



A Supervisor for Control of Mode-switch Processes*

R. A. HILHORST,[†] J. VAN AMERONGEN,[†] P. LÖHNBERG[†]
and H. J. A. F. TULLEKEN[‡]

Controllers for processes which frequently return to a number of modes of operation should be adapted through mode recognition.

Key Words—Adaptive control; automatic tuning; learning systems; time-varying systems.

Abstract—Many processes operate only around a limited number of operation points. In order to have adequate control around each operation point, an adaptive controller could be used. When the operation point changes often, a large number of parameters would have to be adapted over and over again. This makes application of conventional adaptive control unattractive, which is more suited for processes with slowly changing parameters. Furthermore, continuous adaptation is not always needed or desired. An extension of adaptive control is presented, in which for each operation point the process behaviour can be stored in a memory, retrieved from it and evaluated. These functions are co-ordinated by a “supervisor”. This concept is referred to as a supervisor for control of mode-switch processes. It leads to an adaptive control structure which quickly adjusts the controller parameters based on retrieval of old information, without the need to fully relearn each time. This approach has been tested on experimental set-ups of a flexible beam and of a flexible two-link robot arm, but it is directly applicable to other processes, for instance, in the (petro) chemical industry.

1. INTRODUCTION

MANY PROCESSES CANNOT be controlled adequately by a fixed controller. For appropriate control, an adaptive controller or even a variable controller structure is needed. When the process operates in a limited number of operating points, a limited number of controllers suffices. In

practical situations a controller will not only yield satisfactory control performance in the operation point, but also in the neighbourhood of this operating point. The set of operating conditions where one controller performs well is called a mode. In practice there are several processes which exhibit this behaviour and operate in a limited number of modes only. Processes which frequently return to an earlier seen mode will be referred to as mode-switch processes (Hilhorst *et al.*, 1991). Such processes are common, for instance, in the process industry and in robotics. Mode-switch behaviour is encountered in a chemical reactor in which the yield and quality of the product has to be optimized to meet market demands, or in a robot which has to transport a limited number of payloads with different masses.

In order to meet the control demands in each operating point, the use of a conventional adaptive controller (Åström and Wittenmark, 1989) could be considered. However, for mode-switch processes, the time needed for (conventional) adaptation may be too long, that is, larger than the residence time in a process mode. For instance, the closed-loop process signals may not be sufficiently exciting. Although the addition of test signals can increase the adaptation speed, it disturbs the process and hence results in performance loss. On the other hand, it does not seem to be necessary to repeat the whole adaptation cycle each time the process returns to a certain process mode. The problem is that conventional adaptive controllers forget the useful information which was available before.

The problem of forgetting useful information can be solved by construction of a composite model (Skeppstedt *et al.*, 1992) of the process. This implies that each time the process enters a new mode of operation, its operation point and related model and controller are stored. Using

*Received 16 April 1992; revised 16 February 1993; revised 15 July 1993; received in final form 15 October 1993. The original version of this paper was presented at the IFAC Symposium on Intelligent Tuning and Adaptive Control which was held in Singapore, during 15–17 January 1991. (The paper has been completely rewritten based on recent results and insights.) The Published Proceedings of this IFAC Meeting may be ordered from Elsevier Science Limited, The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, U.K. This paper was recommended for publication in revised form by Associate Editor A. Annaswamy under the direction of Editor C. C. Hang. Corresponding author Dr P. Löhnberg. Tel. +31 53 892787; Fax +31 53 340045; E-mail lbg@rt.el.utwente.nl.

[†]Control, Systems and Computer Engineering Group, Department of Electrical Engineering, University of Twente, and Mechatronics Research Centre Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands.

[‡]Department of Mathematics and Systems Engineering, Operational Analysis and Optimization Section, Koninklijke/Shell-Laboratorium (Shell Research B.V.), P.O. Box 3003, 1003 AA Amsterdam, The Netherlands.

this composite model, a gain scheduling controller (Åström and Wittenmark, 1989) can be constructed. Gain scheduling results in fast adaptation of control parameters, as controllers related to each operating point are stored. However, the construction of a composite model as described by Skeppstedt *et al.* (1992) and hence the application of a gain scheduling controller is restricted to processes where the process variables related to the changes in dynamics are measurable. If this is not the case, use can be made of a so-called Multi-Model Adaptive Controller (MMAC) approach as studied by Athans *et al.* (1977) and a similar approach described by Moose *et al.* (1978). Unfortunately, in its pure form this approach has severe drawbacks like adaptation stopping as described by Athans *et al.* (1977) and its non-transparency with respect to tuning the adaptation speed (Moose *et al.*, 1978), that is, no clear rules are present to balance mode-tracking with noise-insensitivity. Also, recent work by Lund *et al.* (1991) shows that the successfulness of application of such schemes can be hampered by the lack of distinction capability between the different models.

In this paper a supervisor for control of mode-switch processes is proposed which is also applicable when the process variables related to changes in dynamics are not measurable. Furthermore, a method for mode detection is presented which solves the problems described above. This approach has the advantage that only recognition of the new mode, and no identification of the process, is needed when the process returns to an earlier visited recognized mode of operation or when the control criterion has changed. A performance monitor can take care of restarting the adaptation whenever necessary.

The supervisor for control of mode-switch processes will be described with the main emphasis on the retrieval of stored information. In Section 2 the modeling of mode-switch processes is presented. Definitions of a "mode" and "mode switch" are given. In Section 3 the structure of the supervisor is discussed. Furthermore, an overview of the functions to be executed by the supervisor is given. Section 4 contains a detailed discussion about mode recognition. These ideas have become realisable, also for systems with relatively fast dynamics, due to the availability of high-speed parallel computers. Experiments are described in Section 5 where the supervisor for control has been tested on experimental set-ups of a flexible beam as well as of a two-link flexible robot arm. Finally, conclusions are drawn in Section 6.

2. MODELLING OF MODE-SWITCH PROCESSES

Many processes vary with time and/or are nonlinear. Such a process, referred to as P, can be described by a nonlinear time-variant model:

$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, \mathbf{w}), \\ \mathbf{y} &= g(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta}, \mathbf{v}),\end{aligned}\quad (1)$$

where \mathbf{x} is a vector of plant states, \mathbf{y} is a vector of process outputs, \mathbf{u} is a vector of control inputs, $\boldsymbol{\theta}$ is a vector of time-variant process parameters, \mathbf{w} is a vector of system disturbances, \mathbf{v} is a vector of measurement disturbances, and f and g are nonlinear, time-invariant functions. Such a complete description of the process in the form of equation (1) is generally not available. Often that is also not needed, as in many cases processes are operated only for small deviations around a limited number of nominal operating points. These processes will be referred to as mode-switch processes. Moreover, around such an operating point the dynamics may be approximated by a linear model such that a fixed linear controller based on this model performs well in the neighbourhood of this operating point. An operation point is identified by an operation vector $\boldsymbol{\omega}$, and a nominal operation point vector by $\bar{\boldsymbol{\omega}}$. This nominal operation vector $\bar{\boldsymbol{\omega}}$ consists of elements which influence the linearized dynamics. Typically, the nominal operation vector $\bar{\boldsymbol{\omega}}$ will contain information about the nominal state vector $\bar{\mathbf{x}}$, nominal input vector $\bar{\mathbf{u}}$, nominal process parameter vector $\bar{\boldsymbol{\theta}}$, nominal system disturbance vector $\bar{\mathbf{w}}$, and nominal measurement disturbance vector $\bar{\mathbf{v}}$:

$$\bar{\boldsymbol{\omega}} = \{\bar{\mathbf{x}}, \bar{\mathbf{u}}, \bar{\boldsymbol{\theta}}, \bar{\mathbf{w}}, \bar{\mathbf{v}}\}.\quad (2)$$

Joining the possible values of the operation vector $\boldsymbol{\omega}$, a space of operation Ω can be defined. Normally, only a limited number of elements of the operation vector is measurable. For instance, process parameters and system disturbances are usually not measurable or known in advance. This problem is addressed later in this section.

