Transient Management in Reconfigurable Control Systems*

Gyula Simon Tamás Kovácsházy Gábor Péceli
{simon, khazy, peceli}@mit.bme.hu

— February 22, 2002 —

Budapest University of Technology and Economics, Budapest, Hungary
Department of Measurement and Information Systems
http://www.mit.bme.hu/

* Funded, in part, by the DARPA ITO's Software-Enabled Control Program under AFRL contract F33615-99-C-3611.

1. Introduction

This report investigates the transient phenomena due to run-time reconfigurations in controlled systems. It is shown that in the case of dynamic systems these reconfiguration transients might be reduced to a considerable extent using dedicated design-time and/or run-time methods. The methodology behind applies the so-called model-integrated approach [SK97], which tries to utilize the available knowledge about the system and its environment by constructing “dynamically” consistent computer-based representations. These representations integrate logic, dynamics and constraints [BeMo99], and provide hierarchical system descriptions for monitoring and control applications. Such descriptions can handle evolution governed by on/off switches, if-then-else rules, system degradation due to faults, etc.

There are very many approaches that deal with such and similar systems, often called hybrid systems [GNRR93], however, very few studies address the problem of transient phenomena due to the interactions of the higher-level logical/discrete mechanisms and the lower-level continuous dynamics. Since these transients might cause intolerable disturbances both in the value and the time domain, the reduction of such side effects becomes an additional design issue.

Throughout this report the interactions causing transients are considered as reconfigurations “implemented” as instantaneous parameter and/or structural changes, and the introduced additional design considerations aim at reducing these transients. The actual course of the reconfiguration transients depends on one hand on the structure of implementation, while on the other on the actual interactions with the environment. For this very reason transient reduction can be achieved partly by selecting proper structure for implementation and partly by techniques utilizing multi-objective, run-time optimization. The term “transient management” tries to reflect the simple fact that to reduce the transients several constraints and control rules are to be applied simultaneously, and therefore a management-like activity is needed.

The first part of Section 2 shows how and why transient phenomena might appear after reconfiguration. The second part includes a literature survey covering studies related to different aspects of transient behavior, summarizes the already available results concerning “transient management”, and outlines the content and the structure of the report.

Section 3 is devoted to introduce the so-called passive transient management methods, which utilize the fact that the different structures of implementation produce quite different transients in the case of otherwise identical reconfigurations. These passive methods are design-time techniques, and do not influence the transient phenomena directly. This section also contains some considerations concerning multiple-step reconfigurations and provides examples illustrating the strength of the propositions both in open-loop and closed-loop control applications.

Section 4 presents the so-called active transient management methods, which either directly influence the controllable variables responsible for transients, or produce anti-transient signals/sequences to compensate the disturbing side effects of the transients. These active methods are techniques evaluated in run-time. The new methods are illustrated both in open-loop and closed-loop control applications.

Section 5 summarizes the most important open problems of transient management in reconfigurable control systems.

2. Reconfigurable systems

The problems considered in this report are related to such computer-controlled complex dynamic systems, which, depending on their mode of operation, can be characterized by different configurations, and where any mode change corresponds to a configuration change, i.e., reconfiguration. Typical examples are the so-called hybrid systems [GNRR93], which are piece-wise continuous dynamic systems governed both by continuous and discrete laws. The discrete control operates, e.g., on/off switches or valves, different selectors, etc. within the physical system resulting in another configuration and piece of continuous operation.

We usually distinguish between intentional and non-intentional changes, mainly because different transient management methods can be applied if the time and the nature of the reconfiguration are predictable, or if both are unpredictable. A dedicated special case is the occurrence of faults, which can be considered from modeling point of view as unpredictable reconfiguration.

In the first part of this section the transient phenomena in reconfigurable systems is described. The second part incorporates a short literature survey including the preliminary results concerning transient reduction. Finally the section concludes with a possible classification of transient management techniques.

2.1 Transient phenomena in reconfigurable systems

The transients investigated in this report arise at switching from one operational mode to another. Switching is considered as instantaneous event, and results in sudden changes of conditions. Transients appear usually as damped oscillatory motions, which persist for a relatively short time after the change has occurred. These motions, however, can get an explicit interpretation only if the different operational modes and domains can be characterized separately. Typically such integral descriptors, similarly to the parameters, take a new value after switching. In a dynamic system such an integral descriptor is the state, which takes after reconfiguration, in the majority of the cases, a different value. In systems where the concept of steady-state is applicable, transients are defined as transitions between two steady-states. The separation of the transients from the stationary values of the states is in general a complicated task. Fortunately for the linear, time-invariant, lumped parameter systems this separation can be solved explicitly: the reconfiguration transient is described as the solution of the homogeneous part of the corresponding differential equation using the initial conditions valid at the time instant of the reconfiguration.

This subsection serves as an introduction to the problem statement: what is the real reason of transients, and what kinds of ideas are available to reduce or avoid these phenomena?

An introductory example: Let's consider a rather simple electrical system, which can serve also as a simplified model of several other physical problems. Here we suppose, that the concept of steady-state is applicable.
Let's consider a charge-free RC divider, (or equivalently a tube and an empty container modeled as an RC divider) (see Fig. 2.1.a) having an input after t=0 consisting of a constant (dc) component and sinusoidal "perturbation":

$$U_1(t) = U_{10} + U_{11} \sin \omega_0 t \tag{2.1}$$

The output of the divider (which is proportional with the content of the capacitor or equivalently with that of the container) for $t > 0$:

$$U_2(t) = U_{10}\left[1 - e^{-\frac{t}{T}}\right] + U_{11} \cos\varphi \left[\sin(\omega_0 t - \varphi) + e^{-\frac{t}{T}} \sin\varphi\right], \tag{2.2}$$

where $T = RC$; $\varphi = arctg\omega_0 T$. The transient signal components can be identified easily, since they can be characterized by the exponential decay with time. It is important to note that both the dc and the sinusoidal component of the input will cause a transient effect. For $t >> T$ the transient response will vanish, and the steady-state value of the output will be reached:

$$U_2(t) = U_{20} + U_{21}\sin(\omega_0 t - \varphi) = U_{10} + U_{11}\cos\varphi\sin(\omega_0 t - \varphi), \tag{2.3}$$

where $U_{20} = U_{10}$; $U_{21} = U_{11}\cos\varphi$. The result is as it is expected: the output voltage of capacitor C will vary around its average, which equals the dc component of the input.

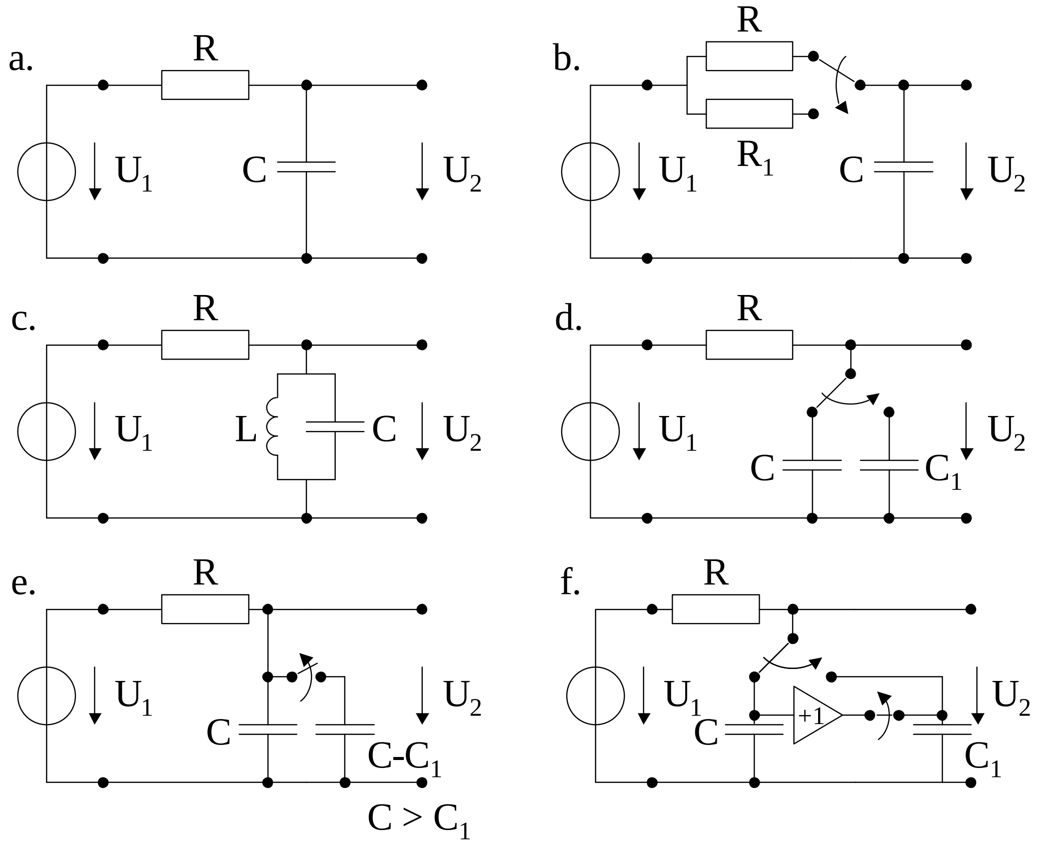


Fig.2.1: RC divider and its reconfiguration

If at $t = t_0$ (where $t_0 >> T$) this simple system is "reconfigured" by changing the value of R ($R \rightarrow R_1$: see Fig. 2.1.b), and consequently the charging current of the capacitor is also changed, then at the output the next response is obtained

$$U_2(t) = U_{10} + U_{11}\cos\varphi_1\sin(\omega_0 t - \varphi_1) + U_{11}\left[\cos\varphi\sin(\omega_0 t_0 - \varphi) - \cos\varphi_1\sin(\omega_0 t_0 - \varphi_1)\right]e^{-\frac{t - t_0}{T_1}} \tag{2.4}$$

where $t \geq t_0$, $T_1 = R_1 C$; $\varphi_1 = arctg\omega_0 T_1$. Here the first two terms give the stationary (steady-state) solution, while the transient component is provided by the third term.

Remarks:

1. The dc component of the input voltage can not cause a transient signal component at reconfiguration (see (2.4)), because the dc component of resistor's current already became zero at $t = t_0$.

2. The transient is proportional to the magnitude difference due to the change in the phase-shift ($\varphi \to \varphi_1$) of the ac component. If this magnitude difference is zero (see the last term of (2.4)) then the reconfiguration will not cause a transient component.
3. Forcing the last term of (2.4) to be zero, we can calculate the time instants (t_0 values), where at reconfiguration no transient component will appear. After some simple manipulations:

$$t_0 = \frac{1}{\omega_0} arctg \frac{\omega_0^2 T T_1 - 1}{\omega_0 (T + T_1)} . \tag{2.5}$$

   This result shows that there are possible techniques to avoid or reduce intolerable transient effects.
4. It is important to note, that the functions described by (2.2) and (2.4) are continuous at t=0 and $t = t_0$, respectively. This is simply due to the fact, that the capacitor in unable to change its voltage abruptly.
5. If we consider the time function of the output voltage of the different configurations under steady-state condition, we can realize, that if the output voltages differ at $t = t_0$, transients will unavoidably appear. We can get rid of these transient phenomena only if this output voltage difference is zero. It is also true, that if we can keep this output voltage difference at a low level, then the size of the transient effects will also be smaller.
6. All the above developments and conclusions can be generalized for input signals with more sinusoidal components.
7. Using frequency transposition techniques the transfer properties of the above simple circuit around zero frequency can be moved to an arbitrary frequency or frequencies (for the simplest case see Fig. 2.1.c). For such transposition the transient-free behavior of the dc component described above will appear at the center frequency of the band-pass filter created by frequency transposition, if the reconfiguration does not change the center frequency.
8. The idea to have the very same behavior at different frequencies is present also within a dedicated signal processing structure, called resonator-based structure [PMP96], which can be reconfigured transient-free if the frequency of the input signal components coincides with that of the resonators, and the resonator frequencies are not changed.

The "reconfiguration" in this simple example can be performed also in such a way, that instead of affecting resistor R the capacitance is changed ($C \to C_1$). While the steady-state response of the new system will remain the same, we will face different alternatives and quite different transient phenomena. There are several choices how to perform such a "reconfiguration":

1. The reconfiguration can be performed by a single switch (see Fig. 2.1.d): capacitor C is replaced by a charge-free C_1. In this case the course of the transient is the same as if the circuit were switched on at t=0.
2. The reconfiguration can be performed in such a way, that the voltage of capacitor C_1 is set somehow to the voltage of capacitor C. See Figs. 2.1.e and 2.1.f for possible solutions. In this case the transient behavior will be the same as with changing the resistor R.
3. The transient behavior can be avoided if the voltage of capacitor C_1 is set to its steady- state value to be reached after reconfiguration. This equals the sum of the first two terms of (2.4) at t_0^+ (t_0^+ stands for the time instant after reconfiguration). In this case the output of the RC divider is typically bumpy.
4. The reconfiguration can be performed in such a way, that we keep the charge of capacitor C, and this charge is loaded on C_1. This can be performed, e.g., by a sudden change of a variable capacitor. Due to this change $U_2(t)$ will take the value:

$$U_2(t_0^+) = \frac{C}{C_1} U_2(t_0^-) ,$$

where t_0^- stands for the time instant before reconfiguration. In this case we get bumpy output, and a quite different transient behavior as before.

We can get a family of further alternatives if both resistor R and capacitor C are changed to get the reconfigured RC divider ($R \rightarrow R_1$, $C \rightarrow C_1$, $R_1 C_1 = T_1$). By selecting a proper ratio of R_1 and C_1, while their product remains unchanged, we can avoid transient behavior at arbitrary time instant $t = t_0$. The only side-effect of this approach is that in the majority of the cases the output will be bumpy.

Based on this simple example we can conclude that

1. Transient phenomena are due to the fact that the calculated steady-state voltage of the capacitor after reconfiguration (at t_0^+) will differ from that of at t_0^-, and therefore a transient motion is unavoidable.
2. The systems before and after reconfiguration are related by the voltage of the capacitor at t_0^-. This voltage governs the transient motion and serves as initial condition at solving the differential equation of the system after reconfiguration.
3. If we are in the position to adjust the voltage of the capacitor at t_0, then we can eliminate or reduce transients; however, transient motions help to provide "smooth" transitions from steady-state to steady-state. If the transients are eliminated or reduced by adjusting the voltage of the capacitor at t_0, the output of the RC divider will be bumpy.
4. It is important to emphasize that in this example the reconfiguration is performed at steady-state, i.e., after the previous transients are already over.

The majority of the alternatives and phenomena described within this simple example will appear in a more detailed form in the subsequent chapters.

2.2 Literature survey

The second part of this introductory section gives a short literature survey about the most important research activities related somehow to reconfigurations and transient phenomena.

Management of reconfiguration transients: There are various reconfiguration methods proposed in the literature to realize the run-time configuration changes at mode transitions, see [S&al93], and [VL98] for overviews. The typical approaches are as follows:

1. one-step reconfiguration;
2. multiple step reconfiguration (series of one-step reconfigurations);
3. input cross-fading methods;
4. output cross-fading methods;
5. signal smoothing;
6. state variable update methods.

The first approach can be considered as a simple switching from one configuration to another, the next four are smoothing techniques that can reduce transient phenomena at the price of a (much) longer transition interval, while the last approach proposes to change, whenever possible, the state variable values, because the transients are due to the mismatch of these variables before and after reconfiguration.

The problem of reconfiguration transients is well understood by control and audio signal processing communities, however, very few research reports exist on strategies for suppressing these transients in time-varying and/or reconfigurable systems. The available results are valid mainly for tunable linear systems, in which the parameters are abruptly changed [VL98], [ZZ88]. The existing methods are based on certain kind of redundancy, which help to "predict" the conditions and settings required to reduce transients at reconfiguration.

The structure dependence of the reconfiguration transients (see Section 3) is, at least to our knowledge, completely unknown in the literature. A systematic approach to handle such transients in nonlinear systems is also missing.

