gradient signals. This should be compared with the linear case where three experiments are enough. The difference is due to the fact that although linear transfer functions commute, i.e., the order of the operators can be changed, this does not hold in the nonlinear case.

Apart from the fact that the plant and controller can be nonlinear, the suggested algorithm has the same properties as its linear sibling. For example, stability is not guaranteed and has to be dealt with separately with some other technique. This contribution concentrates on the nonlinear case and the common features with the linear case are covered in (7; 6).

Our method can be seen as an alternative to the obvious idea of computing gradient estimates of the control criterion by performing experiments with perturbed controller parameters. Indeed, in many practical situations it might be safer to change the reference signal slightly than to experiment with the control parameters.

The suggested tuning scheme becomes inconvenient when the number of controller parameters is high since the number of necessary experiments in each iteration of the minimization grows linearly with the number of parameters. In such cases the insights described here can be used to motivate approximate schemes which demands less experiments by including time-varying models, see (8; 3).

The organization of the paper is as follows. Section 2 formulates the tuning problem and Section 3 presents the idea of the nonlinear tuning algorithm. This is then illustrated by a small example in Section 4. The analysis is given in Section 5, where also the algorithm is formulated. Section 6 contains a discussion on several issues related to the algorithm and the paper is concluded in Section 7.

2 Problem setting

The true system is assumed to be a Single-Input Single-Output (SISO) nonlinear time-invariant system described by

$$\mathcal{S} : y_t = P(Y_{t-1}, U_t, V_t) \tag{1}$$

where P is a time-discrete unknown nonlinear operator describing the plant. Here U_t, Y_{t-1}, and V_t are vectors of lagged input signals u_{t-1}, u_{t-2} ..., lagged output signals y_{t-1}, $y_{t-2}, \ldots$, and lagged values of a white disturbance signals v_t, $v_{t-1}, \ldots$, and y_t is the output at time t. Of course, (1) could be a consequence of a finite order difference equation.

The input signal is determined according to

$$\mathcal{C} : u_t = C(\rho, U_t, R_t, Y_t) \tag{2}$$

where the controller C is a linear or nonlinear operator on R_t, a vector of lagged reference signals r_t, r_{t-1}, $\ldots$, and Y_t a vector of lagged outputs, y_t, y_{t-1}, $\ldots$, and once again (2) could be a consequence of a finite order difference equation. The controller is parameterized by a parameter vector ρ, with $\rho \in \mathcal{D} \subset \mathbb{R}^n$. It is assumed that r_t is independent of the noise sequence v_t.

The plant, the controller and all closed-loop operators are assumed to be

3

smooth functions of all their arguments. See, e.g., (5) for more details on such smoothness assumptions. It is required that the closed-loop system in Figure 1 is incrementally BIBO stable, with initial conditions forgotten exponentially fast independent of excitation. Also the effect of a disturbance at time t is assumed to decay exponentially in time. This means that if two experiments are performed on the closed loop system with close reference signals and close noise sequences, then all other signals will also be close to another. This is a key requirement for the tuning algorithm to be applicable.

An experiment of length N on the closed loop system is obtained by applying a reference signal $\{r_t\}_{t=1}^{N}$. This reference and the noise sequence $\{v_t\}_{t=1}^{N}$ generate trajectories of u and y. It is the system's properties along these trajectories which influence the control performance. The proposed algorithm depends on the linearization of the plant, and controller, along these trajectories. At this stage, the reference signal $\{r_t\}_{t=1}^{N}$ is assumed to be given and fixed. However, it is a design variable which has to be chosen by the user and it will be discussed later on.