Around an operation point, pointed to by an operation vector $\boldsymbol{\omega}$, a linear model $M_{\boldsymbol{\omega}}$ for local perturbational dynamics can be derived. Such a linear description can be obtained analytically if (1) is available and sufficiently smooth, or otherwise by identification of the parameters of a linear process model. By means of system identification, a number of linear models of the process can be obtained, each for a different nominal operation point (mode centres). In order to distinguish between the mode centres, they will be denoted by $\boldsymbol{\omega}_i$, $1 \leq i \leq n$, where n is the number of mode centres. The process model obtained for the i th mode centre will be denoted

by M_i . Based on the model M_i , a linear controller C_i can be designed which satisfies the control criterion.

A special case is that in which the operation vector ω consists only of a process parameter vector θ , which jumps between a limited number of values. Such a process will be referred to as a finite mode-switch process, as the dynamics can be exactly covered by a finite number of linear models. If the process parameter vector moves over a range of values, the process will be considered as a generalized mode-switch process.

Based on the selected mode centres ω_i , $1 \leq i \leq n$, the operation space Ω can be divided into a number of modes. Such a mode can be determined by comparing the properties of a mode centre ω_i with the properties related to an arbitrary operation point ω . These differences in properties can be related to a measure of "distance". In the following, three possible mode definitions (i.e. divisions of the operation space) will be given, and their applicability in practice will be discussed.

If a set of mode centres is known, and if for each mode centre ω_i a controller C_i is designed, then for each controller C_i a subset Ω_i^c in the operation space can be determined in which this i th controller operates well:

$$\Omega_i^c = \{\omega \in \Omega \mid \text{closed-loop plant behaviour using controller } C_i \text{ satisfactory in some sense}\}. \quad (3)$$

Such a set is called a control mode. A division of the operation space Ω in control modes is difficult to make in practice, because it requires that the operation vector is fully measurable and that it is known for each operation point that the controller C_i results in satisfactory behaviour. Also acquiring the knowledge of satisfactory behaviour for each controller can be cumbersome as shown in the design of an adaptive auto pilot for ships by Reid and Williams (1978). Because the aim is to increase the speed of adaptation by using valuable knowledge collected in the past, this solution should be rejected.

When the division of the operation space in control modes is impossible because the exact size of the control modes is not known, approximations have to be made. In a sufficiently small neighbourhood of a mode centre ω_i , any locally well-designed controller C_i performs well. This neighbourhood will be referred to as an operation mode, which can be defined as

$$\Omega_i^o = \{\omega \in \Omega \mid d_1(\omega, \omega_i) \leq \epsilon_i\}, \quad (4)$$

where d_1 is a metric, and ϵ_i is a positive real-valued parameter specified by the user. Such a technique is well known in gain scheduling theory. If the operation vector could be measured noise-free, an obvious choice is to take d_1 equal to the Euclidean norm, that is

$$d_1(\omega, \omega_i) = \|\omega - \omega_i\|, \quad (5)$$

which is a popular gain scheduling function. Also the construction of a composite model as proposed by Skeppstedt *et al.* (1992) can be recognized as being similar to making a division of the operation space in operation modes. However, the full operation vector will seldom be measurable. Therefore application of this method will be limited to special cases.

Another definition concerns the notion of model mode. In a sufficiently small neighbourhood of a mode centre ω_i , the process dynamics encountered will resemble the dynamics described by the model M_i . Furthermore, it is likely that in this neighbourhood the controller C_i performs well. This neighbourhood will be referred to as a model mode Ω_i^m which can be defined as

$$\Omega_i^m = \{\omega \in \Omega \mid d_2(P, M_i) \leq \epsilon_i\}, \quad (6)$$

where d_2 is a distance function, P is the true process, M_i the model related to mode centre ω_i , and ϵ_i is a positive real-valued parameter specified by the user. The distance function d_2 is a measure of the model resemblance. One way to determine the distance function d_2 is to run all models in parallel with the process, like for instance performed in the field of process fault-recognition (Willsky, 1976; Isermann, 1984; Tzafestas and Watanabe, 1990). Based on the process measurements, each model M_i at each time instant k produces a prediction of the output \hat{y}_k^i . From each predicted model output \hat{y}_k^i and from the measured process output y_k , an error e_k^i can be computed. From the computed errors, a distance between the process P and each model M_i can be defined as

$$d_2(P, M_i) = \sum_k (e_k^i)^T Q e_k^i, \quad (7)$$

where Q is a symmetric non-negative matrix which weights the different output elements.

Comparing the definitions of operation mode and model mode shows that the latter poses less restrictions on the process. The requirement of running models in parallel with the process can be met by the use of fast parallel hardware, for example, transputers (INMOS, 1988). Therefore, the definition of model mode will be considered in the following and this will be referred to as a mode when no ambiguity occurs.

The tolerances ϵ_i and the number n of operation points ω_i , $1 \leq i \leq n$, should be chosen such that the union of all modes of operation will cover the whole space of operation Ω and such that a good control of the process is obtained over the whole space of operation Ω , that is

$$\bigcup_i \Omega_i = \Omega, \quad 1 \leq i \leq n. \quad (8)$$

Equation (8) is a condition for having adequate control when the process changes from one mode centre to another one. On the other hand, the modes should preferably not overlap each other too much. If overlap occurs, the operation vector ω corresponding to mode Ω_i will be classified to mode Ω_i if the following holds:

$$d(P, M_i) \leq d(P, M_j) \quad \text{for all } j \in [1, \dots, n]. \quad (9)$$

If the equality holds, additional criteria have to be used to distinguish between the two modes Ω_i and Ω_j .

If the current process mode is given by Ω_i , then a mode switch occurs at an instant later in time when (9) becomes false for some mode index $j (i \neq j)$, that is, the switching rule is

$$d(P, M_i) < d(P, M_j). \quad (10)$$

The procedure described above makes a division of the operation space Ω into a number of modes such that each operation point is contained in a mode and that ambiguity is reduced to the edges only.

3. SUPERVISOR

The mode-switch property of a process can be used for (quick) adjustment of the controller parameters. This can be done by storing the obtained information for each mode centre ω_i in a memory. This information will usually consist of a linear model M_i and an associated linear controller C_i . When the process returns to a previously seen operation point, the controller can be retrieved and installed in the closed loop. In order to recognize whether the process returns to a previously seen mode of operation, a mode detector should be designed. To meet the requirements for industrial usage, such a system should be transparent enough for operators to be willing to work with it. This can be ensured by adopting a supervisory structure approach taken by Årzén (1987, 1989) and by Isermann and Lachmann (1985) and by adapting it for mode-switch processes. Such a supervisory structure has been proposed by Jansma and Tulleken (1986) and by Tulleken (1992). These ideas have been further elaborated by Hilhorst (1992) yielding a supervisory structure consisting

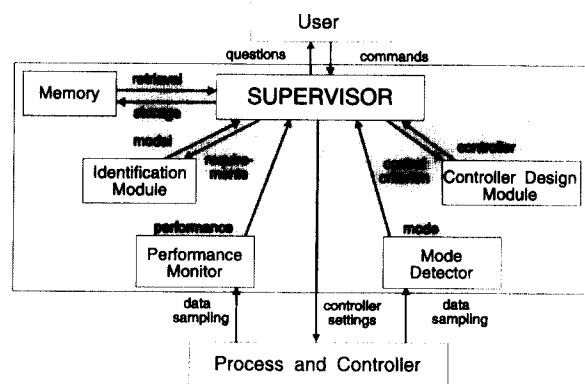


FIG. 1. Supervisor for control of mode-switch processes.

of a supervisor, memory, mode detector and performance monitor as shown in Fig. 1.