Adaptive control using multiple models: The utilization of multiple models is a possible alternative to implement intelligent control, where the ability to operate in multiple environments by recognizing which environment is currently in existence is a fundamental requirement. In the control literature there are studies, e.g., [NB97], offering general methodology for such adaptive control using multiple models, switching and tuning. In these approaches, instead of having complicated controllers capable to operate properly in a wide range of environmental conditions, there are multiple switched controllers running in parallel, which are selected by a higher-level mechanism. The stability problems of these schemes, see, e.g., [LM99], are investigated in several papers, but none of the researchers have studied the transients due to switching. In [BoMe99] it is emphasized, that the multiple model-based reconfigurable control strategy was suggested, because single model-based adaptive controller may be too slow to bring the closed-loop system close to the new operating regime, which may result in unacceptably large transients. In [Mar00] a general theory of systems described by multiple models is initiated, and a complete characterization of the stability of such a system is given, based on the concept of stable motion between two states. This article, and the literature behind can serve as a starting point of investigating reconfigurations in nonlinear systems. Finally in our view the approach applied in gain-scheduling control, see, e.g., [AA98] might also be helpful in studying transient phenomena of nonlinear systems.

Fault detection, diagnosis and reconfigurable control: The control literature is very rich in studies dealing with fault detection and isolation. Recently these research activities are combined with reconfigurable control and address an integrated approach, see, e.g., [ZJ99]. Here the transients appear in different contexts: (1) faults can be are detected due to signal changes, i.e., transients; (2) to minimize fault induced transients and recover the performance of the system quickly, the reconfiguration should be carried out as soon as possible after fault occurrence. However, the need for explicit transient control is not mentioned. [MB99] develops monitoring, prediction, and fault isolation method for abrupt faults in complex dynamic systems. Predicted transient effects of hypothesized faults are captured in the form of signatures that specify future faulty behavior as higher order time-derivatives. [BG97], [LDC99], [MGWD97] serve as characteristic examples how multivariable adaptive control techniques are applied to flight control reconfigurations.

Trajectory sensitivity and tracking: The interpretation of transients is quite simple if they are motions between steady state behaviors, which can be represented by simple models, like constant, or periodic values. There are, however, applications, where the steady state is hard to define: the operation can be characterized as a complex, continuous motion. These systems are permanently in “transient”, and are represented and designed via trajectories. Transients due to reconfiguration will modify these trajectories, and only model-based approaches might help in separating and/or compensating such effects. In our knowledge up till now there are no systematic methods available to perform such signal separations or compensations. The studies concerning trajectory sensitivity analysis (see, e.g., [HP00]), transient stability prediction (see, e.g., [LT00]), and trajectory tracking (see, e.g., [TVSS98]) might help to find solutions for the transient management problem, as well.

Convergence of learning algorithms: There are many papers (e.g., [CMS99], [Mou97], [MA98], [Solo97], and [Sun93]) that investigate the stability and convergence properties of adaptive algorithms. The motion of adaptive systems from their initial states to their final (stationary) state can be considered as a transient motion, therefore every method for the characterization of the convergence properties might be useful in handling reconfiguration transients.

2.3 Approaches of transient management in reconfigurable systems

In the literature, several alternatives of run-time reconfiguration in controlled systems are reported. In the majority of the cases these reconfigurations are followed by transient phenomena, which act as special disturbances and reduce the quality of the overall performance. Several authors in the literature recognize this fact, but very few results are available to counteract. It is a general opinion that transients persist only for a relatively short time after the change has occurred, therefore the best strategy is to wait, and continue the operation under steady-state conditions. This study concerns applications, where this approach is not acceptable: the speed and the performance of the operation requires management actions to avoid disturbances both in the time and the value domain.

It is easy to realize that the efficiency of transient reduction or elimination is highly dependent on the actual perturbation signals at the input of the system. For this very reason transient management techniques should consider the inputs, or at least a model of the inputs.

The transient management techniques reported here are divided into two major groups: (1) passive and (2) active transient management methods. This second classification is new, and it is a direct consequence of the discovery that it is possible to realize the structure dependence of the reconfiguration transients (see, e.g., [PK99]). The passive transient management methods are design-time techniques, which offer selection mechanisms to help the decision which structure would perform best. The structure here can be considered as a model of computation, or equivalently as a signal-flow graph. Several alternative models of computation suitable for implementation are taken into account, and the most insensitive to the actual reconfiguration (planned for the application) will be selected. Unfortunately the decision here is merely a qualitative one, because the run-time environment at design time is only partially known. The intensive computer simulations have shown that the best structures provide considerably better performance as the widely used direct-form family of implementations.

The active transient methods utilize on-line measurements to provide transient reduction. There are two main approaches applied: (1) the initialization of the state-variables if possible, (2) anti-transient signal injection. The first family of methods can be used in case of “artificial” engineering product, e.g., in case of programmable controllers, while the second one is a universal approach, which can be implemented as an auxiliary controller.

Somewhat different transient management techniques are required if (1) the controller, or (2) the plant, or (3) both the controller and the plant are changed, and if the overall system operates in open-loop, or in closed-loop.

The remaining part of this report follows the above classification and in Section 3 the passive, while in Section 4 the active transient management methods are introduced. The aspects still open for further investigation are summarized in Section 5.

3. Passive methods of transient management

This section is devoted to introduce the so-called passive transient management methods, which utilize the fact that the different structures of implementation produce quite different transients in the case of otherwise identical reconfigurations. The passive methods are design-time analysis techniques, and do not influence the transient phenomena directly. On the other hand the selection of the proper structure requires careful “run-time related” considerations as well, because the actual course of the transients is highly influenced by the actual perturbation signals of the system. Investigation of the step-responses can orientate this selection if the time-domain behavior of the signals is critical, while the case of white noise and sinusoidal inputs can help to characterize the “energy” required or to be dissipated by the transient motions.

This section contains also some considerations concerning multiple-step reconfigurations and provides examples illustrating the strength of the propositions both in open-loop and closed-loop control applications.

3.1 Structure dependence of reconfiguration transients

The step response of time-invariant linear systems is well understood and seems to be an appropriate tool to characterize the transient properties of the system. It is also well known that if the internal energy of the system is originally zero, then the step response can be calculated from the corresponding differential or difference equation. If we change the coefficients of an already operating discrete IIR filter, then the output can be considered as the sum of different components. One is the response to the input, while the others are the responses due to the initial conditions, i.e., due to the stored values, which equal state-variable values, if the state variable formulation is applied. These initial conditions generate an impulse response from the output of the storage device to the output of the filter.

Obviously, in the majority of the practical cases these state-variable values are not known. However, their possible range can be estimated. It is well known from the literature of digital signal processing (see, e.g., [RG75]) that different structures have quite different internal dynamic ranges. As an example, consider the first-order direct structure in Fig. 3.1(a).

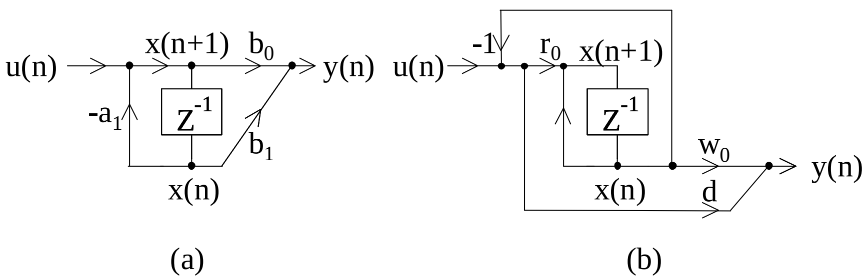


Fig. 3.1. (a) First-order direct structure and (b) first-order resonator-based structure

The state-variable description of this first-order system has the form of

$$
\begin{aligned}
x(n+1) &= a_1 x(n) + u(n) \\
y(n) &= b_0 x(n+1) + b_1 x(n)
\end{aligned}
\tag{3.1}
$$

If $u(n)$ =1 for $\forall n$, $n \geq 0$, then $x(n) \rightarrow 1/(1-a_1)$ as $n \rightarrow \infty$ and can be a large value if a_1 (<1) is close to 1. Note that the dynamic range of this filter is parameter dependent. As a counterexample, consider the structure on Fig. 3.1(b). Its state-variable description is

$$\begin{aligned} x(n+1) &= x(n) + r_0[u(n) - x(n)] \\ y(n) &= w_0 x(n) + d[u(n) - x(n)] \,. \end{aligned} \tag{3.2}$$

If $u(n) = 1$ for $\forall n$, $n \geq 0$, then $x(n) \to 1$ as $n \to \infty$, i.e., it is independent of the coefficients. These two examples show that the nature and the value of the state variables can differ considerably. Obviously, the coefficients of the two structures are different and the reconfiguration will change them differently. Therefore, a complete characterization without knowing the input samples is not possible. Under mild restrictions, however, a very interesting link can be established to structures having minimum round-off noise [MR76], [Hwa77]. These structures can be characterized by relatively uniform energy distribution among the state variables. As a consequence, the output sensitivity to the rounding errors is relatively low, which results in a lower (or minimum) output noise level. In this model, the samples of the noise sources for modeling round-off errors are directly added to the actual state-variable values. Concerning transients we have a similar situation, since the initial conditions behave like additive impulses to the actual state variables. In both cases, the additional energy introduced must be transferred from the state-variables to the output, but in the second case, the time-domain behavior is emphasized more. The requirement for relatively uniform energy distribution is met by the so-called orthogonal structures (see [CP96], [PMP96]). These have very good internal dynamic range, low round-off noise, and can reduce zero-input limit cycles. From the above reasoning, it turns out that they also are good candidates for implementing reconfigurable IIR filters.

It is important to note that in the theory and practice of adaptive filters, this structure dependence is not recognized properly. In the majority of the adaptation schemes the correction terms are based on direct measurements of the output signal. Correction of the parameters can be considered as reconfiguration, producing transients at the output. Since these transients may disturb the overall performance of the adaptive filter, it is reasonable to apply DSP structures with low reconfiguration transients. Another reconsideration is whether adaptation at lower rate results in better performance.

As a simple illustration the step responses of the direct and the resonator-based structures [PMP96] are presented on Figures 3.2-3.7 for different filter orders. The idea behind this is the assumption that the steady-state behavior can be reached faster if, as the first step, a filter with wider bandwidth is operated, followed by a one-step bandwidth reduction. The coefficients for the first-order filters are given in Table 3.1.

direct	resonator-based
$b_{0,old}$ = 0.1367 $b_{1,old}$ = 0.1367 $a_{1,old}$ = -0.7265 $b_{0,new}$ = 0.0155 $b_{1,new}$ = 0.0155 $a_{1,new}$ = -0.9691	$r_{0,old}$ = 0.2735 $w_{0,old}$ = 1.0000 d_{old} = 0.1367 $r_{0,new}$ = 0.0309 $w_{0,new}$ = 1.0000 d_{new} = 0.0155

Table 3.1: Coefficients for the first-order filters

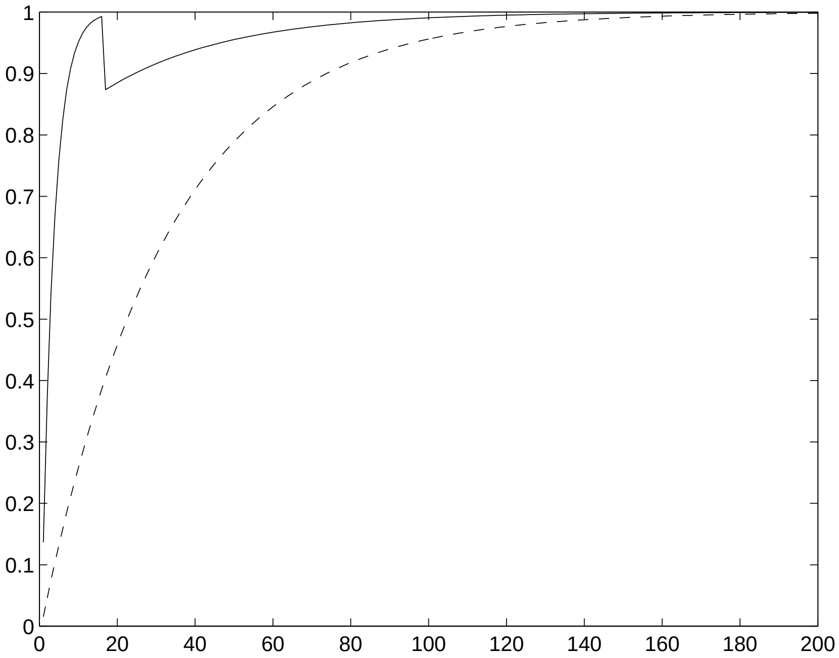


Fig. 3.2: Solid line: Step response of a first-order direct structure reconfigured at step 16.
Dashed line: Step response of the narrow-band first-order direct structure.

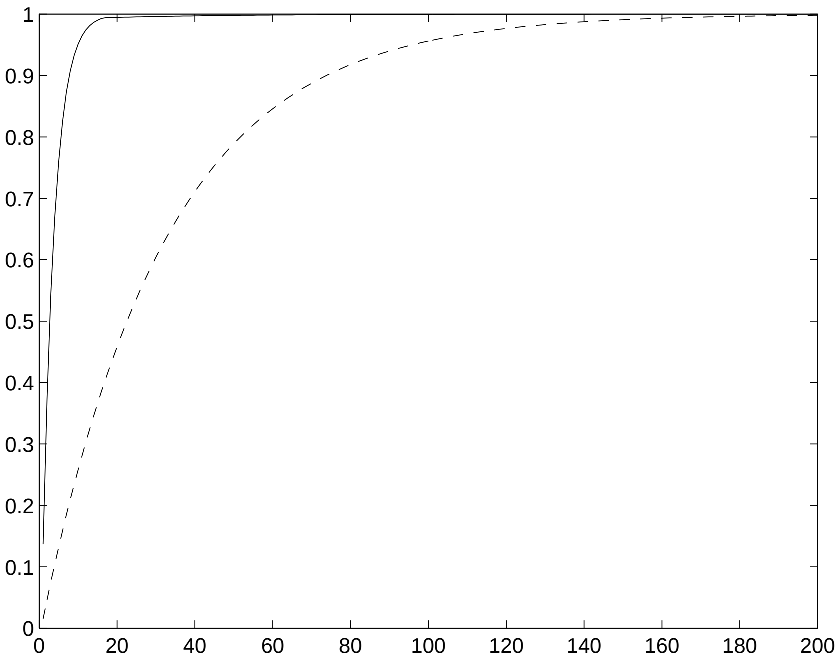


Fig. 3.3: Solid line: Step response of a first-order resonator-based structure reconfigured at step 16.
Dashed line: Step response of the narrow-band first-order resonator-based structure.

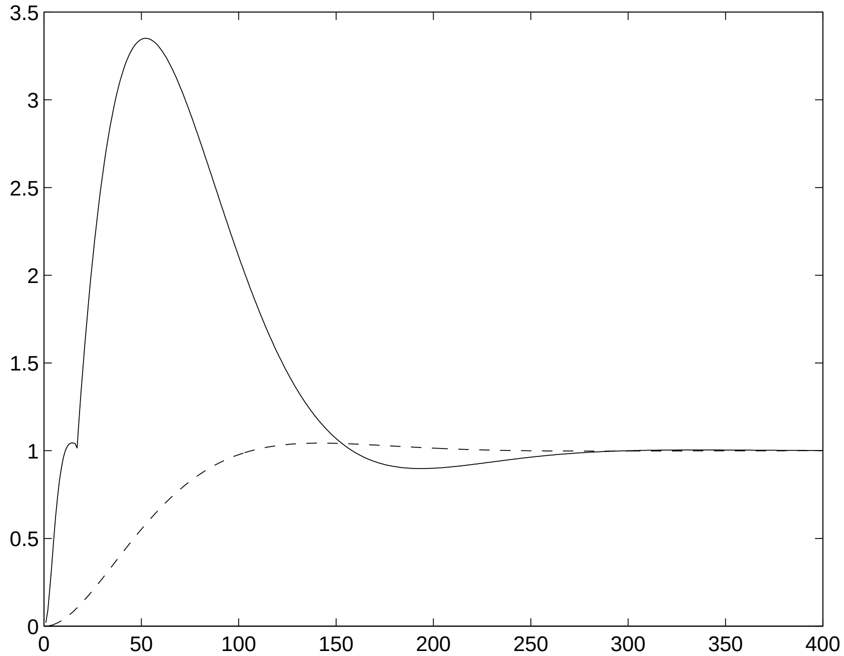


Fig. 3.4: Solid line: Step response of a second-order direct structure reconfigured at step 16.
Dashed line: Step response of the narrow-band second-order direct structure.