We denote by $u(\rho, r)$ and $y(\rho, r)$ the input and output of (1) in feedback with (2) as shown in Figure 1, i.e., $\{u_t\}_{t=1}^{N}$ and $\{y_t\}_{t=1}^{N}$ with input $\{r_t\}_{t=1}^{N}$ and parameter ρ.
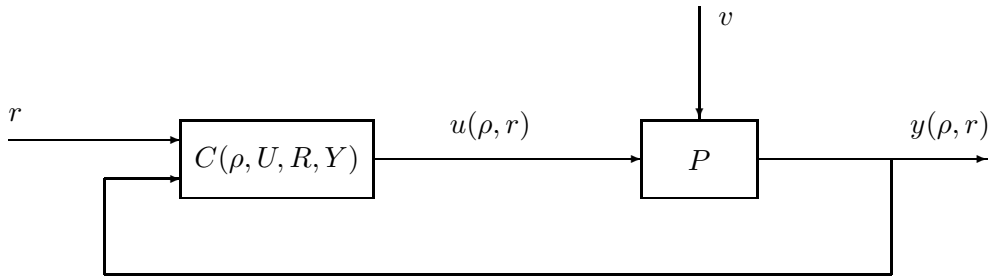

Fig. 1. The nonlinear control loop

Let y_d be the desired closed-loop response function to the reference signal r, i.e. $y_d = T_d(r)$ with T_d some possibly nonlinear reference model. The error between the achieved and the desired response is $\tilde{y}(\rho, r) = y(\rho, r) - y_d$. The control design objective can be any smooth function of $\tilde{y}(\rho, r)$, and it is possible to include also other signals, e.g., $u(\rho, r)$. For simplicity, the squared error is chosen, i.e., the tuning problem can be described as one of seeking

$$\rho^* = \arg \min_{\rho} J(\rho) \tag{3}$$

with

$$J(\rho) = \frac{1}{2} E \frac{1}{N} \sum_{t=1}^{N} [\tilde{y}_t(\rho, r)]^2. \tag{4}$$

4

The expected value is taken with respect to the probability distribution of the noise. For more general criteria including an input signal penalty, see (7; 6).

The problem that is addressed is the minimization of (4) with respect to the controller parameter vector ρ. Indeed, it is standard that one can seek a solution for ρ to

$$0 = J'(\rho) = \frac{1}{N} \sum_{t=1}^{N} E\left[\tilde{y}_t(\rho, r)\, \tilde{y}'_t(\rho, r)\right] \tag{5}$$

by taking repeated steps in the negative gradient direction

$$\rho[i+1] = \rho[i] - \gamma_i\, H_i^{-1}\, J'(\rho[i]) \tag{6}$$

where H_i is some appropriate positive definite matrix, typically an estimate of the Hessian of J and $\{\gamma_i\}$ is a sequence of positive numbers that determines the step size. Here $\rho[i]$ denotes the controller parameter vector ρ at iteration i.

The gradient $J'(\rho[i])$ cannot be obtained exactly. Instead, it has to be estimated and this contribution describes an algorithm for obtaining this estimate.

3 Heuristic explanation of the results

This section explains the idea of the gradient computation algorithm. The gradient estimate is obtained by performing a series of experiments using reference signals close to one another.

A straightforward way to obtain an estimate of the gradient of the output with respect to the controller parameters would be to perform a series of experiments using the same reference signal but with slightly changed controller parameters and then to form a numeric approximation of the gradient. The estimate of $y'_{\rho_j}(\rho, r)$, the derivative of $y(\rho, r)$ with respect to parameter j, is obtained by performing two experiments with controller $C(\rho, U, R, Y)$ and $C(\rho + \Delta\rho_j, U, R, Y)$, and the estimate becomes

$$\hat{y}'_{\rho_j}(\rho, r) = \frac{y(\rho + \Delta\rho_j, r) - y(\rho, r)}{|\Delta\rho_j|} \tag{7}$$

Consider now the change of the control signal due to a small parameter change

$$u(\rho + \Delta\rho_j, r) = u(\rho, r) + C'_{\rho_j}(\rho, U, R, Y)\Delta\rho_j + \partial C_y(\rho, U, R, Y)\Delta y + O(\Delta\rho_j^2) \tag{8}$$

where $O(x)/x < M$ when $x \to 0$ for some finite M. The operator $C'_{\rho_j}(\rho, U, R, Y)$ denotes the derivative of the output of the controller when its inputs $\{r_t\}_{t=1}^N$ and $\{y_t\}_{t=1}^N$ are unaltered. For each $j = 1, \ldots n$, it is a time function defined on $[1, N]$. The symbol $\partial C_y(\rho, U, R, Y)$ denotes a linear, time-varying operator describing the mapping from small perturbations in $\{y_t\}_{t=1}^N$ to the controller output function, given controller inputs $\{r_t\}_{t=1}^N$ and $\{y_t\}_{t=1}^N$. For future use, $\partial C_r(\rho, U, R, Y)$ and $\partial P_u(Y, U, V)$ are similarly defined. In Section 6 the linearized operators are explained further.