It can be seen from the scheme that the supervisor communicates with the user. This supervisor has two strategies: an automatic and a manual one. In the automatic strategy, the proposals made by the supervisor are implemented directly without interference of an operator. In the manual strategy, process operators have the opportunity to over-rule the proposals made by the supervisor or to choose other control criteria. The resulting control behaviour at the various modes of operation can be inconsistent, since different operators do not react in the same way when a particular control situation occurs. Moreover, the changes in process dynamics can occur so fast (for instance, in the case of a robot) that constant interaction between operator and supervisor results in performance loss. Therefore, in these cases the automatic strategy may be preferred.

It can also be seen from the scheme that the supervisor co-ordinates the monitoring, mode detection, retrieval, storage and maintenance of information in the memory, the start of identification and the controller design. Based on measurements and on a performance criterion provided by the user, the performance monitor calculates a performance index over a fixed interval of time. This performance index is used by the supervisor to evaluate the control performance of the current controller. Using the process measurements and the process mode information stored in the memory, the detector determines which mode resembles the current process conditions best. From the information provided by the performance monitor and by the mode detector, the supervisor decides whether or not to start a new identification cycle, or to retrieve a controller from memory. In the manual strategy, the installation of the controller has to be carried out, or at least approved by, the operator. In the automatic strategy, changing

from one controller to another occurs automatically.

If a new identification cycle is started—on the basis of requirements and restrictions stated by the user—the supervisor proposes an identification experiment, that is, a test signal and identification period. If the experiment is approved by the user, the identification module (e.g. Tullenken, 1992) estimates the model parameters. The resulting model is used by the control design module. Based on the model and on the control criterion stated by the user, the controller design module designs a controller. Subsequently, the controller and model are stored together in the memory, and the new controller is installed in the closed loop. In order to keep the memory manageable, data related to less successful or superfluous controllers should be forgotten. These requirements should lead to a system in which adaptation of controller parameters can occur fast, and in which conventional adaptation is used only at the instants when it is really needed. In the following the mode detector will be discussed in more detail.

4. MODE DETECTOR

4.1. Definitions and goals

As a result of changes in process parameters, disturbances or setpoints, the process mode may change. This requires a method which detects whether the process mode has changed and decides whether it is attractive to adapt the controller parameters. Mode recognition is the task of identifying the current mode of operation. Mode-switch detection is a subsequent task, which establishes whether or not the current process mode differs from the previous process mode. These two tasks are performed by the mode detector shown in the supervisory structure of Fig. 1. On the basis of the information provided by the mode detector, the supervisor decides whether the controller parameters should be adapted. If the method is quick in detecting mode switches, then the presence of noise may often give rise to wrong detections. On the other hand, if the method is made insensitive to noise, mode switches may not be detected fast enough. As both situations are undesirable, a good balance between noise sensitivity and mode tracking is essential.

Finally, when the mode detector establishes a mode switch, the controller parameters should be adjusted. In order to prevent bumps in the control signal, bumpless transfer was applied.

4.2. Mode recognition

In order to make a good balance between noise-insensitivity and mode-tracking, effective

use can be made of the ideas developed by Fortescue *et al.* (1981) in the field of least-squares estimation. In order to apply these ideas to the mode recognition problem, they will first be applied to finite mode-switch processes in Section 4.2.1 and second to generalized mode-switch processes in Section 4.2.2.

4.2.1. *Finite mode recognition.* If a finite mode-switch process P is in mode Ω_i , and if a Gaussian measurement disturbance ψ is present, then the process output y at time instant k can be written as

$$y_k = \hat{y}_k^i + \psi_k, \quad (11)$$

where \hat{y}_k^i is the output of model M_i . For each model M_i , the error e_k^i between model output \hat{y}_k^i and process output y_k can be determined, that is

$$e_k^i = y_k - \hat{y}_k^i. \quad (12)$$

When the process is in mode Ω_i , the error e_k^i is equal to the measurement disturbance, that is

$$e_k^i = \psi_k. \quad (13)$$

The expected norm of the error e_k^i is given by

$$E\|e_k^i\|^2 = E\{(e_k^i)^T e_k^i\} = \begin{cases} \sigma_\psi^2, & \text{if process is in mode } \Omega_i \\ \sigma_\psi^2 + \sigma_i^2, & \text{otherwise,} \end{cases} \quad (14)$$

where σ_ψ^2 is the variance due to the Gaussian disturbance ψ , and σ_i^2 is the additional variance due to the error between the process and model M_i . Equation (14) shows that the expected norm of the error is minimal for the model M_i related to the current process mode Ω_i . Hence, one straightforward method for mode recognition is by weighting the sum of errors and to select that model M_i for which the sum is minimal. This can be done by taking the distance $d(P, M_i)$ equal to:

$$d(P, M_i) = \Sigma_k^i \quad (15)$$

where the weighted sum of squared errors

$$\Sigma_k^i = \lambda_k^f (\Sigma_{k-1}^i + (e_k^i)^T e_k^i) \quad (16)$$

with λ_k^f the forgetting factor introduced to track mode-switches, and $\Sigma_0^i = 0$. The effective number of samples taken into account at an instant k is referred to as the memory length χ_k , which is given by

$$\chi_k = \lambda_k^f (\chi_{k-1} + 1) \quad (17)$$

with $\chi_0 = 0$. The expectation of the distance between process P and model M_i at time instant k is given by

$$E\{d(P, M_i)\} = E\{\Sigma_k^i\} = \begin{cases} \chi_k \sigma_\psi^2, & \text{if process is in mode } \Omega_i \\ \chi_k (\sigma_\psi^2 + \sigma_i^2), & \text{otherwise.} \end{cases} \quad (18)$$

Equation (18) shows that the smallest distance $d(P, M_i)$ will converge to $\chi_k \sigma_\psi^2$. Hence, the heuristic idea of Fortescue (1981) can be applied to the mode-recognition problem by keeping the sum Σ_k^i constant for the model with minimal sum Σ_k^i . In the case of weighting both old and new information, the sum Σ_k^i can be kept constant at target Σ_∞ by selecting the forgetting factor

$$\lambda_k^i = \frac{\Sigma_\infty}{\Sigma_{k-1}^i + (\mathbf{e}_k^i)^T \mathbf{e}_k^i}. \quad (19)$$

Åström and Wittenmark (1989) motivate that it is required that

$$0 < \lambda_k^i \leq 1. \quad (20)$$

Because $(\mathbf{e}_k^i)^T \mathbf{e}_k^i \geq 0$, it follows from (19) that this condition is fulfilled by

$$\Sigma_k^i \geq \Sigma_\infty (1 \leq i \leq n). \quad (21)$$

(21) is fulfilled by taking $\Sigma_0^i = 0$ ($1 \leq i \leq n$) and by taking $\lambda_k^i = 1$ until $\Sigma_k^i > \Sigma_\infty$ ($1 \leq i \leq n$).