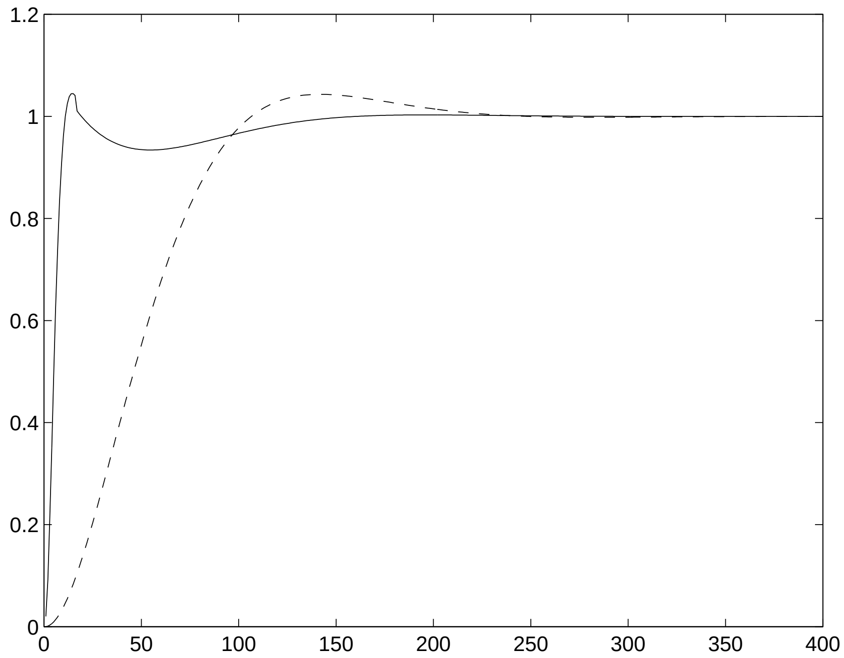


Fig. 3.5: Solid line: Step response of a second-order resonator-based structure reconfigured at step 16.
Dashed line: Step response of the narrow-band second-order resonator-based structure.

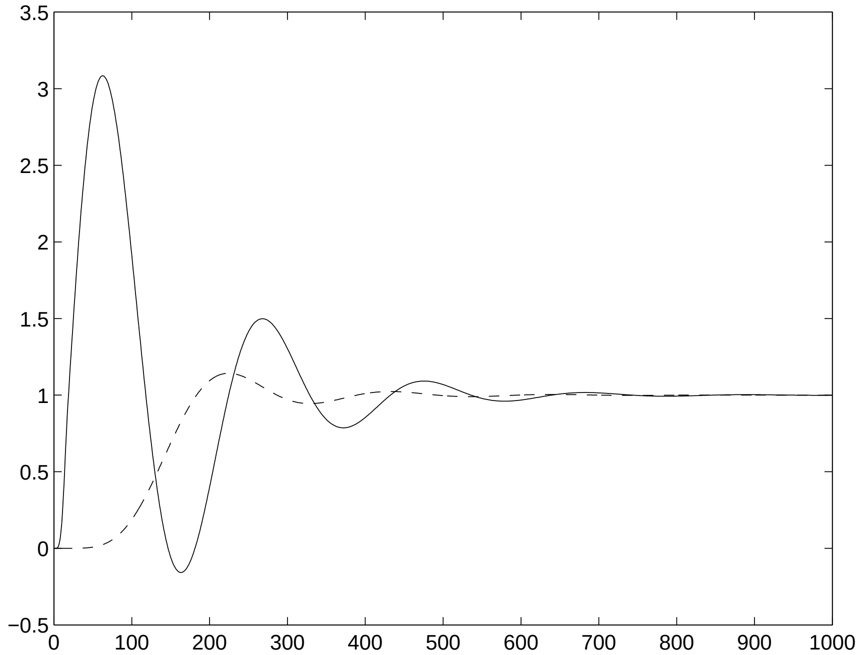


Fig. 3.6: Solid line: Step response of a sixth-order direct structure reconfigured at step 16.
Dashed line: Step response of the narrow-band sixth-order direct structure.

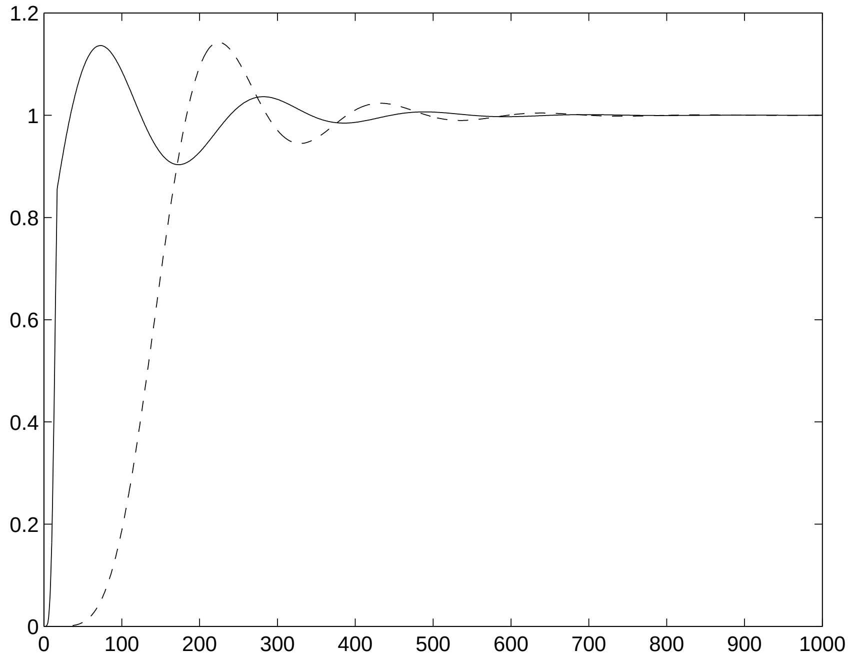


Fig. 3.7: Solid line: Step response of a sixth-order resonator-based structure reconfigured at step 16.
Dashed line: Step response of the narrow-band sixth-order resonator-based structure.

As another simple example Table 3.2 summarizes the behavior of four different alternatives for the realization of the very same first-order difference equation (transfer function). The first three are the most widely used versions of the direct structure [RG75], while the fourth one is again the resonator-based structure [PMP96]. The coefficients provide low-pass behavior, and again the step response is investigated. The outputs $y(n) \to 1$ as $n \to \infty$. It is assumed, that at n_0 the outputs take the value of 0.5. At n_0 +1 with the “old” parameter-set each structure would reach 0.75, while switching to the “new” one results in quite different values. The “new” parameter-set implements a narrower low-pass filter, therefore the step-response will be slower. The most inconvenient behavior is that of the Direct II structure, which produces a negative “jump” at its output. The Direct I, and the transposed Direct II structures have the very same response, while the resonator-based one gives the fastest solution.

Coefficients

direct	resonator-based
$b_{0,old} = 0.25$	$r_{0,old} = 0.5$
$b_{1,old} = 0.25$	$w_{0,old} = 1$
$a_{1,old} = -0.5$	$d_{old} = 0.25$
$b_{0,new} = 0.05$	$r_{0,new} = 0.1$
$b_{1,new} = 0.05$	$w_{0,new} = 1$
$a_{1,new} = -0.9$	$d_{new} = 0.05$

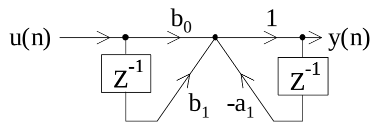
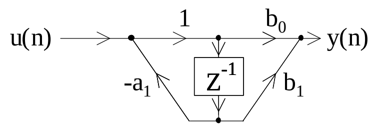
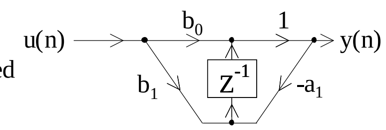
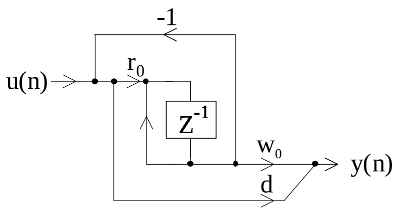
Structures		Output at n_0	Output at n_0+1 old system	Output at n_0+1 new system
Direct I.		0.5	0.75	0.55
Direct II.		0.5	0.75	0.177
Direct II. transposed		0.5	0.75	0.55
Resonator-based		0.5	0.75	0.683

Table 3.2: Reconfiguration transients in different first-order systems during the rising-phase of their step-response.

To get more insight into the transient behavior let’s assume, that at n_0 the outputs take the value of almost 1, and we change the coefficients under such condition. In this case no significant change of the outputs can be detected except at the output of the Direct II structure, where the memory-cell will contain a value of almost 2, and the drastic reduction of the output weights by a factor of five will suddenly reduce the output to $y(n_0+1)=0.24$. This phenomenon is due to the fact described above: the internal (state) variable of the filter is not scaled; therefore it can take very different values for the two configurations. As a conclusion it can be stated that if we apply scaling the size of the transients will be lower.

The simple examples above explicitly show that there is a wide variety of behaviors depending on the implementation structure. It can be concluded that a proper selection from the possible alternatives at design time can significantly reduce the intolerable consequences of the reconfigurations.

3.2 Multiple-step reconfiguration

The multiple-step strategy and the structure dependence are illustrated by a simple pole migration example. The reason for this illustration is the observation that gradually changing the coefficients in certain filter structures may move the poles temporarily out of the unit circle. Unstable temporary pole positions can indicate possible large transients, though they do not cause loss of stability because the proper interpretation of poles [OI01] for time invariant systems gives lighter stability criteria. An 8th order Butterworth low-pass filter was reconfigured from a cut-off frequency of 0.1 f_S to 0.01 f_S, where f_S denotes the sampling frequency. In Figures 3.8-3.11 the pole-migration of four different structures are recorded (the direct [RG75], the normalized lattice [CP96], the resonator-based [PMP96] and the parallel [RG75]). The filter coefficients were linearly interpolated in 100 steps. The direct structure temporarily has lost stability because its poles have migrated out of the unit circle. The parallel structure seems to provide rather good behavior.

A better behavior can be achieved if instead of linearly interpolating the coefficients, the filter design is performed at every step, and in the multiple-step reconfiguration the corresponding coefficients are applied. In this case, the "interpolation" is on the level of the filter specification, i.e., we gradually change the bandwidth and/or the center frequency. The optimal strategy of this "interpolation" is still an open question. However, the considerations related to measurements with sweep generators help.

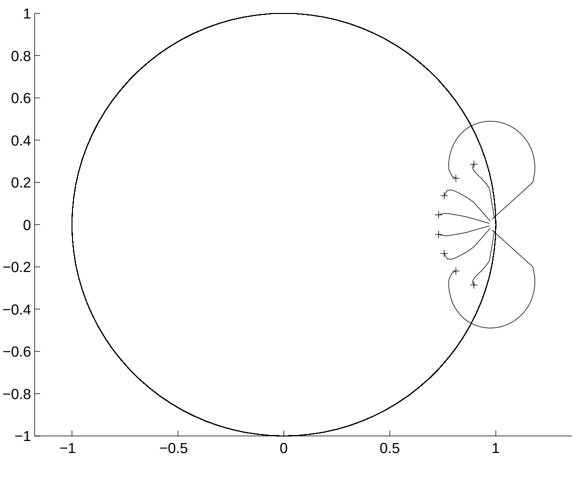


Fig. 3.8: Pole migration of an eighth-order direct filter

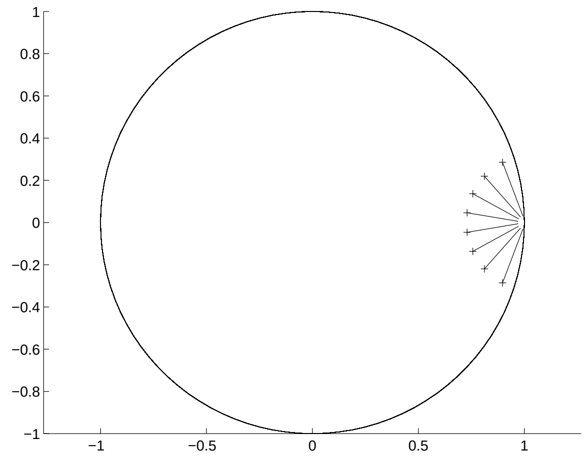


Fig. 3.9: Pole migration of an eighth-order parallel filter

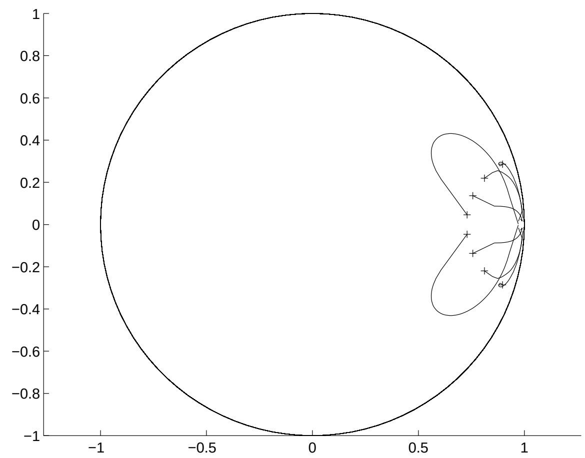


Fig. 3.10: Pole migration of an eighth-order normalized Lattice filter

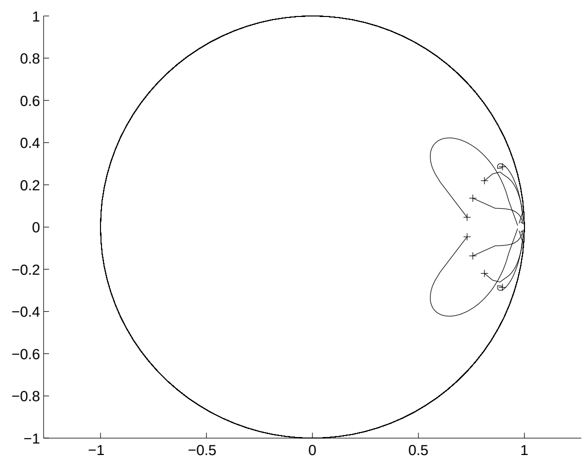


Fig. 3.11: Pole migration of an eighth-order resonator-based filter

3.3 Analysis of reconfiguration transients: the white noise input case

The optimal transition after reconfiguration is highly dependent on the application; therefore, various transient definitions and transient measures can be applied. However, usually the transient of a quantity in the (discrete) time-domain is defined as

$$f_{tr}(n) = f(n) - f_{id}(n) \tag{3.3}$$

where $f_{tr}(n)$ is the transient part or component of the quantity, $f(n)$ is the observed quantity in the investigated reconfigurable system, and $f_{id}(n)$ is the same quantity observed in an ideal system reconfigured without transients. In addition to the definition of the transient, a transient measure is also required to compare the behavior of contending alternatives for transient reduction. For applications, where characterization supposing white noise input seems to be appropriate, the average energy of the transient, defined as

$$\| f_{tr} \|_2^2 = \sum_{n=-\infty}^{\infty} |f_{tr}(n)|^2 \tag{3.4}$$

can be used as a measure of selection. Obviously there are several other possible alternatives, but since in case of (3.4) existing concepts and computational methods can be utilized, as a first attempt it may be an acceptable choice.