If, instead, the parameters are kept unchanged but the reference signal is changed slightly by Δr_j one obtains

$$u(\rho, r+\Delta r_j) = u(\rho, r)+\partial C_r(\rho, U, R, Y)\Delta r_j+\partial C_y(\rho, U, R, Y)\Delta y+O(\Delta r_j^2) \quad (9)$$

Assume that it is possible to choose Δr_j so that (9) equals (8), that is, $u(\rho + \Delta \rho_j, r) = u(\rho, r+\Delta r_j)$. Then the disturbance of the outputs, Δy, also becomes equal in the two equations (8) and (9). This means that

$$\partial C_r(\rho, U, R, Y)\Delta r_j = C'_{\rho_j}(\rho, U, R, Y)\Delta \rho_j$$

and, if the linear time-varying operators $[\partial C_r(\rho, U, R, Y)]^{-1}$ and $C'_{\rho_j}(\rho, U, R, Y)$ exist and if they are stable, then

$$\Delta r_j = [\partial C_r(\rho, U, R, Y)]^{-1}C'_{\rho_j}(\rho, U, R, Y)\Delta \rho_j \quad (10)$$

This trajectory $\{\Delta r_j\}_{t=1}^N$ is a small change of the reference signal giving the same effect (up to first order terms) on the input, and hence output, signals as if one had changed the control parameter. This means that an estimate of the derivative with respect to one of the parameters can be obtained by taking the difference of the outputs from two experiments with slightly different reference signals, specified by (10). Combining (7) and (10) gives a method for the estimation of the gradient by changing the reference signal instead of changing the controller parameter vector ρ.

The change of the reference signal Δr_j in (10) must be sufficient small so that the neglected terms in the Taylor expansions in (8) and (9) do not become too large. On the other hand, too small Δr_j yields a poor signal to noise ratio. This trade-off will be illustrated in the analysis in Section 5. Also, existence and stability of $[\partial C_r(\rho, U, R, Y)]^{-1}$ and $C'_{\rho_j}(\rho, U, R, Y)$ will be discussed later.

4 A small illustration

In this section a small illustration of the algorithm is given. In (2; 9), other examples can be found where the algorithm is used to tune a nonlinear con-

troller, respectively, for an inverted pendulum with a flexible transmission and for a chemical batch reactor.

The plant is described by the following equation

$$\dot{x}_t = x_t^2 + u_t$$
$$y_t = x_t + v_t. \tag{11}$$

where v_t is Gaussian white noise with standard deviation 10^{-5}. The signals are sampled with $T = 0.01$. The desired closed-loop system is a linear first order system with a pole in -0.005 and a unit DC-gain.

The set of nonlinear control laws is described by

$$u_t = -(K_1 y_t^2 + K_2 y_t^3) + r_t \tag{12}$$

which contains two parameters with initial values $[K_1, \; K_2] = [0.5, \; 0.5]$. In Figure 2 the closed-loop performance of the initial controller is depicted. Notice that the true plant and the assumed controller structure do not allow exact matching of the desired closed-loop response for any setting of the controller parameters, and accordingly the quality of the matching will be input dependent.
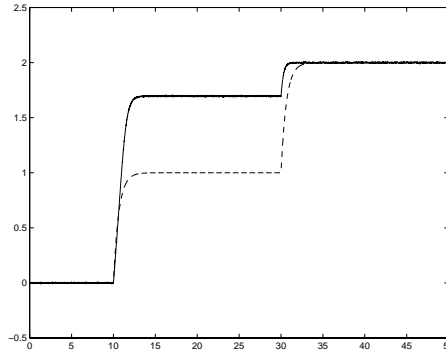

Fig. 2. Desired (dashed) and true (solid) plant output versus time when the system is controlled by the initial controller.