By the use of the forgetting factor (19), the sum Σ_k^i is kept constant. In contrast, the sum Σ_k^j of all other models is variable and may become smaller or larger than the sum Σ_k^i . If a switch from mode Ω_i to mode Ω_j occurs, then the error \mathbf{e}_k^i will increase, and simultaneously the error \mathbf{e}_k^j will decrease. Equation (19) shows that an increase in error \mathbf{e}_k^i results in a smaller forgetting factor and hence in a smaller memory length. Due to the drop in memory length and the fact that \mathbf{e}_k^j is smaller than \mathbf{e}_k^i , the sum Σ_k^j will rapidly decrease and become lower than the sum Σ_k^i . A mode switch is detected at the time instant k at which $\Sigma_k^j < \Sigma_k^i$. At that time instant k , model M_j should be selected as the best model, and accordingly the sum Σ_k^j of squared errors should be kept constant. This is performed by taking the new forgetting factor λ_k^j equal to

$$\lambda_k^j = \frac{\Sigma_\infty}{\Sigma_{k-1}^j + (\mathbf{e}_k^j)^T \mathbf{e}_k^j}. \quad (22)$$

This mechanism guarantees an effective balance between mode-tracking and noise-insensitivity.

4.2.2. Generalized mode recognition. In practice, most processes are generalized mode-switch processes. This implies that when the process is not in the mode centre, the error between model output and process output is larger than the error given in (12). This would lead to a small effective memory length and hence to an enlarged disturbance sensitivity. One ad-hoc solution would be to enlarge the target sum of the *a posteriori* prediction error by increasing the asymptotic memory length, but this goes

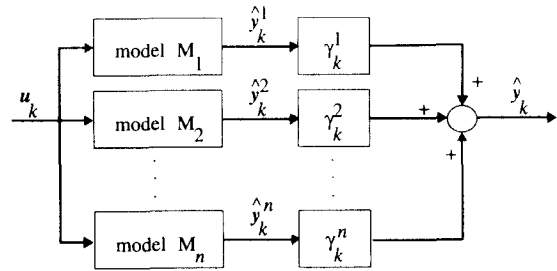


FIG. 2. Linear combination of weighted model outputs.

at the expense of reduced mode-tracking capability.

Another solution is as follows. As the process moves from one mode centre to another one, it can be assumed that the dynamics will change gradually and may be approximated by a linear combination of models. This implies that the process output is approximated by a weighted sum of the model outputs. Therefore, the error between process output and the weighted sum of model outputs will be close to the error caused by the measurement disturbance. Based on these weights, a model criterion can be designed such that the model which describes the encountered process dynamics best can be determined. In the following this approach will be discussed in more detail.

The predicted model output $\hat{\mathbf{y}}$, being the sum of weighted model outputs $\hat{\mathbf{y}}^i$, is given by

$$\hat{\mathbf{y}}_k = \gamma_k^1 \hat{\mathbf{y}}_k^1 + \gamma_k^2 \hat{\mathbf{y}}_k^2 + \dots + \gamma_k^n \hat{\mathbf{y}}_k^n, \quad (23)$$

where $\gamma_k^i \in \mathbb{R}$ are called the mode weights. In Fig. 2, a block diagram is shown of the sum of weighted model outputs.

If the process is in mode centre ω_i at time instant k , the mode weights λ_k^j , $j \in [1, \dots, n]$, are such that

$$\lambda_k^j = \begin{cases} 1, & \text{for } j = i \\ 0, & \text{for } j \neq i. \end{cases} \quad (24)$$

A mode weight vector γ_k which satisfies (24) for $\gamma_k^i = 1$ is referred to as the *i*th unity vector ξ_i . The set of unity vectors is referred to as the unity vector set Ξ . For instance, if $n = 2$, then the mode-vector set Ξ is given by $\Xi = \{\xi_1, \xi_2\}$ with $\xi_1 = (1, 0)$ and $\xi_2 = (0, 1)$. The representation (23) can also be described in vector form:

$$\hat{\mathbf{y}}_k = \Phi_k^T \gamma_k, \quad (25)$$

where $\gamma_k = [\gamma_k^1, \gamma_k^2, \dots, \gamma_k^n]^T$ is an n -dimensional mode weight vector, and $\Phi_k^T = [\hat{\mathbf{y}}_k^1, \hat{\mathbf{y}}_k^2, \dots, \hat{\mathbf{y}}_k^n]$ the transpose of an $n * m$ observation matrix with m the number of elements in \mathbf{y}_k . The error \mathbf{e}_k between process output \mathbf{y}_k and the weighted

sum of model outputs \hat{y}_k is given by

$$\mathbf{e}_k = \mathbf{y}_k - \hat{\mathbf{y}}_k. \quad (26)$$

Equations (25) and (26) show that the vector $\hat{\mathbf{y}}_k$ is linear in the parameter vector $\boldsymbol{\gamma}$. Consequently, the mode vector $\boldsymbol{\gamma}$ can be easily found with least-squares estimation. Furthermore, as the error \mathbf{e}_k will be close to the error caused by the measurement disturbance, an adaptive forgetting scheme such as in Fortescue *et al.* (1981) can be applied in order to track the time-varying mode vector $\boldsymbol{\gamma}$.

From the different forgetting schemes described in the literature (Kulhavy, 1987; Tulleken, 1987), uniform forgetting of both old and new measurements as described by Löhnberg *et al.* (1990) has been adopted. This scheme is superior in tracking time-varying parameters and is most simple. In order to prevent the occurrence of covariance blow-up, a well-known phenomenon of uniform forgetting, a measure is taken which will be described later on in more detail.

For weighted recursive least-squares estimation with uniform forgetting of both old and new data, the sum of weighted summed least-squared errors is given by

$$\Sigma_k = \lambda_k^g (\Sigma_{k-1} + \mathbf{e}_k^T [\mathbf{I} + \mathbf{D}_k]^{-1} \mathbf{e}_k) \quad (27)$$

with $\lambda_k^g \in (0, 1]$ the forgetting factor, $\Sigma_0 = 0$, and where the condition matrix \mathbf{D}_k and recursion matrix \mathbf{P}_k are

$$\mathbf{D}_k = \Phi_k^T \mathbf{P}_{k-1} \Phi_k \quad (28)$$

$$\mathbf{P}_k = [\lambda_k^g \mathbf{P}_{k-1}^{-1} + \lambda_k^g \Phi_k \Phi_k^T]^{-1}. \quad (29)$$

The recursive least-squares mode vector estimate $\hat{\boldsymbol{\gamma}}_k$ minimizing (27) is given by

$$\begin{aligned} \hat{\boldsymbol{\gamma}}_k &= \arg \min_{\boldsymbol{\gamma}} \Sigma_k \\ &= \arg \min_{\boldsymbol{\gamma}} \lambda_k^g (\Sigma_{k-1} + \mathbf{e}_k^T [\mathbf{I} + \mathbf{D}_k]^{-1} \mathbf{e}_k). \end{aligned} \quad (30)$$

Applying the same original ideas of Fortescue (1981) as in Section 4.2.1, the sum of squares Σ_k can be kept at a constant target sum Σ_∞ by use of the forgetting factor

$$\lambda_k^g = \frac{\Sigma_\infty}{\Sigma_\infty + \mathbf{e}_k^T [\mathbf{I} + \mathbf{D}_k]^{-1} \mathbf{e}_k}, \quad (31)$$

which is an obvious generalization of the results obtained by Löhnberg *et al.* (1990). Comparison of the forgetting factor (31) obtained for generalized mode-switch processes to the forgetting factor (19) shows that the structure is similar, but that the denominator terms are different. Yet the first term in the denominator of (19) Σ_{k-1}^i equals that in (20) as long as no mode-switch occurs. The forgetting factor (31)

shows that at each time instant k a correction is made for estimation errors. For instance, when the condition matrix $\mathbf{D}_k = 0$ (i.e. the prediction is zero), then the denominator terms for both forgetting factors are equal.