The input-output mappings of (linear) dynamic systems defined by their impulse response in the time-domain or by their transfer function in the frequency-domain do not specify how the internal processing is to be performed within the system. It is the state variable formulation, which provides an appropriate framework to describe such situations, where the internal behavior considerably influences the overall performance. Since this is the case with reconfiguration transients, we also try to utilize this approach. With the usual notation

$$\begin{aligned} \mathbf{x}(n+1) &= \mathbf{Ax}(n) + \mathbf{Bu}(n) \\ \mathbf{y}(n) &= \mathbf{Cx}(n) + \mathbf{Du}(n) \end{aligned} \tag{3.5}$$

where the state variable $\mathbf{x}(n)$ describes motions governed also by internal mechanisms of the system. From now, we will consider only state-variable formulations, which are minimal in the sense of state variables and have a nonsingular state transition matrix **A** with full eigenvector system. It is well known that the transfer function of the represented system is invariant to the so-called similarity transformations

$$(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}) \rightarrow (\mathbf{T}^{-1}\mathbf{AT}, \mathbf{T}^{-1}\mathbf{B}, \mathbf{CT}, \mathbf{D}) \tag{3.6}$$

where **T** is a nonsingular transformation matrix. Consequently there exist an infinite number of systems that realize the very same transfer function. Various IIR filter structures [RM87], [PMP96] have been developed that utilize this invariance of the realized transfer function and to achieve certain advantages, e.g., better performance under finite word-length realizations.

In our investigations we consider the transients of a one-step reconfiguration at $n=k$. We compare the transients produced by the tested systems to the outcome of the output switching method [S&al93]. The experimental setup is shown in Fig. 3.12.

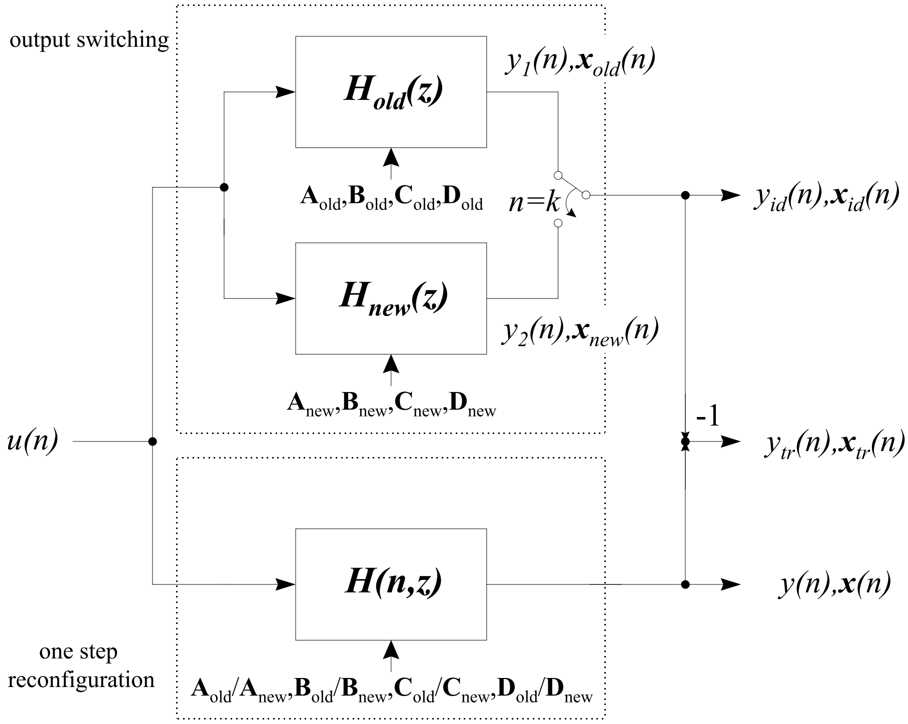


Fig. 3.12: Experimental setup to compare the transient properties of the output switching and the one-step reconfiguration.

We assume that before the reconfiguration all the examined components, i.e., the system to be reconfigured, represented by a time-varying transfer function $H(n,z)$, the "old" system, represented by $H_{old}(z)$, and the "new" system, represented by $H_{new}(z)$, are in steady-state, or at least very close to that. After reconfiguration, however, the state variables inherited from the old configuration [$H(n,z)$ with $\boldsymbol{A}_{old}$, $\boldsymbol{B}_{old}$,$\boldsymbol{C}_{old}$, and $\boldsymbol{D}_{old}$] are not necessarily steady state values for the new configuration [$H(n,z)$ with $\boldsymbol{A}_{new}$, $\boldsymbol{B}_{new}$,$\boldsymbol{C}_{new}$, and $\boldsymbol{D}_{new}$]. The transient of the state variable vector $\mathbf{x}_{tr}(k)$ is the difference of the state variable vector of $H_{new}(z)$ and the state variable vector of $H(n,z)$, which is identical to the state variable vector of $H_{old}(z)$. Therefore, the transient of the state variable vector in the system reconfigured by the one-step reconfiguration can be expressed as

$$\mathbf{x}_{tr}(k)=\mathbf{x}_{old}(k)-\mathbf{x}_{new}(k) \tag{3.7}$$

and the output transient can be computed as

$$\mathbf{y}_{tr}(n)=\begin{cases}0, & \text{for } 0\le n<k\\ \mathbf{C}_{new}\mathbf{A}_{new}^{n-k}\mathbf{x}_{tr}(k), & \text{for } n\ge k\end{cases} \tag{3.8}$$

Based on (3.7) and (3.8), it is possible to estimate the power of the internal state variables, and the output transient for white noise inputs. Here we derive an estimate of the "energy" of the output transient, which gives us a direct way how to select system structures with low transients. Equation (3.7) shows that the state variables play a central role in generating reconfiguration transients. The covariance matrix $\mathbf{K}$ of the state variables

$$\mathbf{K}_{tr}=E\left[\mathbf{x}_{tr}(k)\mathbf{x}_{tr}^{T}(k)\right] \tag{3.9}$$

can be efficiently used in estimating the energy relationships of the system, because for it can be computed from the state variable formulation as

$$\mathbf{K} = \sum_{l=0}^{\infty} \left(\mathbf{A}^l \mathbf{B}\right)\left(\mathbf{A}^l \mathbf{B}\right)^T = \mathbf{AKA}^T + \mathbf{BB}^T \tag{3.10}$$

(**K** is called also as the controllability Grammian (see, e.g., [RM87]).) Equation (3.8) shows how the transient error after reconfiguration settles. This process of the settling in terms of signal energies is characterized by the observability Grammian **W**, the dual of **K**. Matrix **W** is computed as

$$\mathbf{W} = \sum_{l=0}^{\infty} \left(\mathbf{CA}^l\right)^{\mathrm{T}} \left(\mathbf{CA}^l\right) = \mathbf{A}^T \mathbf{WA} + \mathbf{CC}^T \tag{3.11}$$

The matrices **K** and **W** play essential roles in the theory of finite world-length effects in infinite impulse response (IIR) digital filters as shown in [RM87].

The expected value of the energy of the transient signal at the output can be expressed

$$E\left[\sum_{l=0}^{\infty} y_{tr}^2(l)\right] \tag{3.12}$$

With direct substitutions of (3.7), and 3.8) into (3.12) and using (3.11)

$$\begin{aligned} E\left[\sum_{l=0}^{\infty} y_{tr}^2(l)\right] &= E\left[\sum_{l=k}^{\infty} \left(\left(\mathbf{C}_{new}\mathbf{A}_{new}^{l-k}\mathbf{x}_{tr}(k)\right)^T \left(\mathbf{C}_{new}\mathbf{A}_{new}^{l-k}\mathbf{x}_{tr}(k)\right)\right)\right] \\ &= E\left[\mathbf{x}_{tr}^T(k)\mathbf{W}_{new}\mathbf{x}_{tr}(k)\right] \\ &= E\left[\sum_{j=1}^{N}\sum_{i=1}^{N} w_{new,ij} x_{tr,i}(k) x_{tr,j}(k)\right] \\ &= \sum_{j=1}^{N}\sum_{i=1}^{N} w_{new,ij} E\left[x_{tr,i}(k) x_{tr,j}(k)\right] \end{aligned} \tag{3.13}$$

can be derived because $\mathbf{x}_{tr}^T(k)\ \mathbf{W}_{new}\ \mathbf{x}_{tr}(k)$ is a quadratic form and **W** is a real symmetric matrix. The scalar N stands for the order of the systems.

Let us define the covariance matrix of the state variables transient at $n=k$ as

$$\mathbf{K}_{tr} = E\left[\mathbf{x}_{tr}(k)\mathbf{x}_{tr}^T(k)\right] \tag{3.14}$$

which can be converted by substitutions to

$$\mathbf{K}_{tr} = E\left[\mathbf{x}_{tr}(k)\mathbf{x}_{tr}^T(k)\right] = \mathbf{K}_{old} + \mathbf{K}_{new} - E\left[\mathbf{x}_{old}(k)\mathbf{x}_{new}^T(k)\right] - E\left[\mathbf{x}_{new}(k)\mathbf{x}_{old}^T(k)\right] \tag{3.15}$$

The first two terms are the observability Grammians of the old ($\mathbf{K}_{old}$), and the new systems ($\mathbf{K}_{new}$). The last two cross-terms are the covariance of the state variables of the new and the old systems showing how similar the two systems are. Here we use the stationary property of $\mathbf{x}_{tr}$ defined in (3.7). The elements of matrix $\mathbf{K}_{tr}$ are just exactly the $E[x_{tr,i}x_{tr,j}]$ terms in (3.13); therefore, a direct substitution is possible and (3.13) can be rewritten in matrix form as

$$E\left[\sum_{l=0}^{\infty} y_{tr}^2(l)\right] = Tr\left(\mathbf{W}_{new}\mathbf{K}_{tr}\right) \tag{3.16}$$

Both the $\mathbf{W}_{new}$, and the $\mathbf{K}_{tr}$ matrices depend only on the used system structure and the realized transfer functions.

Illustrative example: As an example, four filter structures realizing 10th order pass-band Butterworth filters with the center frequency f_c=0.1 f_s/2, and bandwidth B_{wide}=0.1 f_s/2 for the wide-band, and B_{narrow}=0.01 f_s/2 for the narrow-band filter are designed. The filter structures are the transposed Direct II structure, the parallel, the l_2 scaled orthogonal resonator-based structure [PMP96], and the l_2 scaled orthogonal reverse normalized lattice structure [CP96].

Table 3.3 shows the theoretical estimates of the average output transient energy of the previously listed filters. Table 3.4 lists the average energy of the output transients observed in simulations using the experimental setup of Fig. 3.12. The simulation consisted of 10000 experiments for an identical set of filters.

Structure	Decreasing bandwidth	Increasing bandwidth
Direct structure II transposed	$8.5653\ 10^8$	2.6681
Parallel	2.5123	$2.0047\ 10^8$
Resonator-based (orthogonal)	4.3685	4.5985
Normalized lattice (orthogonal)	4.8686	4.9399

Table 3.3: Average energy of output transients based on our estimation [$Tr(\mathbf{W}_{\mathrm{new}}\mathbf{K}_{\mathrm{tr}})$] for different filter structures.

Structure	Decreasing bandwidth	Increasing bandwidth
Direct structure II transposed	$8.7510\ 10^8$	2.6987
Parallel	2.5268	$2.2190\ 10^8$
Resonator-based (orthogonal)	4.3485	4.5829
Normalized lattice (orthogonal)	4.8658	4.9413

Table 3.4: Average energy of the output transients from 10000 experiments.

As the theory predicts, the direct structure and the unscaled parallel structure produce very high-energy transients in one of the mode-change directions. Mode changes are considered bi-directional; therefore, the application of the direct and unscaled parallel structures is not suggested in reconfigurable systems using the one-step reconfiguration method. The l_2 scaled orthogonal resonator-based and the reverse normalized lattice structures produce the lowest transients overall, independently of the direction of the mode change. Orthogonality of these structures assures that $\mathbf{K}=\mathbf{I}$ independently of the realized transfer function limiting $\boldsymbol{K}_{tr}$, and $\boldsymbol{W}_{new}$ has very good properties too. The transient of the direct and parallel structures can be orders of magnitude higher than that of the filters realized by l_2 scaled orthogonal structures.

3.4 Analysis of reconfiguration transients: the sinusoidal input case

In case of sinusoidal input again there are several alternatives to characterize the transient motions. For linear systems well-known methods are available to provide analytical solutions to describe transient responses as it was illustrated already in Section 2. Here the approach of the previous subsection is followed: the energy of the output transient is calculated as measure for comparing different structures.

We consider a single sinusoid as input

$$u(n) = U\cos(\Theta n + \varphi) \tag{3.17}$$

where U, Θ and φ stand for the amplitude, frequency and phase, respectively. Using the complex notation u(n) = $Re(\overline{U}e^{j\Theta n})$, where $\overline{U} = Ue^{j\varphi}$ is the complex amplitude.

The steady-state behavior of linear, time-invariant and discrete-time systems can be examined using its transfer characteristics $\overline{H} = He^{j\rho}$ and the complex amplitude $\overline{U} = Ue^{j\varphi}$ of the input. The complex amplitude of the output can be expressed as

$$\overline{Y} = \overline{H}\,\overline{U} = He^{j\rho}Ue^{j\varphi} \tag{3.18}$$

H and ρ are functions of Θ, however, this is indicated only in case of possible misunderstanding. The output samples can be given as

$$u(n) = \mathrm{Re}\left(\overline{Y}e^{j\Theta n}\right) = HU\cos(\rho + \varphi + \Theta n) \tag{3.19}$$

For the approach presented in Fig. 3.12 we can derive the transfer values from the input to the state variables x_i at the discrete frequency Θ: $\overline{H}_{o,x_i}$ for the old and $\overline{H}_{n,x_i}$ for the new configuration. The transfer value responsible for the transient from the input to the state variables x_i at the discrete frequency Θ: $\overline{H}_{tr,x_i}$ is exactly the difference of the difference of the two previous transfer values:

$$\overline{H}_{tr,x_i} = \overline{H}_{old,x_i} - \overline{H}_{new,x_i} \tag{3.20}$$

Using (3.20) and (3.19) the transient state $x_{tr,i}(k)$ can be expressed as:

$$x_{tr,i}(k) = H_{tr,i}U\cos(\rho_{tr,i} + \varphi + \Theta k) \tag{3.21}$$

The energy of the transient at the output can be calculated using (3.12) and (3.13), the only difference is that the expectation operator can be neglected. By substituting (3.21) into (3.13) also for i and j we have:

$$\sum_{l=0}^{\infty} y_{tr}^2(l) = \sum_{j=0}^{N}\sum_{i=0}^{N} w_{n,ij}H_{tr,i}H_{tr,j}U^2\cos(\rho_{tr,i} + \varphi + \Theta k)\cos(\rho_{tr,j} + \varphi + \Theta k) \tag{3.22}$$

Utilizing equality $2\cos A\cos B = \cos(A-B) + \cos(A+B)$, where $A = \rho_{tr,i} + \varphi + \Theta k$ and $B = \rho_{tr,j} + \varphi + \Theta k$ we get the form:

$$\sum_{l=0}^{\infty} y_{tr}^2(l) = \sum_{j=0}^{N}\sum_{i=0}^{N} w_{n,ij}H_{tr,i}H_{tr,j}U^2\frac{1}{2}\left(\cos(\rho_{tr,i} - \rho_{tr,j}) + \cos(\rho_{tr,i} + \rho_{tr,j} + 2\varphi + 2\Theta k)\right) \tag{3.23}$$

Equation (3.23) can be decomposed into two parts

$$\begin{aligned}\sum_{l=0}^{\infty} y_{tr}^2(l) \quad = \quad & \sum_{j=0}^{N}\sum_{i=0}^{N} w_{n,ij}H_{tr,i}H_{tr,j}U^2\frac{1}{2}\cos(\rho_{tr,i} - \rho_{tr,j}) + \\ + \quad & \sum_{j=0}^{N}\sum_{i=0}^{N} w_{n,ij}H_{tr,i}H_{tr,j}U^2\frac{1}{2}\cos(\rho_{tr,i} + \rho_{tr,j} + 2\varphi + 2\Theta k)\end{aligned} \tag{3.24}$$

The first part is independent of φ, Θ and k, while second is not. Accordingly assuming sinusoidal input with given frequency and phase, we might reduce transient energy by properly setting the time of the reconfiguration. Unfortunately, however, the determination of this time instant in practical applications requires intensive run-time calculations. The problem here is rather similar to the switching problems of electrical power distribution systems, but in our case the size of the problem space is larger, because the number of components having different magnitude and phase equals the number of the state variables.