The control performance criterion is chosen as in (4) and the desired output signal y_d is given by the dashed line in Figure 2. The minimum of the criterion is computed by iterating (6) where $J'(\rho)$ is replaced by an estimate formed from (5), (7) and (10). According to (10) there is one trajectory Δr_j for each of the two control parameters. Using (12) $C'_{\rho_j}(\rho, U, R, Y)$ becomes

$$C'_{K_1} = -y_t^2 \tag{13}$$
$$C'_{K_2} = -y_t^3 \tag{14}$$

7

and
$$\partial C_r = 1 \qquad (15)$$
The derivative of the plant output with respect to the controller parameters is now obtained, as described in Section 3, by performing three experiments with reference signals r_t, $r_t + \Delta r_i^1$, and $r_t + \Delta r_i^2$ where

$$\Delta r_i^1 = -\mu_1^i y_t^2$$
$$\Delta r_i^2 = -\mu_2^i y_t^3$$

and where μ_j^i is a scaling constant chosen so that $\max_t |\Delta r_i^j| = \max_t |r_t|/20$.

In Table 1 the mean-square error J is shown for the initial controller and after each of the 3 iterations of (6).

Iteration #	$J(\hat{\rho})$
0	0.18
1	0.0097
2	0.0032
3	0.0029

Table 1
The value of the criterion of fit $J(\rho)$ for the initial controller and after each of three iterations.

The parameters of the final tuned controller are $[K_1, \ K_2] = [2.36, \ -0.429]$ and its performance is depicted in Figure 3. From the figure we make the following
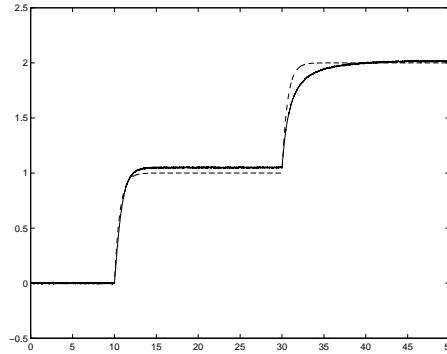

Fig. 3. Final tuned controller: Desired (dashed) and true (solid) plant output.

conclusions:

- The performance of the controller has been improved.
- The control is not perfect due to that the controller structure is not the optimal one for this plant. This is natural since the purpose of the proposed method is to tune a parameterized controller. Choosing the best control structure is a different (but important) topic.

8

- A different choice of reference signal would have given a different result. Hence, the choice of reference signal is an important design variable.

In the following section the algorithm will be described in somewhat more detail.

5 Analysis

Introduce the closed loop operators T and S as defined by the control loop of Figure 1,

$$y(\rho, r) = T(\rho, r, v) \tag{16}$$
$$u(\rho, r) = S(\rho, r, v) \tag{17}$$

Consider the change of the output signal due to a small parameter change, it can be described using the closed loop operator T

$$y(\rho + \Delta\rho, r) = T(\rho + \Delta\rho, r, v) = T(\rho, r, v) + T'_{\rho}(\rho, r, v)\Delta\rho + O(\Delta\rho^2) \tag{18}$$

or using the plant operator P and the change of the control signal u, giving

$$y(\rho + \Delta\rho, r) = y(\rho, r) + \partial P_u(Y, U, V)\Delta u + O(\Delta\rho^2) \tag{19}$$

For a given $\Delta\rho$, these two expressions are equal, and by combining them one obtains in the limit $\Delta\rho \to 0$

$$y'_{\rho}(\rho, r) = T'_{\rho}(\rho, r, v) = \partial P_u(Y, U, V)u'_{\rho}(\rho, r) \tag{20}$$

An expression of $u'_{\rho}(\rho, r)$ follows from (8) if $\Delta\rho \to 0$

$$u'_{\rho}(\rho, r) = C'_{\rho}(\rho, U, R, Y) + \partial C_y(\rho, U, R, Y)\, y'_{\rho}(\rho, r) \tag{21}$$