The probability of the occurrence of covariance blowup can be decreased by requiring that the actual memory length should not drop below a certain minimum memory length χ_{\min} . This can be accomplished by constraining the forgetting factor λ_k^g to the range $(\lambda_{\min}, 1]$, where $\lambda_{\min} = \chi_{\min} / (\chi_{k-1} + 1)$.

The parameter vector estimate $\hat{\boldsymbol{\gamma}}_k$ and the covariance matrix estimate $\hat{\mathbf{P}}_k$ determined at time instant k can be used for determining the model which describes the current process dynamics best. This can be performed by determining the likelihood of all unity vectors, and by selecting as the best model that model M_i for which the likelihood of unity vector ξ_i is maximal. If the measurement disturbance is Gaussian and has variance σ_ψ^2 and if the mode vector $\boldsymbol{\gamma}$ has mean $\hat{\boldsymbol{\gamma}}_k$, then the likelihood p_k of a unity vector ξ_i at time instant k is given by Goodwin and Payne (1977)

$$p_k(\xi_i) = \frac{e^{-1/2(\hat{\boldsymbol{\gamma}}_k - \xi_i)^T (\mathbf{P}_k \sigma_\psi^2)^{-1} (\hat{\boldsymbol{\gamma}}_k - \xi_i)}}{c_k}, \quad (32)$$

where c_k is a constant depending on \mathbf{P}_k , on σ_ψ^2 and on the number of modes n . As only the likelihood of the n unity vectors is considered, it is convenient to write $p_k(\xi_i)$ as p_k^i and to normalize the likelihoods by taking c_k such that $\sum_{i=1}^n p_k^i = 1$. Most importantly, the model M_i with maximal likelihood p_k^i is selected by the model detector.

4.3. Model output

The mode-recognition methods discussed above require the computation of $\hat{\mathbf{y}}_k^i$ for each model M_i at each time instant k . One method to compute these predictions is with a Kalman filter (Willsky, 1976). This approach requires that for each model M_i the disturbance characteristics of both measurement and system disturbances are Gaussian, and that the variances are known. Then the stationary Kalman gain is determined such that the state predictions are a compromise between the effects of the state and measurement errors. Lund *et al.* (1991) argue that the Kalman gain has a large influence on the distinction between the predictions $\hat{\mathbf{y}}_k^i (1 \leq i \leq n)$. If the Kalman gain would be zero, the model states may drift away from the process state. This will result in bad predictions $\hat{\mathbf{y}}_k^i$ and in large differences between the various predictions $\hat{\mathbf{y}}_k^i$. On the other hand, when the Kalman gain is large for all models, the model predictions $\hat{\mathbf{y}}_k^i$

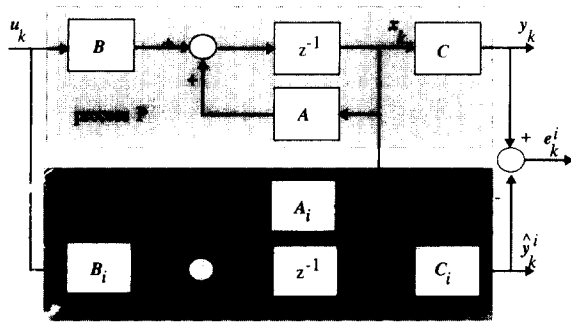


FIG. 3. Parallel structure with regular resetting.

will be influenced by the feedback. This will result in predictions which will be close to each other. Hence, the distinction between the predictions would be difficult to make. Furthermore, in practice the variances of the system disturbance are generally not known.

Another choice for computation of \hat{y}_k^i is to make use of a parallel structure with regular resetting. In Fig. 3, a parallel structure with regular resetting is shown for a discrete-time state-space process and model. The matrices \mathbf{A} , \mathbf{B} and \mathbf{C} characterize a discrete-time state-space description of the process, and the matrices \mathbf{A}_i , \mathbf{B}_i and \mathbf{C}_i characterize a discrete-time state-space description of the model M_i . Furthermore, T_o denotes the observation time, which is given by $n_o T_s$, with number of samples $n_o \in \mathbb{IN}$, $n_o \geq 1$, and T_s the sampling time. At the beginning of each observation period, that is, after every n_o samples, the model states of all models are made equal to the process state \mathbf{x} .

The advantage of this structure is that drift in the model states can be avoided by regular resetting of the model states to the process state. This is performed by selecting an appropriate observation period T_o . If T_o is selected equal to one sampling period, the structure is just the well-known one-step ahead predictor. Van Amerongen (1982) shows that it may be advantageous to select T_o larger than one sampling period, as the predictions become less sensitive to noise. Practical experiments show that taking the observation period T_o faster than the dominant time constant of the closed loop (e.g. half) results in a good distinction between models.

If not all process states are measurable, then a discrete-time input-output model of the process can be used. Then the states can be reconstructed from the discrete-time output measurements.

4.4. Mode-switch and bumpless transfer

Once the mode detector has detected a mode switch, a new controller has to replace the

current one. At the instant the parameters in the controller are adapted, an unwanted "bump" in the control signal may be introduced. Bumps in the control signal should be avoided. One way of avoiding bumps in the control signal is to make use of interpolation between controllers as studied by Athans *et al.* (1977). However, this results in a continuous adjustment of the control signal, and hence in an increased noise-sensitivity. Furthermore, continuous adjustment of the controller is not needed and wanted for mode-switch processes. Another way of avoiding bumps is to make a fast and smooth transition, such that after transition no continuous adjustment is made. This can be performed by applying a bumpless transfer algorithm. The applied method of bumpless transfer depends on the type of the controller. Here only static state-feedback controllers will be considered. A static state-feedback controller C_i can be described as

$$\mathbf{u} = \bar{\mathbf{u}}_i - \mathbf{K}_i \mathbf{e}, \quad (33)$$

where $\bar{\mathbf{u}}_i$ is the nominal control input, \mathbf{K}_i is the static gain, and \mathbf{e} is the error vector between desired state and actual state. In order to create bumpless transfer (Åström and Wittenmark, 1990) for a static state-feedback controller, a number of internal states \mathbf{x}_b can be added to the controller, that is

$$\dot{\mathbf{x}}_b = -\frac{\mathbf{I}}{\tau_b} \mathbf{x}_b, \quad (34)$$

$$\mathbf{u} = \bar{\mathbf{u}}_i - \mathbf{K}_i \mathbf{e} + \mathbf{x}_b, \quad (35)$$

where \mathbf{I} is a unity matrix of appropriate dimensions, and τ_b is the bumpless transfer time constant. The value of τ_b determines how fast the transient between the old control signal and the new control signal will be. Practical experiments (Hillhorst, 1992) show that taking the value of τ_b approximately equal to the dominant time constant of the closed loop results in a fast and smooth transition. The number of internal states should be made equal to the dimension of the control signal \mathbf{u} . In order to obtain bumpless transfer at time instant t_m when the controller switches from C_i to C_j , the internal state \mathbf{x}_b is reset to

$$\mathbf{x}_b(t_m) = \mathbf{u}_i(t_m) - \mathbf{u}_j(t_m), \quad (36)$$

where $\mathbf{u}_i(t_m)$ is the current control signal based on controller C_i , and $\mathbf{u}_j(t_m)$ is the control signal based on the parameters of the new controller C_j if no bumpless transfer was applied.