3.5 Transient reduction in reconfigurable control systems

If the operational conditions of a controlled plant change, then usually some modifications are to be performed to keep the overall behavior of the within prescribed limits. Reconfigurable control systems are designed to be able to react to failures, compensate the time-variant and/or nonlinear nature of plant, or accommodate changes in the control objectives. Run-time controller reconfigurations, however, may cause unacceptable transients in control loops. In this subsection, the issue of reducing transients due to controller reconfiguration and/or plant changes is investigated. The proposed solutions are based on the "management" of the state-variables of the controller by (1) selecting proper realization structures for the controller and/or (2) by modifying the state-variables of the controller. The properties of the proposed solutions are evaluated by simulation in a "test-bed" consisting of a plant with two modes (operational and failure) and a PID controller.

A typical reconfigurable control system is a compound of a continuous-time plant, a discrete-time controller, a reconfiguration manager, and a supervisory sub-system. The block diagram of such a system is shown in Fig. 3.13.

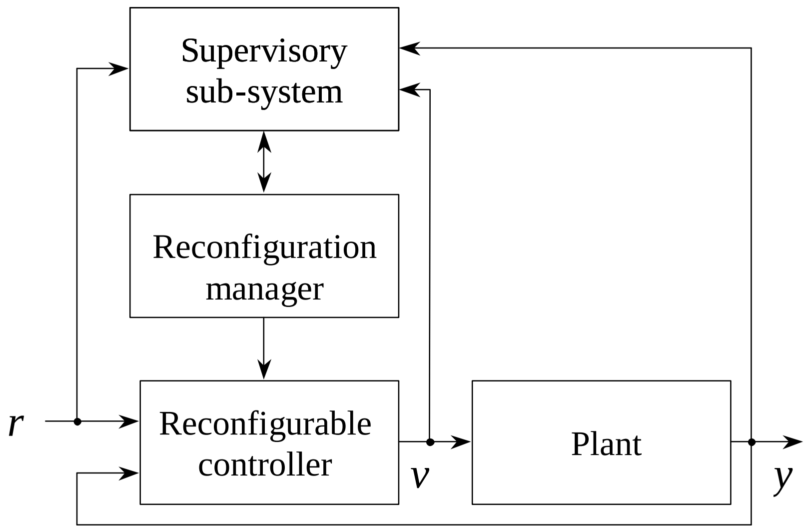


Fig. 3.13: Block diagram of a conceptual reconfigurable control system.

The supervisory system detects the changes and faults of the plant, the actuators, the sensors, the controllers, including the underlying computing infrastructure, and works out a new configuration of the controller that assures proper operation under the given circumstances. In the majority of the cases, the supervisory sub-system can be considered as discrete-event system issuing reconfiguration commands to the controller through a reconfiguration manager. One of the primary tasks of the reconfiguration manager is to minimize the reconfiguration transients caused by the reconfiguration of the controller. Recent investigations ([S&al93], [AW97], [VL98], [PK99], [KPS01]) show that without any interaction, reconfiguration transients might not be tolerated because of their possibly high amplitude and extreme dynamics. For this very reason, it is necessary to address the transient problem by introducing methods that guarantee acceptable transient behavior. But the problem here is far more complicated than in open-loop systems, especially if the plant itself also changes; this is demonstrated in the subsequent analysis.

Let us assume that the plant changes due to a fault, which causes unavoidable transients. Fault Detection and Isolation (FDI) algorithms use the difference of the nominal and observed system outputs, as defined in (3.3), to work out possible fault hypotheses [MB99]. To diagnose faults from transient, the FDI system must have a detailed built-in model of the plant that captures structural information, practically the information required to back-propagate the observed transient to possible changes in the internal energy flow in the physical system, to the actual fault. In essence, the FDI algorithms use the structure dependence of reconfiguration transient.

An immediate effect of the transient due to plant failure is the perturbation of the control loop. The controller tries to compensate for the transient, but it is not designed for that task. The faulty plant requires a different controller selected from stored controller alternatives to satisfy the control objectives. If an appropriate new controller is available, the actual controller has to be reconfigured. This reconfiguration may result in transients, which can be reduced to some extent, because we have full access to the controller realization, i.e., we can modify the signal flow graph, the internal states, and the coefficients, as well. Here we must note, however, that we have very limited access to the plant, we can possibly drive some of its inputs, we can measure some of its internal variables, and sometimes we can make quite limited changes within the plant, therefore a complete elimination of transients within the loop is hardly achievable.

Illustrative example: The experimental simulated setup, shown in Fig. 3.14, consists a continuous time plant and a discrete time PID controller.

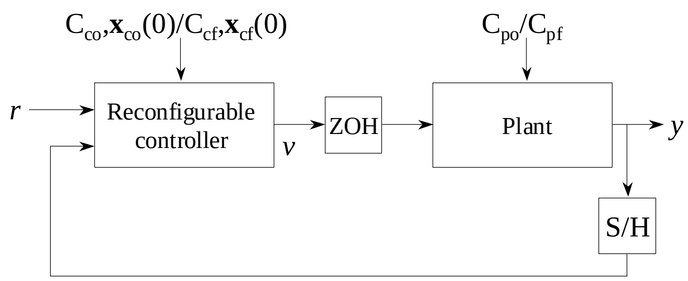


Fig. 3.14: The experimental setup.

The failure of the plant is modeled by the transition from the operational configuration C_{po} to the failure configuration C_{pf}, and the repair is modeled by the transition from C_{pf} to C_{po}. The controller is to be reconfigured from its operational mode C_{co} to its failure mode C_{cf} to properly control the failed plant, and from C_{cf} to C_{co} when the plant is repaired. The state variables of the controller, $\mathbf{x}_{co}$ and $\mathbf{x}_{cf}$, are accessible as shown on Fig. 3.14, they can be read or written any time during operation. The properties of the used system configurations (C_p and C_c pairs) are shown in Table 3.5.

	C_{po}	C_{pf}
C_{co}	no overshot	over-compensated large phase margin
C_{cf}	under-compensated small phase margin	no overshot

Table 3.5: Closed loop properties in case of plant and controller configurations

The failure of the plant is identified by the supervisory sub-system, typically from the transient [MB99] caused by the failure. In one scenario, this step is followed by the controller selection or design. In most of the cases, it is reasonable to assume that the reconfiguration of the controller occurs after the settling of the transient. The repair of the plant is done intentionally, which can be performed separately from the controller reconfiguration, or a joint repair of the plant and the reconfiguration of the controller is also possible. In the experimental setup, the changes of configuration are time driven. The control functions governing the configurations in the experimental setup are shown in Fig. 3.15. Here we must note that the supervisory sub-system should take reconfiguration transients into account, and it should not initiate new reconfigurations resulting in an infinite repetitive sequence of reconfigurations.

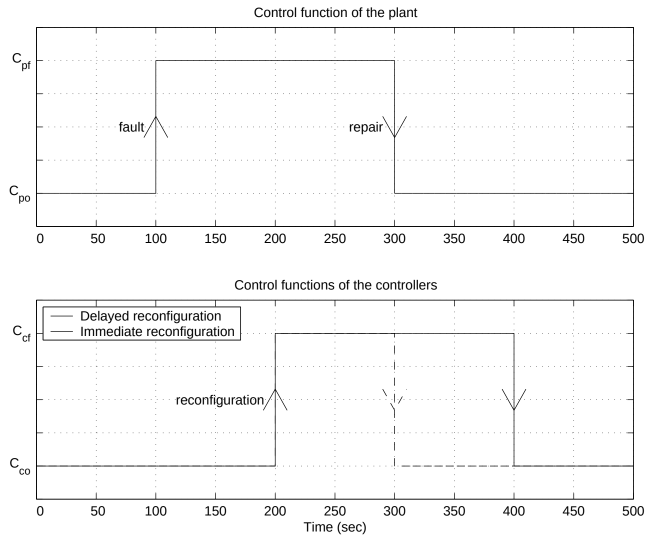


Fig. 3.15: Control functions for the plant and the controllers.

For implementing discrete-time controller the so-called direct structure (see Fig. 3.16 (a) for signal flow graph) is widely because as it can be derived directly from its transfer function. This structure, however, has very bad reconfiguration transient properties as it was shown for IIR filters in [KPS01]. The input scaled state-space structure, shown in Fig. 3.16 (b) and abbreviated as scaled SS in the figures, promises low transient in IIR filters [KPS01]. The scaled state-space structure is used with minor modification as PID realization when the error signal and its derivative are both directly measured from the plant, but its application is possible in all other cases.

Another approach to reduce transients is the modification of the state variables of the controller. The simplest form of state variable modification is the zeroing of the state variables during the reconfiguration. More complex methods of computing new state variables for the new controller are possible, for example, by setting the state variables of the new controller to their future steady states.

In the experiments the reconfigurations are done using the one-step reconfiguration in a direct and a scaled state-space structure, and for reference a third experiment is included in which the state variables of the direct-form controller are zeroed.

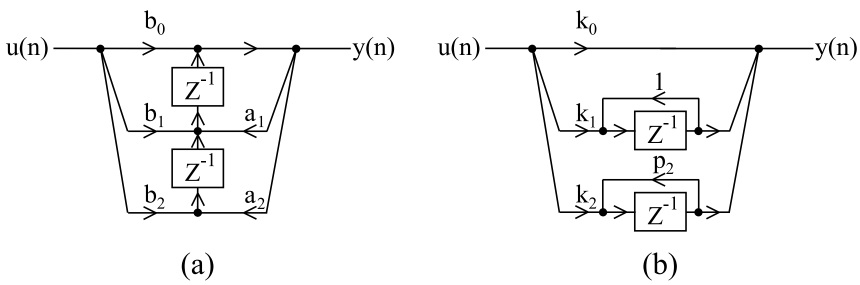


Fig. 3.16: Filter structures for controller realization. (a) Direct structure (b) State-space structure

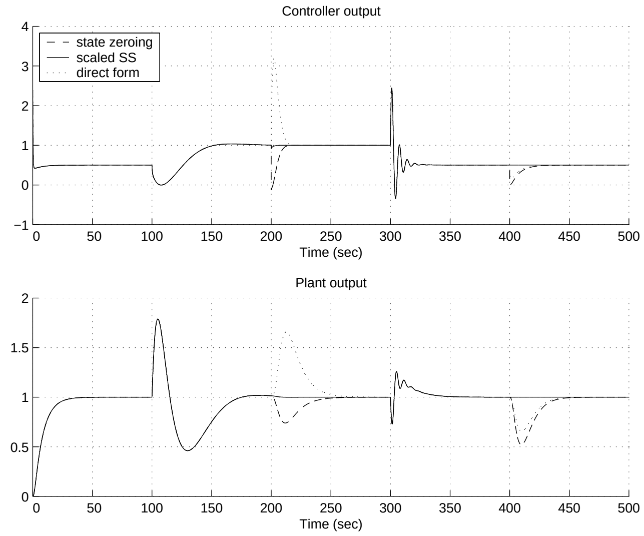


Fig. 3.17: Reconfiguration transients. The fault occurs at 100 sec, the controller is reconfigured according to the plant at 200 sec, at 300 sec the plant is repaired, and finally at 400 sec the controller is reconfigured according to the repaired plant.

First the case of such a scenario is investigated, where the controller reconfiguration after the recovery or repair of the plant is delayed. The controller and plant outputs are shown in Fig. 3.17.

As expected, the failure and repair of the plant cause transients, the course of which is invariant to the structure of the controller applied. On the other hand, the transients due to controller reconfiguration are structure and method dependent. The one-step reconfiguration of the direct structure and the state zeroing method produce large transients compared to the transients of the scaled state-space structure. If the reconfiguration of the scaled state-space controller is performed under steady state conditions, practically no transients are produced. This preferable property is due to the state-space structure, because in steady state the input of the PID controller (the error signal) is practically zero, and the change of the coefficients of all forward paths k_0, k_1, and k_2 do not effect the output of the controller. The same property does not hold for the direct structure.

The joint repair of plant and controller reconfiguration is investigated as the next experiment. The controller and plant outputs are shown in Fig. 3.18.

All experiments show large transients at 300 sec. Unfortunately, state variables of both the direct and state-space controllers inherited from the failure configuration are not appropriate for the new controllers. To reduce the transients even in the case of joint plant repair and controller reconfiguration the state variables of the controller should be modified. By setting the state variables of the controllers in both the direct and state-space structures to their operational steady state we may achieve reduction of transients. The experiment is shown on Fig. 3.19. The system utilizing the state-space controller shows moderate reduction of transients, while the transients of the direct structure controller are increased slightly. The appropriate method of computing the new state variables is under development, but the limits of modifying the state variables are identifiable, as we have no access to the state variables of the plant, which produce transients on its own. At this point it is important to note that the calculation of new state-variable values is considered as active transient management method (for the details see Section 4).

Structure dependence is identified as a key factor in transient management of reconfigurable control systems. Using proper structure for controller realization guarantees no transient if the reconfiguration is performed in steady state of the system, and assures smaller transients than other structures for small disturbances. Transient property of the joint reconfiguration of the plant and the controller is an open question, but proper initialization of the state-variables of the controller can reduce transients.

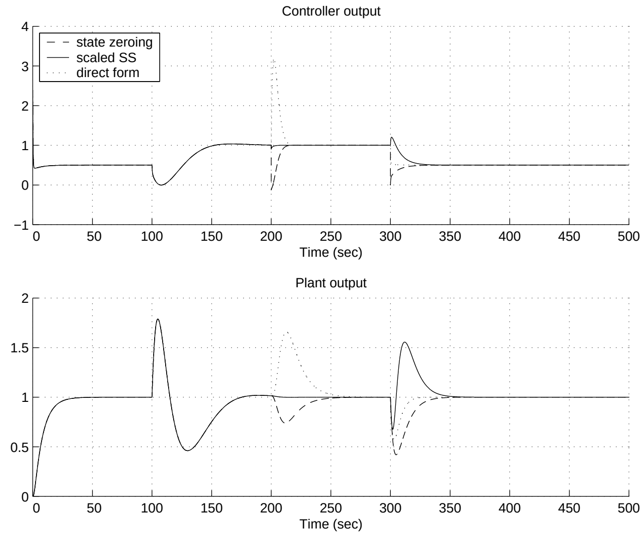


Fig. 3.18: Reconfiguration transients. The fault occurs at 100 sec, the controller is reconfigured according to the plant at 200 sec, and at 300 sec the plant is repaired parallel with the controller reconfiguration.

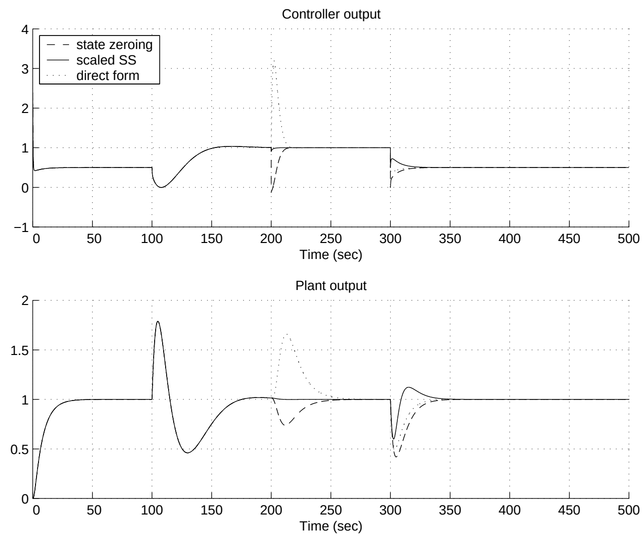


Fig. 3.19: Reconfiguration transients with state modification. The fault occurs at 100 sec, the controller is reconfigured according to the plant at 200 sec, at 300 sec the plant is repaired parallel with the controller reconfiguration. The internal states of the controllers are set to their final steady states at 300 sec.

4. Active methods of transient management

Although the proper choice of the system's structure helps to decrease transients caused by the system's reconfiguration, usually the passive methods do not give full protection against the undesired transients effects (especially in those cases when the system's structure is given, and the designer has no free choice). Active methods can be effective counterparts of the passive methods in every case when run-time interaction is possible and the system has free computational power to run the transient management algorithms. Unlike the passive methods, active methods manipulate the system run-time when the reconfiguration happens in order to decrease the transient effects.