Combining (20) and (21) gives an expression for the derivative of the output signal

$$y'_{\rho}(\rho, r) = T'_{\rho}(\rho, r, v) =$$
$$\frac{1}{1 - \partial P_u(Y, U, V)\partial C_y(\rho, U, R, Y)}\partial P_u(Y, U, V)C'_{\rho}(\rho, U, R, Y) \tag{22}$$

That is, $y'_{\rho}(\rho, r)$ is obtained by filtering $C'_{\rho}(\rho, U, R, Y)$ through a time-varying linear filter, as depicted in Figure 4. [Notice that the incremental BIBO stability assumption of Section 2 guarantees stability of the linear time-varying
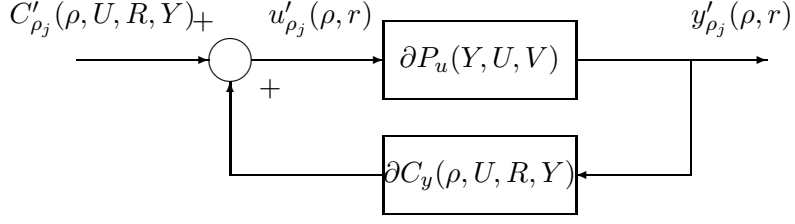
9

Fig. 4. Generation of $u'_{\rho_j}(\rho, r)$ and $y'_{\rho_j}(\rho, r)$

feedback loop in Figure 4] We will now describe how this filtering will actually be obtained by differencing the outputs of two experiments.

Consider the output from a first experiment on the plant, where the signals from this experiment are indexed with "1".

$$y_1(\rho, r) = T(\rho, r, v_1) \tag{23}$$

Suppose that we perform a second experiment on the system with a slightly perturbed reference signal $r + \Delta r$. Denote the corresponding output by $y_2(\rho, r + \Delta r)$. Then, using the small signal assumption on Δr and a smoothness assumption on T, we have that

$$
\begin{aligned}
y_2(\rho, r + \Delta r) &= T(\rho, r + \Delta r, v_2) = \\
&\quad T(\rho, r + \Delta r, v_1) + \partial T_v(\rho, r + \Delta r, v_1)\Delta v + O(\Delta v^2) \\
&= T(\rho, r, v_1) + \partial T_r(\rho, r, v_1)\Delta r + \partial T_v(\rho, r + \Delta r, v_1)\Delta v + O(\Delta r^2) + O(\Delta v^2)
\end{aligned}
\tag{24}
$$

where v_2 is the disturbance during the second experiment and $\Delta v = v_2 - v_1$. The last terms, $O(\Delta r^2)$ and $O(\Delta v^2)$ describe the approximation due to the Taylor expansion in Δr and Δv. Combining (23) and (24) gives

$$
\begin{aligned}
y_2(\rho, r + \Delta r) - y_1(\rho, r) &= \\
&\partial T_r(\rho, r, v_1)\Delta r + \partial T_v(\rho, r + \Delta r, v_1)\Delta v + O(\Delta r^2) + O(\Delta v^2)
\end{aligned}
\tag{25}
$$

An expression of $\partial T_r(\rho, r, v_1)$ using input-output operators can be obtained with calculations similar to those which gave $T'_\rho(\rho, r, v)$ in (22),

$$\partial T_r(\rho, r, v_1) = \frac{1}{1 - \partial P_u(Y, U, V_1)\partial C_y(\rho, U, R, Y)}\partial P_u(Y, U, V_1)\partial C_r(\rho, U, R, Y) \tag{26}$$

Now, if Δr is chosen according to (10) as Δr_j for some given j, it follows from (25) and (26) that $\partial T_r(\rho, r, v_1)\Delta r_j$, the first term of (25), equals the $y'_{\rho_j}(\rho, r)\Delta \rho_j$: see (22). The rest of the terms in (25) give a disturbance to the estimate.