5. EXPERIMENTAL RESULTS

In order to study the supervisor for control of mode-switch processes in practice, it has been

applied to both a finite mode-switch process and a generalized mode-switch process. A practical set-up of a one-link flexible beam represents a finite mode-switch process. An experimental pending two-link flexible robot arm represents a generalized mode-switch process. Both are present in the Control Laboratory in Twente (Kruise, 1990). The experimental results will be discussed in Sections 5.1 and 5.2.

5.1. One-link flexible beam

The experimental set-up of the one-link flexible beam is given in Fig. 4. It consists of a dc-motor which drives a 1.9 m long flexible aluminium beam which can move in the horizontal plane. Therefore, there is no influence of gravity. The dimensions of the beam are such that only transversal vibrations in the horizontal plane have to be taken into account. The control algorithm is based on feedback of a state-estimate obtained from the angle measurement of the motor axis, as well as on feedback of the state variables of two of the vibrational modes, which are estimated by an observer from bending measurements by strain gauges (Kruise, 1990).

Kruise (1990) demonstrated that in order to have a proper control for various payloads m_p , a robust controller can be designed. In the experiments three values of the payload were considered: 0, 0.25 and 0.5 kg. A disadvantage of such a robust controller is that in most cases its performance is not optimal. It gives almost no overshoot in all three cases, but for smaller payloads the responses are unnecessarily sluggish. Therefore, this set-up was a good candidate for testing the mode-switch concept. Because the (linear) dynamics of the beam only change with the payload mass, and as the number of payloads is limited, this process is a finite mode-switch process. In addition, it is a typical example of a

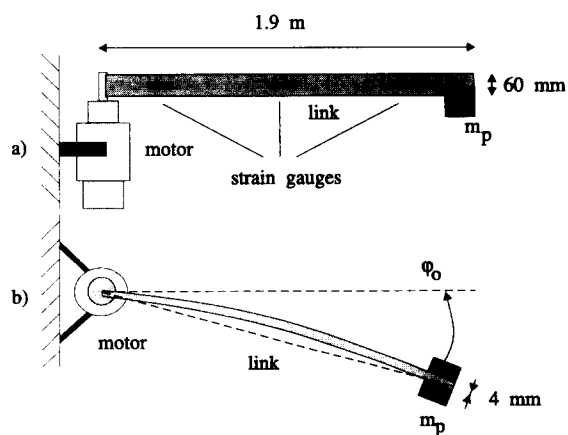


FIG. 4. Experimental set-up of a one-link flexible beam: (a) side view; (b) top view.

process where a very fast adaptation is required because otherwise the task is completed before the parameters are properly adjusted.

The experimental conditions were as follows. The finite mode recognition of Section 4.2.1 was used. Earlier identified models for the three different payloads were present in the memory, as were the corresponding state-feedback controllers designed for negligibly small overshoot and a fast response. Step changes of the reference signal for the tip angle were made from -90 to $+90^\circ$. The purpose of the experiments was to test whether the appropriate models were selected by the mode detection algorithm, and whether improved control performance was obtained compared to the robust controller. The controller for a payload of 0.5 kg was used as the robust controller. Figure 5 gives the results.

In Fig. 5 u is the control signal, φ_o is the tip angle and M_i is the selected model. The solid lines give the response for the mode-switch controller, while the dotted lines give the responses for the robust controller. The experimental results with payload mass $m_p = 0, 0.25$ and 0.5 kg are shown in Fig. 5(a)–(c), respectively. It can be seen that for the mode-switch controller, all responses show almost no overshoot. For the smaller payloads the response time is smaller with the mode-switch controllers than with the robust controller. The appropriate model was not always selected. Further analysis revealed that at the start of the response the models in the memory did not properly describe the non-minimum phase behaviour of the processes. However, this had no influence on the performance because the control signal is saturated in any case. The wrong model selection when the tip has reached its final position can be explained by the presence of Coulomb friction, which was not modelled. It results in a smaller acceleration than predicted by the models. The best resemblance is found for model M_3 , which corresponds to the largest payload mass and thus to the smallest acceleration.

5.2. Two-link flexible robot arm

A second series of experiments was carried out with a generalized mode-switch process, represented by a pending flexible robot arm consisting of two links (see Fig. 6).

Because the robot arm is pending from the ceiling and moves in the vertical plane, gravity has a large influence. Due to the flexibility, the model of this robot arm is rather complex (eighth-order), and due to the influence of gravity it is not possible to achieve a good

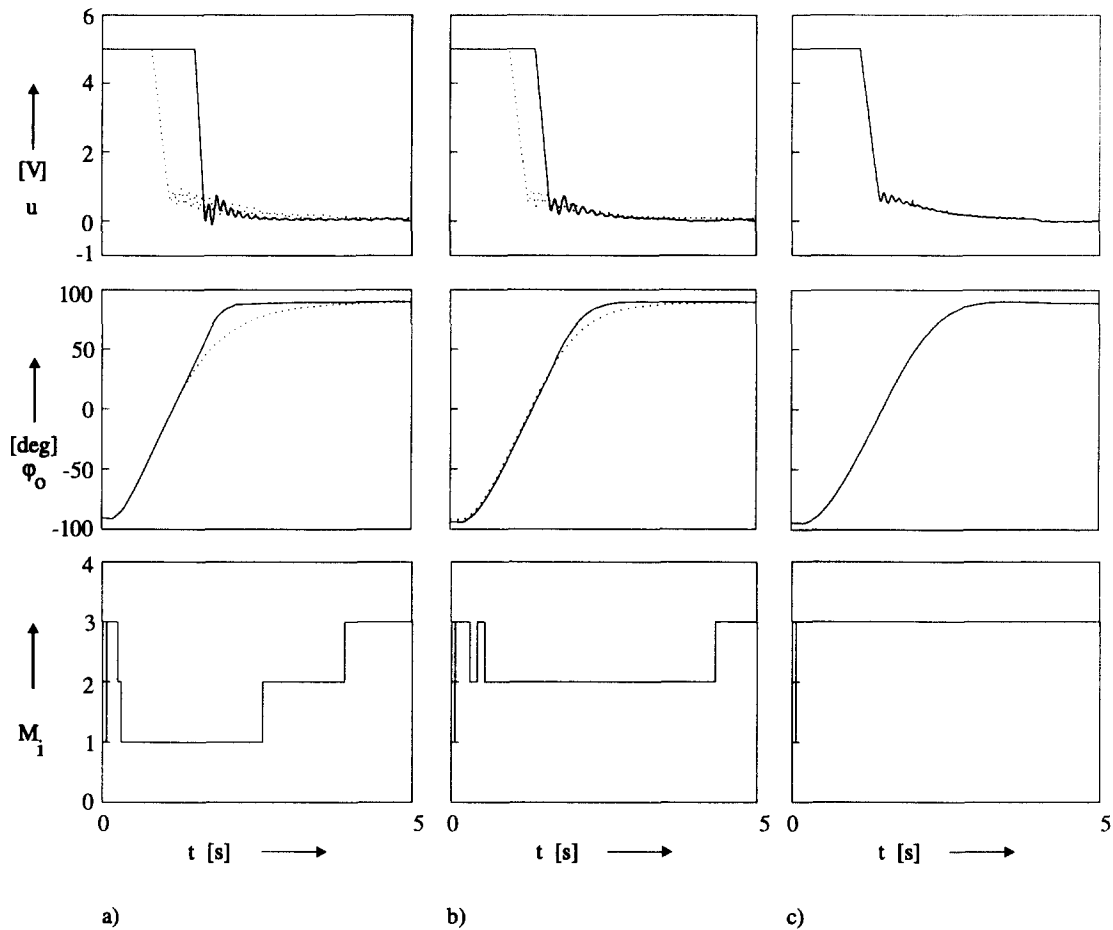


FIG. 5. Experimental results with a one-link flexible beam for payload mass (a) Ω_1 : $m_p = 0$ kg, (b) Ω_2 : $m_p = 0.25$ kg and (c) Ω_3 : $m_p = 0.5$ kg. u = control signal of the supervisor for adaptive control (solid line) and robust controller (dotted line), φ_0 = tip angle using the supervisor for adaptive control (solid line) and using robust controller (dotted line), and M_i = model selected.