However, active transient management methods have serious drawbacks: additional (sometimes remarkable) computational power is required, and their application field is much narrower than that of the passive methods. Due to the model-based nature of the active methods, certain restrictions may apply to the reconfigured system:

- The reconfigured system must not be in transient state before reconfiguration;
- The reconfigured system is supposed to be linear, and its linear model must be known;
- The system is reconfigured abruptly, in one time instant;
- The input of the reconfigured system may be restricted to a narrow class of signals (slowly changing or constant).

Two basic active transient management methods will be described: state initialization and anti-transient signal injection. The state initialization methods can be effectively used when the inner states of the reconfigured system are accessible and can be modified, which is the case in most software components (e.g. digital filters and digital controllers). When the state variables of the reconfigured system component cannot be modified, the anti-transient signal injection method still may be applied to reduce the transient effects.

The transient management techniques can be used after the reconfiguration (or at the same time), or some of them may be applied before the reconfiguration act, if it is know in advance.

Different versions of the transient management methods can be used in both the open loop (signal processing chain) and closed loop (control) scenarios.

Section 4.1 describes the basic active transient management techniques in open loop scenarios, with special emphasis on the applicability in closed loops. In Section 4.2 the described methods are generalized for closed loops, and the necessary modifications are described. In both sections the applicability of the methods are investigated. In Section 4.3 two examples illustrates the described methods in simulated control environments.

4.1 Active anti-transient methods in open loop systems

The effect of reconfiguration (e.g. change of parameters) of a system in a processing chain may affect the elements connected to the reconfigured system; it may cause harm and degrade the system's overall performance. In the signal-processing framework successful techniques have been proposed to cancel or suppress transients caused by the abrupt changes of filter coefficients (see [VL98] for an overview). In a frequently used model the coefficients (and possibly the order of the filter as well) are changed abruptly.

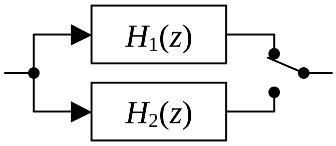


Fig. 4.1. Output switching method

In this context the transient is defined as the difference between the actual output and the ideal steady-state output of the new filter. The so-called *output-switching* method uses the above definition and runs filters in parallel, as shown in Fig. 4.1. This solution provides transient-free output but the overhead is high. More sophisticated methods use (usually simpler) transient eliminator filters to collect information and to aid the initialization of the new filter. At the time instant of the change not only the filter parameters, but also the filter states are updated using the information gathered by the transient eliminator filter [VL98, ZZ88]. This elegant solution is part of a more general framework where the state variables are updated so that the transient effects are minimized or cancelled.

The state variables of a system are often hidden, cannot be accessed or modified (typical examples are the mechanical systems). During the reconfiguration of such systems transient management methods utilizing state variable initialization cannot be used. In such cases the undesirable transient effects can still be suppressed or cancelled. The basic idea is the injection of an anti-transient signal in the signal processing chain in order to decrease the undesired transient effect at a critical point of the processing chain. The place of injection depends on the physical system. An obvious choice would be the injection *after* the reconfigured system (see Fig. 4.2.a). In this case the injected signal is simply the inverse of the transient caused by the reconfigured system. The calculation of the injected signal is straightforward in model-based systems, where all the necessary information is available (measured or calculated), and theoretically, the transient can perfectly suppressed. However, the intervention is hard or even impossible after the reconfigured system in many practical cases, when the output of the changed system is not an electrical quantity (typical examples are again mechanical systems). In such cases the injection must be placed *before* the reconfigured system, usually at the actuator's input (see Fig. 4.2.b). The latter situation is harder to solve, in many cases perfect suppression cannot be achieved, but the reduction of the transients is usually still possible.

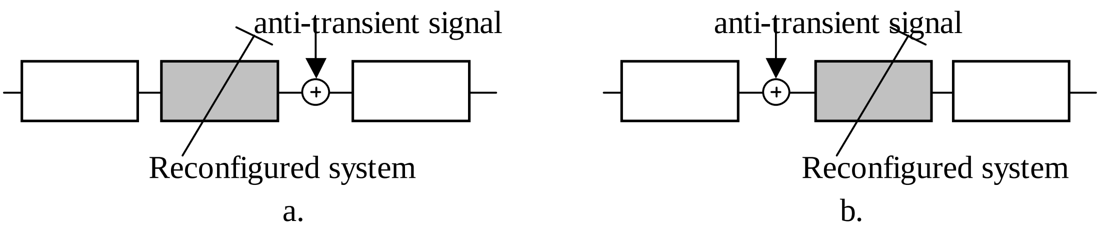


Fig. 4.2. Injection of anti-transient signals. *a*. after, *b*. before the reconfigured system

Although in the next sections the open loop transient management is considered, emphasis is made to those methods, which are applicable (directly or in a modified version) to closed loop scenarios as well.

4.1.1 State variable initialization

Reconfiguration transients depend significantly on the actual internal energy conditions of the reconfigured systems. With a state-variable interpretation the internal energy of a dynamic system is distributed among their state variables. At reconfiguration these state variable values serve as initial conditions causing transients. These state variables also give the possibility to suppress reconfiguration transients through proper initialization.

In practical cases two simple state variable initialization techniques are used if reconfiguration is needed: state zeroing and state preserving. Both methods can be considered as very simple state updating algorithms.

State Zeroing Method: When the system's parameters are changed, the state variables are set to zero. This is a usual and 'safe' way of reconfiguration, since the new system starts from a zero-energy state, thus its behavior is predictable. Another advantageous property of this simple method is that the structural representation has no effect on the transient. However, apart from the rare case when the state variables are all zeros, this method always produces transients.

State Preserving Method: This method preserves the old system's state variables after the parameter change. This method may be useful when the successive systems are similar (i.e. the successive parameter sets are similar), and the old states are also 'meaningful' in the new system. The application of this method is troublesome if the structure is also changed. A drawback of this solution is that the transient

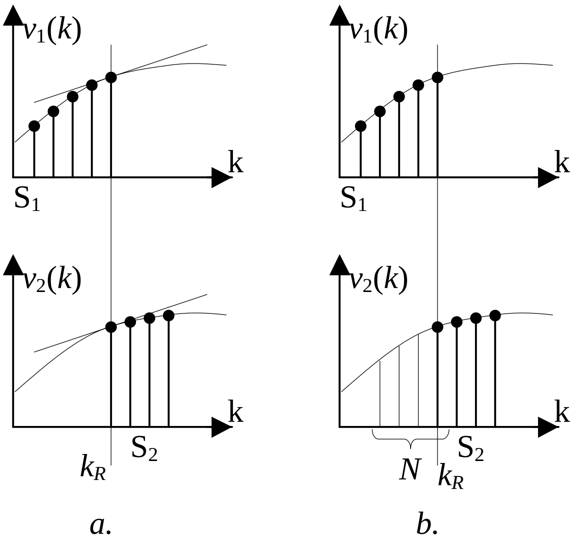


Fig. 4.3. Output Fitting strategies. The reconfiguration is made at time instant k_R. *a.* The new system (S_2) is initialized to continue the output of the old system (S_1) smoothly, by fitting the derivatives.
b. The new system is initialized as if it produced the past few samples of the old system.

behavior of the new system depends on the last state of the old system, thus it is unpredictable in design time.

Both state zeroing and state preserving methods may have their field of application, but generally neither of them provide satisfactory results. Hereinafter a more adequate solution will be described [SKP00].

Output Fitting Method: In practical cases 'smoothness' or 'bumplessness' is often a useful characterization of the output of a reconfigured system. Abrupt changes or large peaks are usually undesirable, while a smooth transition is natural and often unobservable (e.g. in case of an audio system). Based on this fact the transient management algorithm can be constructed, which tries to keep the output signal of the system 'smooth' around the reconfiguration. A reasonable definition of smoothness is based on Taylor-series. Let

$$\begin{aligned} y_1(t_R) &= y_2(t_R), \\ \dot{y}_1(t_R) &= \dot{y}_2(t_R), \\ y_1^{(2)}(t_R) &= y_2^{(2)}(t_R), \\ &\vdots \\ y_1^{(N-1)}(t_R) &= y_2^{(N-1)}(t_R), \end{aligned} \tag{4.1}$$

where $y_1^{(l)}(t_R)$ is the lth (left-hand side) derivative of the old system's output at the reconfiguration time instant t_R, and $y_2^{(l)}(k_R)$ is the lth (right-hand side) derivative of that of the reconfigured system. In case of an equidistantly sampled discrete system the derivatives are replaced simply by differences $y^{(l)}(k_R) \cong y(k_R) - y(k_R - l)$ and $y^{(l)}(k_R) \cong y(k_R + l) - y(k_R)$ for the left-hand side and right-hand side, respectively, where k_R is the discrete reconfiguration time.

The transient management algorithm using 'equal derivatives' to provide smooth transition is illustrated in Fig. 4.3. *a*.

Assuming for a while, that the new system operates in parallel with the old one, and thus produces output values *before* the reconfiguration, the right-hand side derivatives of the new output signal can be replaced by the left-hand side derivatives. Having discrete samples, the equality of the left-hand side derivatives in Equation (4.1) implies the equality of the samples preceding the reconfiguration:

$$y_1(k_R - i) = y_2(k_R - i), \text{for } i = 0 \ldots N-1. \tag{4.2}$$

Thus a possible way to ensure the smoothness of the system output is to run the old and new systems *virtually* in parallel so that both systems produce the same outputs before reconfiguration, as illustrated in Fig. 3. *b*. In this case the new system can continue the operation 'smoothly', or, in other words, its state variables contain values which enable the smooth reconfiguration. Of course, instead of running the systems in parallel (which would be impossible in closed loop, anyway), the state variables are computed from the constraint that the past N output samples of the new system are the same as that of the old one, provided their inputs are the same.

Note that this method cannot ensure the *complete* rejection of transients, since the initial state variables enforced to the new system are not really produced by it. These imperfect initial values change, and a new stationary state is reached, which may also cause small transients. But the level of these secondary transients is much smaller than that of the transients without proper management, according to practical experiments.

Computation of the initial values: There are multiple ways for the calculation of the state variables if the output fitting method is used, depending on the time and hardware resource constraints. If the reconfiguration must be performed immediately, then the recalculation of the state variables must be made in one time instant. In this case a linear equation system must be solved with N unknown variables (the state variables) and N equations (the relationship between the input and output in the past N samples). The form of the equations strongly depends on the reconfigured system's structure, and can be solved using different well-known techniques, but in general for higher N it has high computational complexity (order N^2).

If the reconfiguration can be delayed by N samples, or the reconfiguration command is known at last N samples before the required reconfiguration time instant, or it is possible to run an auxiliary filter in parallel with the system to be reconfigured, then it is possible to solve the equations in a recursive manner. This recursive method will briefly be described here.

Let us consider the state-variable representation of the system *after* the reconfiguration:

$$x_{k+1} = A_2 x_k + B_2 u_k ,$$
$$y_k = C_2 x_k + D_2 u_k , \tag{4.3}$$

where x is the state vector, u and y are the input and output of the system, respectively, A_2 is the state transition matrix, and B_2, C_2 are the input and output coupling matrices of the reconfigured system, respectively. For this system a dead-beat observer [Lue71] can be designed with the following structure:

$$z_{k+1} = A_2 z_k + B_2 u_k + g(y_k - C_2 z_k) - gD_2 u_k , \tag{4.4}$$

where z is the observed state vector and g is a parameter vector which has to be chosen adequately to provide the dead-beat behavior. The constraint is that all roots of the polynomial $\det(\lambda I - A_2 + gC_2)$ are zeros. From this constraint parameter vector g can be calculated off-line. The complexity of the above observer is the same as that of the reconfigured system. When operated, the observer must be switched on N samples before the reconfiguration.

Note that it is possible, that the order of the system is higher than the required number of fitted derivatives (N). In this case the initialization can be made by different sets of parameters, all of them satisfying the constraint in Equation (4.2), but they may be different from the viewpoint of transient suppression. In these cases, unfortunately, no general solution exists; the structure of the system must be taken into account to decide which variables to update, and how. See Section 4.3 for an example.

4.1.2 Applicability of the state initialization methods

The state zeroing and state preserving methods – due to their simplicity – do not have limit in application. It must be emphasized that the state zeroing method will always produce the same reconfiguration transient, while the state preserving method is sensitive to the actual inner states as well as to the system structure.

The output fitting method tries to modify the inner state so that the output is bumpless. To the effectiveness of this method some care must be taken. As was mentioned, the goal of the initialization is to mimic a parallel operation of the new system before the reconfiguration, and the output of the new system will be a 'smooth' sequel of the output of the old system. Smoothness was provided by the equality of the derivatives of the output systems. However, the derivatives were approximated by differences in the sampled system; and for N derivatives a sequence of length N was used in the approximation. For the validity of this approximation it must be true that the output signal's derivatives are (approximately) constant during the time period of length N. For this the input signal must be slowly changing according to the sampling period.

Also note that there is no sense using the output fitting technique if the system is already in a transient state during the reconfiguration.

Thus the assumptions and requirements for the usage of the output-fitting method can be summarized as follows:

- The system is reconfigured abruptly, in one time instant.
- The reconfigured system is linear and its state variables can be read and arbitrarily initialized.
- The system's input signal is sufficiently over-sampled (or simply speaking, the input signal is 'slowly' changing).

- The system is in steady state (or close to it) before it is reconfigured.

4.1.3 Anti-transient signal injection

As shown in Fig. 4.2, the anti-transient signal injection can interfere with the transient signal either applying it after or before the reconfigured system. In many practical cases (e.g. mechanical systems with electrical input) the only reasonable solution is to inject the anti-transient signal *before* the reconfigured system (Fig. 4.2.b). In this case the perfect anti-transient signal would be:

$$u(k) = H^{-1}(z) y_{tr}(k), \tag{4.5}$$

where $u(k)$ is the anti-transient signal, $y_{tr}(k)$ is the transient to be suppressed, and $H(z)$ is the reconfigured system's transfer function. Unfortunately the inverse system can be sensitive to noise, and usually it is not causal. Using another approach, the design of the anti-transient signal can be considered as an optimization problem [SPK01].

The goal is to minimize the output transients in some sense. A physically reasonable measure is the power of the output transient. If the power of the transient is taken into consideration for very long time, then this measure perfectly describes the transient's effect in terms of its undesired power. Note that in practical cases the calculation of the transient is usually made for a shorter time to save computational power. In some cases it may be dangerous, since the system's inner states may not contain the ideal values by the end of the considered time interval, and thus the transient may continue.

Another measure can be the energy of the error in the inner states of the system, with respect to the true energy values. If the state error is zero then the transients have decayed.

The two measures can be combined, and thus the measurement of a shorter period of the output error can provide information on the caused transient during the measurement, and the state error is a reasonable measure of the *possible* transients of the system after the measurement period.

The following injection solution uses the simultaneous minimization of the state variable mismatch and the plant output transients. The problem is formulated as a least-squares problem: The measure of the transient error is a sum of two components: the output error power during the time of the injection, and the state-mismatch power after the injection. The first component keeps the output error low during the injection, while the second component tries to minimize the state error at the end of the injection, and thus to provide low output error after the injection stopped.

The operation of the transient injection in time can be *a priori* or *a posteriori,* with respect to the reconfiguration.

- *A posteriori transient signal.* Typical example is when an unexpected, but quickly detected fault occurs in the system. The anti-transient injection starts *after* the system's 'reconfiguration' (i.e. the fault).
- *A priori transient control*. If the system reconfiguration is known in advance (e.g. deliberate mode change) then the transient injection can begin *before* the system itself is reconfigured.

Since the *a posteriori* case can be considered as a special case of the *a priori* control, in the following the more general *a priori* anti-transient injection will be described. The number of injected samples after the reconfiguration is N_2 (in both cases), while N_1 denotes the number of injected samples before the reconfiguration (in case of *a priori* anti-transient injection).