It follows from the problem setting that the noise sequence v_2 in the second experiment is different from v_1. This gives the disturbance terms $\partial T_v(\rho, r +$

$\Delta r, v_1)\Delta v$ and $O(\Delta v^2)$ in (25). The first one of these also exists if the system is linear, but the second one is a nonlinear effect due to the fact that different noise sequences give slightly different trajectories and linearizations. It follows that if the noise level is low and if the closed-loop performance is fairly good so that the noise sequence does not influence the trajectories substantially, then the term $O(\Delta v^2)$ should be small.

As with the nonlinear contribution of the noise, the remaining disturbance term in (25), $O(\Delta r^2)$, is also due to that the trajectories become slightly different when r is changed between different experiments. However, since the size of Δr is a user-designed parameter, it can be chosen so that a good balance is obtained between signal to noise ratio and the bias caused by $O(\Delta r^2)$ and $O(\Delta v^2)$.

To conclude, by choosing Δr to be a scaled version of (10) and by performing two experiments with r and $r + \Delta r$ and then, using (25), one obtains an estimate of y'_{ρ_j}.

We can now formulate the tuning algorithm.

5.1 Experimental generation of the gradient

If the local linear $[\partial C_r(\rho, U, R, Y)]^{-1}$ operator exists and if it is stable, define the signals

$$\tilde{r}_j(\rho) = [\partial C_r(\rho, U, R, Y)]^{-1} \, C'_{\rho_j}(\rho, U, R, Y) \ \ \text{for} \ \ j = 1, \cdots, n. \qquad (27)$$

which can be computed given the data trajectories r and y from the first experiment.

At each iteration step i of the controller parameter tuning, use $n + 2$ experiments with the fixed controller $C(\rho[i], U, R, Y)$. The corresponding reference signals are

$$\begin{cases} r_i^1 & = r, \\ r_i^2 & = r + \mu_1^i \, \tilde{r}_1(\rho[i]), \\ \vdots & \vdots \ \vdots \\ r_i^{n+1} & = r + \mu_n^i \, \tilde{r}_n(\rho[i]), \\ r_i^{n+2} & = r \end{cases} \qquad (28)$$

with the size of the scalars μ_j^i, $j = 1, \cdots, n$, chosen so that a good trade-off between bias and variance in the estimate is obtained. The corresponding

11

output and disturbance signals are, respectively, denoted by y_i^k and v_i^k, for $k = 1, \cdots, n+2$. We have the following estimates

$$\tilde{y}(\rho[i], r) = y_i^{n+2} - y_d \tag{29}$$

$$\hat{y}'_{\rho_j}(\rho[i], r) = \frac{1}{\mu_j^i}\left(y_i^{j+1} - y_i^1\right) \tag{30}$$

for $j = 1, \cdots, n$. Thus $\hat{y}'_{\rho_j}(\rho[i], r)$ is a noisy estimate of $y'_{\rho_j}(\rho[i], r)$. Note that the noise terms in $\hat{y}'(\rho[i], r)$ are independent of the noise of $\tilde{y}(\rho[i], r)$. This is the motivation for the $(n+2)$-th experiment. In a low noise situation, if an additional small bias contribution can be accepted, this experiment can be omitted with its corresponding output replaced by the output of the first experiment.

The estimate of the gradient of J is then obtained as

$$\hat{J}'(\rho[i]) = \frac{1}{N}\sum_{t=1}^{N}\tilde{y}_t(\rho[i], r)\,\hat{y}'_t(\rho[i], r). \tag{31}$$

A biased estimate $\hat{H}_i$ of the Hessian is obtained using

$$\hat{H}_i = \frac{1}{N}\sum_{t=1}^{N}\hat{y}'_t(\rho[i], r)\,[\hat{y}'_t(\rho[i], r)]^T. \tag{32}$$

An unbiased estimate of the Hessian is of less priority than an unbiased estimate of the gradient. The reason for this is that the estimate (32) of the Hessian is always positive definite so that the parameter update is always in a descent direction of the criterion. This is enough to guarantee convergence to the true minimum of the control performance.

If the control cost includes a control signal penalty an estimate of $u'(\rho[i], r)$ is also necessary. It can be obtained by combining (21) with (30). The iterative algorithm is obtained by replacing $J'(\rho[i])$ by $\hat{J}'(\rho[i])$ and H_i by $\hat{H}_i$ in (6).