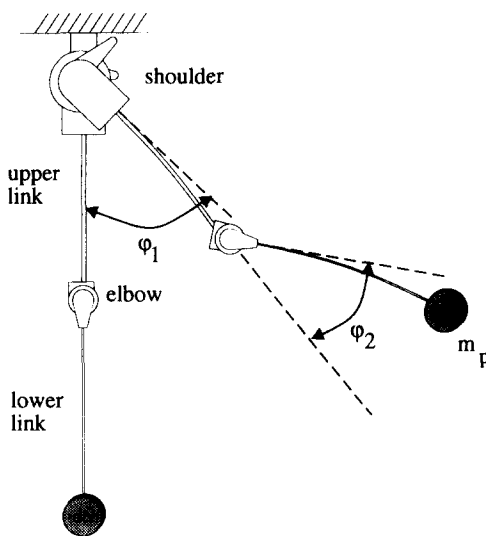


FIG. 6. Experimental set-up of a two-link flexible robot arm. φ_1 = angle of the upper link, φ_2 = angle of the lower link relative to the upper link and m_p = mass of the payload.

control performance with a single (linear) controller. Therefore, for different operating points, models were identified and linear

state-feedback controllers were developed. Also, for this set-up use was made of pole-placement for negligible steady-state error in each operation point, and negligible overshoot and fast settling when the robot moves from one operation point to another. The generalized mode recognition algorithm of Section 4.2.2 was used.

The following experiments were carried out. The reference angle for the upper arm was kept at $\varphi_1^r = 0^\circ$ while the reference angle φ_2^r of the lower arm was changed from 0 to 90° and then back to 60, 30 and 0° . A payload of 0.5 kg was transported. In the first experiment only a controller based on a model identified for $\varphi_2^r = 0^\circ$ was used. The results of this experiment are given in Fig. 7.

It can be seen that except for $\varphi_2^r = 0^\circ$, there are large steady-state errors and that the responses have quite some overshoot. In the next experiment there were four controllers and models in the memory. The results of this experiment are given in Fig. 8.

It can be seen that the steady-state errors

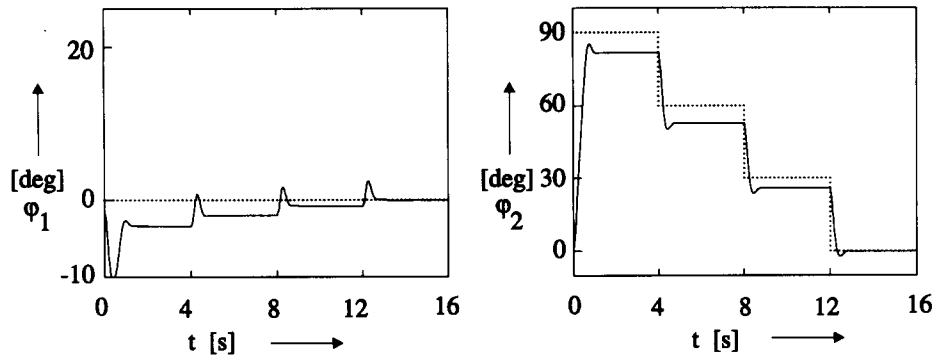


FIG. 7. Only model M_1 in memory. φ_1, φ_2 = angles of link 1 and 2 (solid line). φ_1^r, φ_2^r = reference angles of link 1 and 2 (dotted lines).

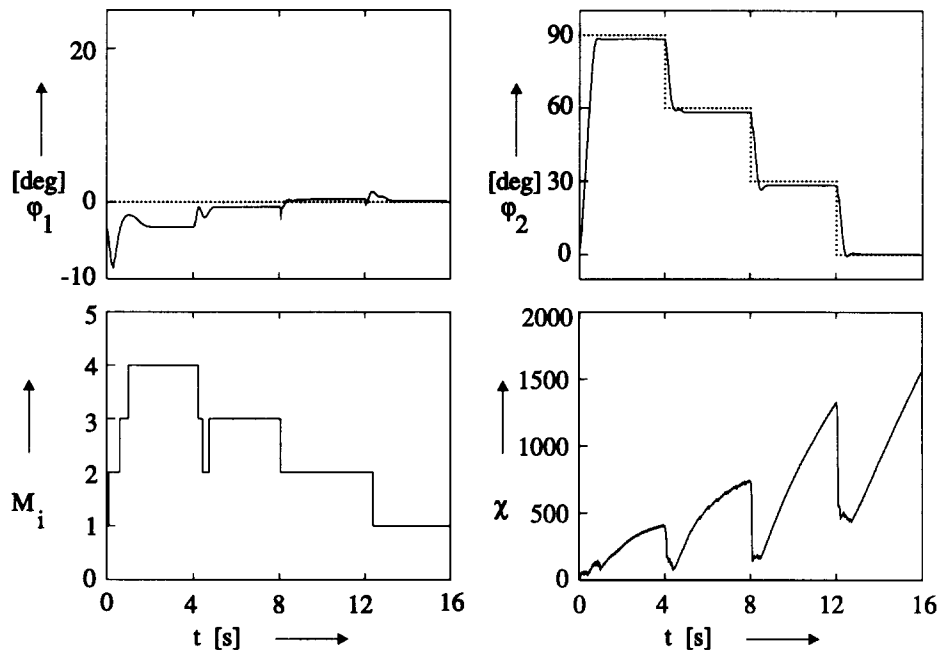


FIG. 8. Four models in memory. Symbols as in Fig. 7 with M_i the model selected and recorded memory length χ .

are much smaller, although not equal to zero. The reason is that the models do not properly describe the static gains. It would be a task for the performance monitor to detect this. In most situations the overshoot is smaller than when only one single controller is used. When the lower arm moves from $\varphi_2^r = 0^\circ$ to $\varphi_2^r = 90^\circ$, the different models M_i are appropriately selected. The only wrong model selection is made at $t = 4.2$ s.

Figure 8 also shows that immediately after the time instants of a mode switch, that is, at $t = 4, 8,$ and 12 , the memory length χ drops dramatically. This implies that at the time instants of a mode switch the old information is forgotten rapidly in favour of new incoming information. After the process reaches the new mode centre (i.e. $t = 1, 5, 9$ and 13), the memory length increases almost linearly with

time. This demonstrates that the algorithm does what one would intuitively expect.