The system to be reconfigured is supposed to be linear, so it's discrete state-space approximation is:

$$x_{k+1} = A_i x_k + B_i u_k \tag{4.6}$$

$$y_k = C_i x_k \tag{4.7}$$

where x, u, y are the state variables, the input, and the output, respectively. The matrices A_i, B_i, and C_i describe the system before (i=1) and after (i=2) the reconfiguration. The error system can be described similarly:

$$\tilde{x}_{k+1} = A_i \tilde{x}_k + B_i \tilde{u}_k \tag{4.8}$$

$$\tilde{y}_k = C_i \tilde{x}_k \tag{4.9}$$

where $\tilde{x}$ is the state error (i.e. the difference between the ideal and the actual states), $\tilde{y}$ is the output error (i.e. the difference between the ideal steady-state output, and the actual output), and $\tilde{u}$ is the injected anti-transient signal. The output error sequence can be expressed as a function of the injected input sequence:

$$\mathbf{y} = \begin{bmatrix} \Xi_1 & \mathbf{0} \\ \Phi & \Xi_2 \end{bmatrix} \mathbf{u} + \begin{bmatrix} \Theta_1 & \mathbf{0} \\ \Psi & \Theta_2 \end{bmatrix} \begin{bmatrix} \tilde{x}_{01} \\ \Delta x_{id} \end{bmatrix} \tag{4.10}$$

where

$$\mathbf{y} = \begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \vdots \\ \tilde{y}_{N_1} \\ \tilde{y}_{N_1+1} \\ \tilde{y}_{N_1+2} \\ \vdots \\ \tilde{y}_{N_1+N_2} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \tilde{u}_0 \\ \tilde{u}_1 \\ \\ \tilde{u}_{N_1-1} \\ \tilde{u}_{N_1} \\ \tilde{u}_{N_1+1} \\ \\ \tilde{u}_{N_1+N_2-1} \end{bmatrix}, \tag{4.11}$$

$$\Xi_i = \begin{bmatrix} C_i B_i & & & & \\ C_i A_i B_i & C_i B_i & & & \\ C_i A_i^2 B_i & C_i A_i B_i & C_i B_i & & \\ & \vdots & & \ddots & \\ C_i A_i^{N_i-1} B_i & C_i A_i^{N_i-2} B_i & C_i A_i^{N_i-3} B_i & \cdots & C_i B_i \end{bmatrix} \tag{4.12}$$

$$\Phi(j,k) = C_2 A_2^j A_1^{N_1-k} B_1, \ \ j = 1 \ldots N_2, \ \ k = 1 \ldots N_1 \tag{4.13}$$

$$\Theta_i(j) = C_i A_i^j, \quad j = 1 \ldots N_i \tag{4.14}$$

$$\Psi(j) = C_2 A_2^j A_1^{N_1}, \quad j = 1 \ldots N_2 . \tag{4.15}$$

The initial error vector $\tilde{x}_{01}$ contains the error of the state variables before the injection, while Δx_{id} is the difference between the ideal state variables in steady state before and after the reconfiguration of the system:

$$\tilde{x}_{01} = x_{id,1} - x_0 \tag{4.16}$$

$$\Delta x_{id} = x_{id,2} - x_{id,1}, \tag{4.17}$$

where $x_{id,1}$ and $x_{id,2}$ are the ideal (steady state) state values of the systems before and after the reconfiguration, respectively.

The state error variables after N_1+N_2 steps can be expressed as follows:

$$\tilde{x}_{N_1+N_2} = \left[A_2^{N_2}\Gamma_1 \middle| \Gamma_2 \right]\mathbf{u} + \left[A_2^{N_2} A_1^{N_1} \middle| A_2^{N_2} \right]\begin{bmatrix} \tilde{x}_{01} \\ \Delta x_{id} \end{bmatrix}, \tag{4.18}$$

where

$$\Gamma_i = \left\lfloor A_i^{N_i-1}B_i \middle| A_i^{N_i-2}B_i \middle| \cdots \middle| A_i B_i \middle| B_i \right\rfloor \tag{4.19}$$

The goal is to minimize the effect of the transient (i.e. the power of the output error), and the state error after the injection. The power of the combined vector $\begin{bmatrix} \mathbf{y} \\ \tilde{x}_{N_1+N_2} \end{bmatrix}$ can be minimized by solving the following equation in the least-squares sense:

$$0 \cong \begin{bmatrix} \mathbf{y} \\ \tilde{x}_{N_1+N_2} \end{bmatrix} = \begin{bmatrix} \begin{bmatrix} \Xi_1 & \mathbf{0} \\ \Phi & \Xi_2 \end{bmatrix} \\ A_2^{N_2}\Gamma_1 | \Gamma_2 \end{bmatrix}\mathbf{u} + \begin{bmatrix} \begin{bmatrix} \Theta_1 & \mathbf{0} \\ \Psi & \Theta_2 \end{bmatrix} \\ A_2^{N_2}A_1^{N_1} | A_2^{N_2} \end{bmatrix}\begin{bmatrix} \tilde{x}_{01} \\ \Delta x_{id} \end{bmatrix} \tag{4.20}$$

The least-square solution is:

$$\mathbf{u} = \begin{bmatrix} \begin{bmatrix} \Xi_1 & \mathbf{0} \\ \Phi & \Xi_2 \end{bmatrix} \\ A_2^{N_2}\Gamma_1 | \Gamma_2 \end{bmatrix}^{\#}\begin{bmatrix} \begin{bmatrix} \Theta_1 & \mathbf{0} \\ \Psi & \Theta_2 \end{bmatrix} \\ A_2^{N_2}A_1^{N_1} | A_2^{N_2} \end{bmatrix}\begin{bmatrix} \tilde{x}_{01} \\ \Delta x_{id} \end{bmatrix} \equiv \mathbf{S}\begin{bmatrix} \tilde{x}_{01} \\ \Delta x_{id} \end{bmatrix}, \tag{4.21}$$

where $X^{\#}$ denotes the pseudo-invert of X.

Notes:

- If the reconfiguration is made deliberately (mode change), and both the old and new system models are known *a priori* then the matrix $\boldsymbol{S}$ in (4.21) can be computed off-line. In this case the on-line computational complexity is quite low and proportional to the number of states and the length of the injected signal.

- The effect of the output error and state mismatch can be weighted to provide an unequally balanced solution.

- The least-squares formulation is unable to set limits on the input signal; thus the amplitude of the solution can be higher than a given physical limit. In this case other ‘guard-states’ are introduced (e.g. delayed or filtered version of the input), the weighted power of which is also taken into consideration in the solution. By the adjustment of this weight factor the amplitude of the solution can be arbitrarily limited. Note that this is not the optimal solution any more.

If the injection can be made *after* the reconfigured system, as shown in Fig. 4.2.a, the solution can be calculated as follows:

Now the error system is an autonomous system, producing the transient signal

$$\tilde{x}_{k+1} = A_2 \tilde{x}_k \qquad (4.22)$$

$$\tilde{y}_k = C_2 \tilde{x}_k \qquad (4.23)$$

with $\tilde{x}_0 = x_{id,2} - x_0$ at he reconfiguration time. The injected anti-transient is simply $-\tilde{y}_k$.

4.1.4 Applicability of the anti-transient signal injection

Since the anti-transient signal method implicitly assumes the knowledge of the transient signal to be suppressed, it's application field is narrower then that of the state initialization methods. During the solution it was used that the system is in a constant steady-state ($x_{id,1}, x_{id,2}$ are constants). Observability and controllability of the controlled system were also implicitly assumed (it is necessary for the existence of the inverse in Eq. 4.21). So the assumptions and requirements for the usage of the anti-transient signal can be summarized as follows:

- The system is reconfigured abruptly, in one time instant.
- The system is in steady state before it is reconfigured, the input is constant (or slowly changing, relative to the sampling interval).
- The linear model of the reconfigured system (before and after the reconfiguration) is available.
- The reconfigured system is observable and controllable.

4.2 Active anti-transient methods in closed loop systems

In the following transients in the classical series-compensation control loop (shown in Fig. 4.4) are investigated. The discrete time controller gets the error signal *e*(*k*), (which is the difference of the reference signal *r*(*k*) and the plant output *y*(*k*)) as input, and produces the control input signal *v*(*k*). Transients in this control loop can occur due to three different sources:

(a) changing the reference; perturbations,
(b) controller (C) adaptation/reconfiguration,
(c) plant (P) “reconfiguration” (mode changes, faults, degradation, recovery, repair).

Case (a) is handled by the properly chosen controller. Cases (b) and (c), however, can be considered as special disturbances that the designer may have information about, and the appropriate usage of this additional information may enable the effective reduction of the caused transients in certain cases. From now on only transients due to reconfigurations (cases (b) and (c)) are considered.

If the controller (C) is reconfigured, and steady-state state-variable values of the controller are different

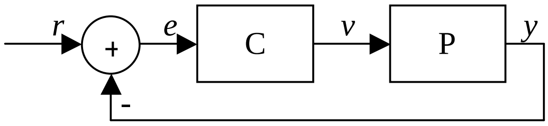


Fig. 4.4. The feedback control loop system. C: controller, P: plant

before and after the reconfiguration, then a transient phenomenon will appear, unless proper counteractions are taken. For this scenario the initialization of the controller's state variables can be a good solution, especially in case of processor-based digital controllers, where state variables are available and can easily be modified.

A more difficult situation is the plant (P) reconfiguration, which can occur because the plant is changed either deliberately (e.g., multi-modal operation), and thus the change is known *a priori*; or because of a sudden system fault or external event, and the change is detected only after its occurrence. If no counteractions are taken, the transient reduction depends completely on the properties of the closed loop. Unfortunately, methods based on state variable initialization are not applicable, because usually the inner states of the plant are not accessible and/or cannot be modified arbitrarily. However, anti-transient signal injection can be a good alternative.

In a third scenario the plant (P) and controller (C) are changed simultaneously. A typical example is when certain criteria require the reconfiguration of the plant and the controller. This is the most important, and of course, the most difficult situation, which will be handled by the mixture of methods used in the simple controller and simple plant reconfiguration cases. In conjunction with the controller reconfiguration, an adequate initialization of the new controller's state variables is carried out so that the reconfiguration of the controller does not induce additional transients. The plant's transients are suppressed by an extra anti-transient signal.

Although the transient suppression usually cannot be perfect, the closed control loop cancels the rest of the transients. The purpose of the anti-transient injection is to decrease the effect of the transients and speed up the cancellation.

A typical history of a fault-adaptive control system is shown in Fig. 4.5. After normal operation the plant is 'reconfigured' (fault). The new operational mode is recognized, and a new adequate controller is designed and applied (controller reconfiguration). When the plant is repaired, both the plant and the controller are reconfigured. These three typical reconfiguration scenarios are discussed in the remainder of this subsection.

The definition of the transient in the case of a closed loop control system is a straightforward generalization of that of the open loop systems. The transient caused by the reconfiguration is the difference between the actual plant output and the hypothetical steady-state plant output (i.e. assuming the new controller or plant was used for a long time).

4.2.1 Transient management in case of controller reconfiguration

The following assumptions are made when the controller is reconfigured in the control loop:

- The controller is changed abruptly, in one time instant.
- The controller is a linear system of which the state variables can be read and arbitrarily initialized.
- The controller's input signal is sufficiently over-sampled.
- The system is in steady state (or close to it) when the controller is reconfigured.
- The reconfiguration is independent of the reference signal (the reference is not changed at
- reconfiguration).

These requirements are the same as those of the open loop case in Section 4.1.2, except for the last one, which is a special case of the steady-state requirement: this is to emphasize that the change of the reference signal may induce transients that make the bumpless fit worthless and meaningless.

In the case of the controller initialization there are no restrictions concerning the plant's behavior, so it may be a nonlinear dynamic system, as illustrated in Section 4.3 through an example.

The solution is simply the bumpless fit described in Section 4.1.1. The controller is a linear system described by its state space representation according to Equation (4.3), where x is the controller's state vector, A,B,C,D are the coupling matrices, and

$$\begin{aligned} v(k) &\equiv y(k), \\ e(k) &\equiv u(k). \end{aligned} \tag{4.24}$$

The calculation of the initial state variables can be carried out by using the dead-beat observer defined by Eq. (4.4).

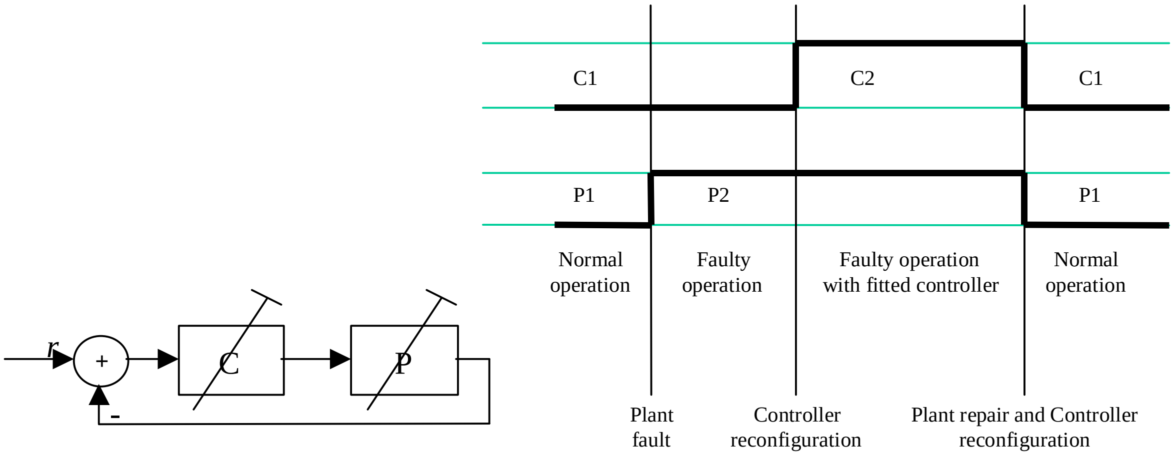


Fig. 4.5. A typical fault adaptive control system history

4.2.2 Transient management in case of plant reconfiguration

When the plant is reconfigured in the control system, the state variable initialization methods cannot be used to decrease the transient effect since the inner states of a physical plant are usually cannot be arbitrarily modified. However, the anti-transient signal injection still remains a possible solution. In the open-loop case two typical scenarios were discussed (see Fig. 4.2). Considering a real control loop with a mechanical plant, only case *b* has technical meaning, and the anti-transient signal can be added to the control input signal before the actuator (Fig. 4.6). To use the injection method, the following restrictions and assumptions are made, similarly to the open-loop case, described in Section 4.1.4:

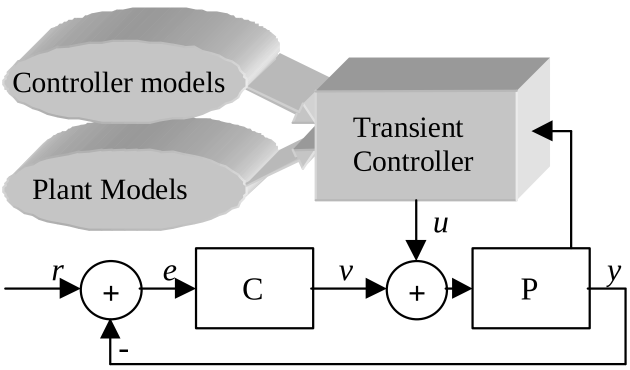


Fig. 4.6. Active transient control with anti-transient signal injection

- The plant is reconfigured abruptly, in one time instant.
- The states of the plant cannot be modified.
- The control system is in steady state when the reconfiguration happens.
- The desired signal is constant or slowly changing, relative to the sampling interval.
- The linear model of the plant (before and after the reconfiguration) is available. Note that here the discrete state-space representation (approximation) of the plant is used.
- The plant and the controller are both observable and controllable.