The algorithm is now described with the following steps:

(1) Start with a stabilizing controller $C(\rho[0], U, R, Y)$.
(2) With the current controller $C(\rho[i], U, R, Y)$ in the loop, perform one experiment on the actual system.
(3) Generate the signals $\tilde{r}_j(\rho[i])$ as in (27).
(4) Perform n experiments as in (28).
(5) Perform the $n+2$th experiment and compute $y(\rho[i], r) - y_d$.
(6) Compute the estimates $\hat{y}'_{\rho_j}(\rho[i], r)$ of the gradient signals $y'_{\rho_j}(\rho[i], r)$ as shown in (30).
(7) Compute $\hat{J}'(\rho[i])$ and $\hat{H}_i$ as shown in (31) and (32).

(8) Update the parameter vector using

$$\rho_{i+1} = \rho_i - \gamma_i \, \hat{H}_i^{-1} \, \hat{J}'(\rho[i]) \qquad (33)$$

(9) If further tuning is necessary, goto Step 2.

6   Discussion

- Exactly as in the linear IFT, the reference signal must be chosen with great care. The controller will only be tuned for those signals used in the experiments. The general recommendation is to choose a "typical" reference signal which excites the important modes.
- The controller optimization procedure converges to a local minimum of the design criterion under the condition of bounded signals, i.e. it is assumed that stability is preserved while iterating. The variable step size is also the principal ingredient in proving convergence of the algorithm in the presence of process noise. These features are analogous to the linear version of IFT and we refer to (7; 6).
- The initial state of the system will influence the estimate. Because of the assumption of exponential stability, the extent of its influence will decrease with increased experimental duration.
- The requirement of small deviations of the reference signals between the different experiments in (28) might give poor excitation of the system. This is an inherited problem of the proposed method due to that it relies purely on experimental data and does not use any model of the plant. The size of the deviations is a trade-off between noise sensitivity and bias due to nonlinearity. If the linearization of the plant is changing slowly as function of the reference signal during an experiment, which is common in many practical problems, then the excitation can be made larger. Note also that in real problems, where we always have nonlinear plants, a linear control design is made at some signal level with a limited validity outside the design conditions. Hence, also in the linear design case we have a similar problem.

  Also, in the proposed nonlinear IFT setting, poor signal to noise ratio can always be compensated by performing longer experiments. By repeating the reference signal several times it is possible to make the noise contribution arbitrarily small.
- The linear IFT is easily obtained. Assume that (1) and (2) reduce to

$$
\begin{aligned}
y_t &= P \, u_t + v_t, \\
u_t &= C(\rho, U, R, Y) = C_r(\rho) r_t + C_y(\rho) y_t
\end{aligned}
\qquad (34)
$$

  where P is the transfer function describing the plant, and $C_r(\rho)$ and $C_y(\rho)$ are linear transfer functions.

Then, considering a parameter ρ_j in $C_r(\rho)$, and since $\partial C_r(\rho) = C_r(\rho)$ (27) simplifies to

$$\tilde{r}_j(\rho) = \frac{1}{C_r(\rho)}\frac{dC_r(\rho)}{d\rho_j}r_t \qquad (35)$$

and from (22) and the fact that linear time-invariant filters commute it now follows that

$$y'_{\rho_j}(\rho, r) = \frac{dC_r(\rho)}{d\rho_j}\frac{1}{C_r(\rho)}G_c r_t \qquad (36)$$

where

$$G_c = \frac{PC_r}{1 - PC_y}$$

is the closed loop transfer function.

This means that only one experiment with reference signal r_t is necessary to obtain the derivative with respect to all the parameters in $C_r(\rho)$; the result is filtered through $\frac{dC_r(\rho)}{d\rho_j}\frac{1}{C_r(\rho)}$ to obtain the result with respect to a specific parameter ρ_j.

In the same way, to obtain the derivative for a parameter in $C_y(\rho)$ one obtains

$$y'_{\rho_j}(\rho, r) = \frac{dC_y(\rho)}{d\rho_j}\frac{1}{C_r(\rho, q)}G_c y_t$$

which is an experiment with y_t as reference signal. Hence, estimates of $y'(\rho, r)$ (and $u'(\rho, r)$) can be obtained by only two experiments thanks to the fact that the operators commute.