The figure also shows that both the minimal memory length χ reached after the time instants of a mode switch and the maximal memory length χ reached after the drops increase with time. Because simulations for several movement patterns for exact models showed that the memory length was independent of the angle (Hillhorst, 1992), the increase in minimal and maximum memory length must be due to model mismatch. This is also reflected by larger steady-state control errors. When this model mismatch is large ($\varphi_2^r = 90^\circ$), a small memory length χ is needed for having Σ_k constant at Σ_∞ . Alternatively, when the model mismatch is small ($\varphi_2^r = 0$), a large memory length χ is needed.

It can be seen from the recorded model M_i and the memory length χ_k in Fig. 8 that the

wrong selection of model M_2 at time instant 4.2 s occurs at the instant of a minimal memory length χ_k . Hence, at that instant the method is most sensitive to noise.

6. CONCLUSIONS AND SUGGESTIONS

The concept of a supervisor for control of mode-switch processes was worked out into a new method which can be seen as an attractive alternative for both robust control and conventional adaptive control for mode-switch processes. With the introduction of the mode concept, a controller is obtained which behaves in a less conservative manner than a robust controller, having the ability to adjust the control parameters quickly by making more effective use of process knowledge than a conventional adaptive controller does. The main problem in this approach is how to detect a mode switch. The attractive properties of a newly developed method of exponential forgetting, dedicated for the use on mode-switch processes, have been demonstrated by real experiments. It has been demonstrated that a supervisor for control of mode-switch processes results in a better overall performance than what is achieved with a fixed linear controller. For example, for the two-link robot arm, the steady-state error and the amount of overshoot were reduced. This improvement could be obtained in the presence of nonlinearities, of a large number of fastly changing parameters, and of model inaccuracies and disturbances. The results show that fast adaptation of control parameters can be obtained without disturbing the process. Therefore, the supervisory structure should be used as standard for the control of mode-switch processes.

Because of the complexity of the controlled system consisting of a mode-switch process together with a supervisor and several controllers, no analysis of stability, convergence and robustness [like that performed by Shamma and Athans (1990, 1991) for gain-scheduling control of nonlinear plants and linear parameter-varying plants] has been carried out yet. Nevertheless, the adequate behaviour of not only simulations, but also of several real-world controlled systems indicates that this approach works satisfactorily.

It is rather obvious that the mode switch concept will not introduce stability problems as long as the switching occurs relatively slowly with respect to the process dynamics. However, the real world examples demonstrate that also when there are rapid switches the system still behaves well and better than with a fixed controller. On the other hand, the need for a

bumpless transfer algorithm is also due to the fact that very rapid switches lead to an "unstable" system. Determination of the exact stability condition is rather complex and is a subject for further research.

In this article only linear controllers were used, because for most processes only a linear model in a mode centre is known. For nonlinear time-variant processes such as robots, often a nonlinear model is available for each payload mass. A fruitful extension would be the switching between nonlinear controllers for such processes. Then mode-recognition methods described in this article are expected to be usable for this case also. Then instead of linear models running in parallel with the process, nonlinear models should be used.

REFERENCES

- van Amerongen, J. (1982). Adaptive steering of ships—a model reference approach to improved manoeuvring and economical course keeping. Ph.D. thesis, Delft University of Technology, Delft, The Netherlands.
- Årzén, K. E. (1987). Realization of expert system based feedback control. Ph.D. thesis, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.
- Årzén, K. E. (1989). An architecture for expert system based feedback control. *Automatica*, **25**, 813–827.
- Åström, K. J. and B. Wittenmark (1989). *Adaptive Control*. Addison Wesley, NY.
- Åström, K. J. and B. Wittenmark (1990). *Computer Controlled Systems*. Prentice-Hall, Englewood Cliffs, NJ.
- Athans, M., D. Castañón, K. P. Dunn, C. S. Greene, W. H. Lee, N. R. Sandell Jr and A. S. Willsky (1977). The stochastic control of the F-8C aircraft using a multiple model adaptive control (MMAC) method—part I: equilibrium flight. *IEEE Trans. Automatic Control*, **AC-22**, 768–780.
- Fortescue, T. R., L. Kershenbaum and B. E. Ydstie (1981). Implementation of self-tuning regulators with variable forgetting factors. *Automatica*, **17**, 831–835.
- Goodwin, G. C. and R. L. Payne (1977). *Dynamic System Identification, Experimental Design and Data Analysis*. Academic Press, NY.
- Hilhorst, R. A. (1992). Supervisory control of model-switch processes. Ph.D. thesis, University of Twente, Enschede, The Netherlands.
- Hilhorst, R. A., J. van Amerongen, P. Löhnberg and H. J. A. F. Tulleken (1991). Intelligent adaptive control of mode-switch processes. In *Proc. IFAC Conference ITAC'91*, Vol. 2 Pergamon, Oxford.
- INMOS Limited (1988). *Transputer Reference Manual*. Prentice-Hall, Hemel Hempstead, London.
- Isermann, R. (1984). Process fault detection based on modeling and estimation methods—a survey. *Automatica*, **20**, 387–404.
- Isermann, R. and K. H. Lachmann (1985). Parameter-adaptive control with configuration aids and supervision functions. *Automatica*, **21**, 625–638.
- Jansma, H. J. and H. J. A. F. Tulleken (1986). An industrial approach towards the development of a multi-variable self-tuner. *Journal A*, **27**, 175–183.
- Kruise, L. (1990). Modeling and control of a flexible beam and robot arm. Ph.D. thesis, University of Twente, Enschede, The Netherlands.
- Kulhavy, R. (1987). Restricted exponential forgetting in real-time identification. *Automatica*, **23**, 589–600.
- Löhnberg, P., A. Stienstra and H. J. A. F. Tulleken (1990). Time-varying parameter estimation combining directional and uniform forgetting. In *Proc. 11th IFAC World Congress*, Vol. 7, pp. 232–237, Tallinn, Estonia.

- Lund, E. J., J. G. Balchen and B. A. Foss (1991). Multiple model estimation with inter-residual distance feedback. In *Proc. 9th IFAC/IFORS Symp. on Identification and System Parameter Estimation*, 8–12 July, Budapest.
- Moose, R. L., H. F. Van Landingham and P. E. Zwicke (1978). Digital set point control of nonlinear stochastic systems. *IEEE Trans. Industrial Electronics Control Instruments*, **IECI-25**, 39–45.
- Reid, R. E. and V. E. Williams (1978). A new ship design criterion for improving heavy weather steering. In *Proc. 5th Ship Control Systems Symp.*, Vol. 1, pp. C1-1–C1-65, David W. Taylor Naval Ship Research and Development Center, Bethesda, MD 20084 (Attn: Code 273).
- Shamma, J. S. and M. Athans (1990). Analysis of gain scheduled control for nonlinear plants. *IEEE Trans. Automatic Control*, **35**, 898–907.
- Shamma, J. S. and M. Athans (1991). Guaranteed properties of gain scheduled control for linear parameter-varying plants. *Automatica*, **27**, 559–564.
- Skeppstedt, A., L. Ljung and M. Millnert (1992). Construction of composite models from observed data. *Int. J. Control.*, **55**, 141–152.
- Tulleken, H. J. A. F. (1987). An adaptive forgetting strategy for on-line identification of multivariable processes. *Journal A*, **28**, 195–208.
- Tulleken, H. J. A. F. (1992). Grey-box modelling and identification topics. Ph.D. thesis, Delft University of Technology, Delft, The Netherlands.
- Tzafestas, S. G. and K. Watanabe (1990). Modern approaches to system/sensor fault detection and diagnosis. *Journal A*, **31**, 42–57.
- Willsky, A. S. (1976). A survey of design methods for failure detection in dynamic systems. *Automatica*, **12**, 601–611.