This gives a different description of the Model-Free Tuning applied to linear systems than the one given in (6; 7).

- To generate $\tilde{r}_j$ one must filter the gradient signals through $[\partial C_r(\rho, R, Y)]^{-1}$ which has to exist and to be stable. The linear filter equation looks like

$$\Delta u_t = \partial C_r(\rho, R, Y)\Delta r_t = \frac{T_0 + T_1 q^{-1} + \cdots + T_m q^{-m}}{S_0 + S_1 q^{-1} + \cdots + S_n q^{-n}}\Delta r_t$$

where q^{-1} is the backward shift operator, $q^{-1}u_t = u_{t-1}$. The coefficients $T_0, \ldots, T_m$ and $S_0, \ldots, S_n$ are evaluated along the trajectories of the experiments which makes them time dependent. The inverse becomes

$$\Delta r_t = \frac{S_0\Delta u_t + S_1\Delta u_{t-1} + \cdots + S_n\Delta u_{t-n} - T_1\Delta r_{t-1} + \cdots - T_m\Delta r_{t-m}}{T_0}$$

and it exists if $T_0 \neq 0$ for all t. Hence, an existence test can easily be included in the algorithm. In the event of instability, there is in the linear case an easy work-around using non-causal filtering which might be possible also in some nonlinear problems, but in general it is hard to apply. A simple, practical, solution would be to include a test of the obtained Δr in the algorithm so that one can stop the tuning in such cases.

7 Conclusions

In this paper, we have presented a nonlinear extension of the controller optimization method proposed in (7; 6). It is shown that one can obtain an estimate of the gradient experimentally using $n + 2$ experiments on the actual system with slightly different reference signals.

If the initial controller is fairly good and the noise level is moderate so that the output trajectories from different experiments are close to another then the bias in the gradient estimate can be expected to be small.

The algorithm offers a safer alternative than if the gradient is estimated by changing the parameter values. The tools used in the algorithm are easy to implement or readily available in commercial software packages. A numerical simulation example with an experimental-based generation of the gradient illustrates the algorithm.

References

[1] B. Codrons and B. Ceysens. Iterative identificationless control design. *Journal A*, 38:26–30, 1997.

[2] B. Codrons, F. De Bruyne, M. De Wan, and M. Gevers. Iterative feedback tuning of a nonlinear controller for an inverted pendulum with a flexible transmission. In *Proceedings of the IEEE International Conference on Control Applications*, pages 1281–1285, Trieste, Italy, 1998.

[3] F. De Bruyne, B.D.O. Anderson, M. Gevers, and N. Linard. Iterative controller optimization for nonlinear systems. In *Proceedings of the Conference on Decision and Control*, pages 3749–3754, San Diego, USA, 1997.

[4] F. De Bruyne and P. Carrette. Synthetic generation of the gradient for an iterative controller optimization method. In *CD Proceedings of the European Control Conference*, Brussels, Belgium, 1997. TH-A-F-2 (52).

[5] C.A. Desoer and M. Vidyasagar. *Feedback Systems: Input and Output Properties*. Electrical Science Series, Academic Press, New York, 1975.

[6] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin. Iterative feedback tuning: theory and applications. *IEEE Control Systems Magazine*, 18:26–41, 1998.

[7] H. Hjalmarsson, S. Gunnarsson, and M. Gevers. A convergent iterative restricted complexity control design scheme. In *Proceedings of the Conference on Decision and Control*, pages 1735–1740, Orlando, Florida, 1994.

[8] J. Sjöberg and M. Agarwal. Nonlinear controller tuning based on linearized time-variant model. In *Proc. of American Control Conference, Albuquerque, New Mexico, 4-6 June*, 1997.

[9] J. Sjöberg and M. Agarwal. Tuning neural controllers for trajectory track-

ing in batch processes. In *Proceedings of the IMACS International Multiconference: CESA'98, Computational Engineering in Systems Applications*, pages 449–454, Hammamet, Tunisia, 1998.