If the plant is reconfigured deliberately (e.g., repair after a fault, or mode change), *a priori* injection can be used. If the reconfiguration is not known in advance (e.g., a fault) the *a posteriori* injection must be used. The plant's transients are suppressed by the anti-transient signal, which is produced by the Transient Controller (ATC see Fig. 4.6). The ATC knows the plant and controller models and has information on their actual states. The calculation of the injected signal can be carried out using the results obtained in the open-loop case (see Section 4.1.2). However, in order to take into consideration the closed loop's dynamics, the open-loop results must be generalized and the controller's dynamic behavior must also be described. Let us use the controller's linear state-space representation:

$$x_{k+1}^{(C)} = A^{(C)} x_k^{(C)} + B^{(C)} e_k \tag{4.25}$$

$$v_k = C^{(C)} x_k^{(C)}, \tag{4.26}$$

where $e_k = r_k - y_k$. The plant's state space representation is the following:

$$x_{k+1}^{(P)} = A_i^{(P)} x_k^{(P)} + B_i^{(P)} (v_k + u_k) \tag{4.27}$$

$$y_k = C_i^{(P)} x_k^{(P)}, \tag{4.28}$$

where lower index i in the coupling matrices represents the plant's behavior before (i=1) and after (i=2) the reconfiguration. The error systems can be described as follows:

$$\tilde{x}_{k+1}^{(C)} = A^{(C)} \tilde{x}_k^{(C)} + B^{(C)} \tilde{y}_k \tag{4.29}$$

$$\tilde{v}_k = C^{(C)} \tilde{x}_k^{(C)} \tag{4.30}$$

for the controller and

$$\tilde{x}_{k+1}^{(P)} = A_i^{(P)} \tilde{x}_k^{(P)} + B_i^{(P)} (\tilde{v}_k + u_k) \tag{4.31}$$

$$\tilde{y}_k = C_i^{(P)} \tilde{x}_k^{(P)} \tag{4.32}$$

for the plant. The whole loop can be described by the unified state-space error system:

$$\begin{bmatrix} \tilde{x}_{k+1}^{(P)} \\ \tilde{x}_{k+1}^{(C)} \end{bmatrix} = \begin{bmatrix} A_i^{(P)} & B_i^{(P)} C^{(C)} \\ B^{(C)} C_i^{(P)} & A^{(C)} \end{bmatrix} \begin{bmatrix} \tilde{x}_k^{(P)} \\ \tilde{x}_k^{(C)} \end{bmatrix} + \begin{bmatrix} B_i^{(P)} \\ 0 \end{bmatrix} u_k \tag{4.33}$$

$$\tilde{y}_k = \begin{bmatrix} C_i^{(P)} & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_k^{(P)} \\ \tilde{x}_k^{(C)} \end{bmatrix} \tag{4.34}$$

Now using the following notations:

$$A_i \equiv \begin{bmatrix} A_i^{(P)} & B_i^{(P)} C^{(C)} \\ B^{(C)} C_i^{(P)} & A^{(C)} \end{bmatrix}, \tag{4.35}$$

$$B_i \equiv \begin{bmatrix} B_i^{(P)} \\ 0 \end{bmatrix}, \tag{4.36}$$

$$C_i \equiv \begin{bmatrix} C_i^{(P)} & 0 \end{bmatrix}, \tag{4.37}$$

$$\tilde{x}_k \equiv \begin{bmatrix} \tilde{x}_k^{(P)} \\ \tilde{x}_k^{(C)} \end{bmatrix} \tag{4.38}$$

$$\tilde{u}_k \equiv u_k \tag{4.39}$$

the problem can be formalized as follows:

$$\tilde{x}_{k+1} = A_i \tilde{x}_k + B_i \tilde{u}_k \tag{4.40}$$

$$\tilde{y}_k = C_i \tilde{x}_k \,, \tag{4.41}$$

which is the same as in case of the open loop.

Using this notation, the solution is technically the same as in the case of open loop (Equations (4.10)–(421)).

The dynamics of the closed loop make the computation more complex, especially if the controller is of high order. To keep the computational cost lower, a simplified version can be used, where only the plant is taken into consideration as if it was in an open loop. This solution is not optimal, since it doesn't take into consideration the effect of the remaining transients in the control loop, but according to experiments, together with the suppression of the closed loop, it also gives satisfactory results with a modest computational cost (see example in Section 4.3).

4.2.3 Transient management in case of joint controller-plant reconfiguration

The most complicated scenario is when the plant and controller are changed simultaneously. It may happen in case of a deliberate mode change, when the change of the plant requires a new controller for the optimal performance. The tools used in the simple controller and plant reconfiguration cases are applicable here as well. The transients caused by the controller's reconfiguration are effectively suppressed by the proper initialization of the controller, while the transients of the plant reconfiguration are handled by an injected anti-transient signal.

The following constraints are supposed to be valid when the combined state-initialization/transient injection method is used:

- The system is in steady state when the reconfiguration happens
- The controller and the plant are reconfigured abruptly, in one time instant.

- The desired signal is constant or slowly changing, and the control goal is to provide the smooth steady-state output even if the controller and the plant are changed.
- The controller's state variables can be modified, but the states of the plant cannot.
- The model of the controller and the plant (before and after the reconfiguration) is available.
- The plant and the controller are observable and controllable.

The proposed anti-transient design has the following main steps:

- Determine the state variables of the plant and the controller, and the values of v and y in steady state in the closed-loop system, for the given reference input (r). Note that having the (observable and controllable) models of the plant and the controller, the steady-state values can be calculated [Shi88].
- Calculate the injected control signal to minimize the effect of the plant's reconfiguration. The goal is to minimize the power of the transient as well as the state mismatch after the injection. Use either method described in Section 4.2.2.
- Calculate the controller's states to provide smooth transition as described in Section 4.2.1, neglecting the transients possibly caused by the plant's reconfiguration.
- Reconfigure the plant and the controller with state initialization for the controller, and anti-transient signal injection for the plant.

Notes:

- Since the output-fitting methods requires transient-free controller, for this combined method *a priori* injection cannot be used.
- The proposed algorithm is not optimal in the sense that it handles the effects of the plant and controller reconfiguration separately.

4.3 Examples

The transient management algorithms described in the previous sections are illustrated here with simulation examples. The first example is the control of a nonlinear two-link planar robot arm, while the second is the roll-control of a simple airplane. The robot example illustrates the controller reconfiguration in the case of a continuously changing nonlinear plant, while the airplane example illustrates the aggressive reconfiguration of the plant with anti-transient injection.

Two-link planar robot arm: The controlled system is a two-link planar robot arm (see Fig. 4.7), which is a strongly nonlinear mechanical system [Cra89]. Two separate is digital controllers were used, one on each joint. The robot arm is controlled so that the end position of the second arm (where the tool is applied) moves on a triangle. To illustrate the reconfiguration effects, two modes are present in the system. When the modes change, the joint controllers are reconfigured. In the example, MODE 1 is the lower part of the triangle, and MODE 2 is the upper part (see the upper right plot of Fig. 4.8). In MODE 1 controller set 1, while in MODE 2 controller set 2 is used.

The joints are controlled by simple digital PID controllers. The input of each controller is the appropriate joint position error e_k (the difference between the desired and measured joint angle). The simple control algorithm is the following:

$$x_1(k) = x_1(k-1) + T_S \cdot e(k-1), \tag{4.42}$$

$$x_2(k) = e(k-1), \tag{4.43}$$

$$y(k) = P \cdot e(k) + I \cdot x_1(k) + D \cdot (e(k) - x_2(k)) / T_S, \tag{4.44}$$

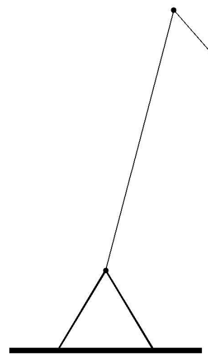


Fig. 4.7. The simulated two-link planar robot used in the example. The links are 0.6m and 0.8 m long, their mass is 1kg and 0.5 kg (concentrated in the joints), and the friction coefficients are 3 kgm²/s.

where *P*, *I*, and *D* are the parameters of the controller, T_S is the sampling interval, and $x(k)$ is the two-element parameter vector of the controller. The structure of the controller is a parallel representation, as can be seen in Equations (4.42)–(4.44). In order to make the effects more eye-catching, the structure is not scaled.

Thus the reconfiguration is made in time instant 10s (MODE 1 $\rightarrow$ MODE 2) and in time instant 14s (MODE 2 $\rightarrow$ MODE 1). Four different transient management (TM) techniques were used to conduct system transitions, as it's shown in Fig. 4.8. The left column in the figure shows the planned and measured joint angles. The uppermost plot shows the planed joint angles for each joint (in radians). The lower four plots show the measured joint errors in mrad, for each reconfiguration techniques.

The upper-right plot shows the planned tool trajectory, showing MODE 1 (thin line) and MODE 2 (thick line). The lower plots show the tool-trajectory error for each reconfiguration techniques. The thick line shows the trajectory error caused by the reconfiguration, while the thin line is the tracking error caused by the not perfect controller.

State zeroing method: The state variables are set to zero when the controller's parameters are changed. As the second row of Fig. 4.8 clearly shows, the reconfiguration caused large transients in the joint angles, thus producing large tool position errors, too.

State preserving method: The state variables of the new controller are initialized to the values of the old ones. This solution may be useful when the successive parameter sets are similar, but in the example, the reconfiguration causes large transients (because the controllers are quite different), as shown in the third row of Fig. 4.8.

Output fitting method 1: The state variables are initialized so that the new controller's output and it's first derivative be the same as those of the old controller's. The transients caused by the reconfiguration can be seen in the fourth row of Fig. 4.8.

Output fitting method 2: The second state variable (x_2) keeps the value of that of the old controller, while the first one (x_1) is used to fit the output trajectory of the controllers. Having one degree of freedom the derivatives are not fitted in this case, but the results are very nice: the transient effects are low as shown in the last row of Fig. 4.8.

Airplane roll control: As an illustration the roll control of an airplane has been simulated ([Shi88], P7.5). Only dynamics concerning the roll motion were simulated using a simple linear model. The original aircraft has two identical poles at *s*=-4, while the reconfigured dynamics have poles at *s*=-3 and

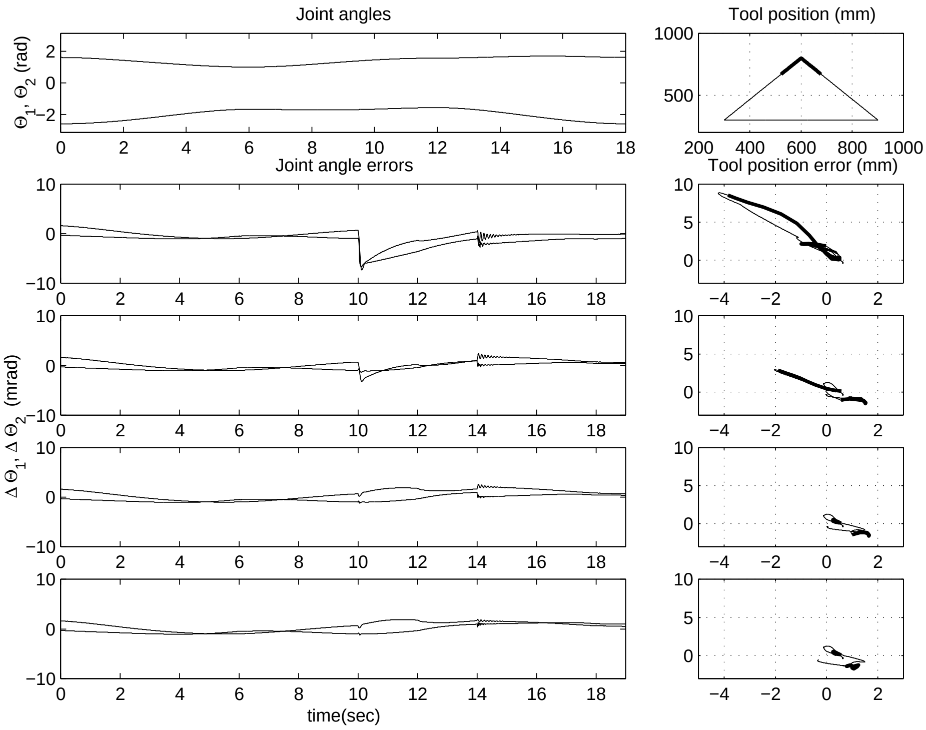


Fig. 4.8. Reconfiguration transients of the two-joint planar robot arm.

s=-5. The plant (airplane) changed at time instant 50s. The controller was not changed in the example. Fig. 4.9 shows the transient behavior without transient control, and with the introduction of two different anti-transient signals. The *a posteriori* control used N_1=0 and N_2=100, while the a priori signal was designed with values N_1=40 and N_2=100. It is clearly visible that the anti-transient signal decreased both the amplitude and the length of the undesired reconfiguration transient.

Table 4.1 contains additional information on different anti-transient signals: the output error power and the computational times. T_{cacl1} and T_{calc2} are the computational times with and without the calculation of matrix $\boldsymbol{S}$ in (4.21), respectively, measured on a Pentium III class computer in Matlab.

N_1	N_2	Output error power	T_{calc1}	T_{calc2}
0	0	1.81	0	0
0	20	0.93	0.1s	0.08ms
0	40	0.48	0.2s	0.10ms
0	100	0.20	0.9s	0.12ms
10	20	0.64	0.1s	0.08ms
10	40	0.32	0.2s	0.11ms
20	100	0.08	1.1s	0.13ms
40	100	0.04	1.2s	0.14ms

Table 4.1: Properties of the Anti-Transient Control signals for the roll-control example with different parameters.

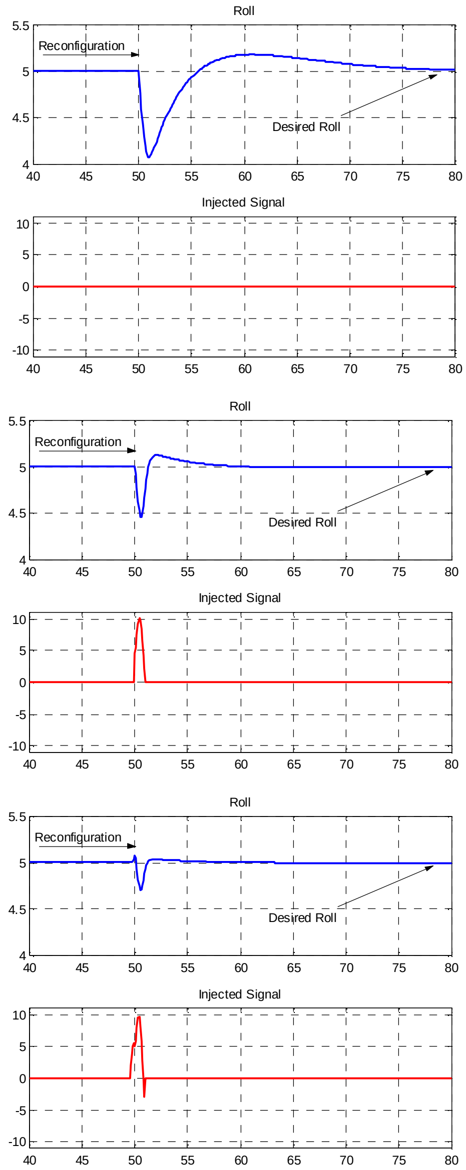


Fig. 4.9. Simulated anti-transient control of an airplane.
a. no injection, b. *a posteriori* injection, c. *a priori* injection

5. Open problems of transient management

There are several research and implementation problems that deal with reconfigurations. What we can offer by now to reduce or eliminate reconfiguration transients can be summarized as follows:

1. Linear models of computation and selection mechanisms supporting design-time activities.
2. Controller state estimation and initialization mechanisms to be implemented in run-time.
3. Anti-transient injection strategies based on on-line state observation.

In the majority of our developments we have considered situations, where reconfigurations are performed while the overall system is in steady state, the “reconfiguration” appears as an abrupt change, and it is assumed that the specification requires reaching a new steady state position. The structure dependence was investigated only for linear models of computation, and also the illustrative examples deal with linear systems. However, due to the inherently nonlinear nature of switching systems, the majority of the proposed methods is not based on assumptions valid only for linear systems, and therefore can be generalized for handling reconfiguration transients of nonlinear systems.

In our view the most important research problems in this field are the followings:

1. Extension of the already existing techniques toward nonlinear control systems.
2. Investigations to get more insight into joint plant and controller reconfigurations.
3. Investigation of reconfigurations, where, in addition to the parameters, the structure is also to be changed (e.g., changing to a controller of different order).
4. Investigation of structure dependence in case of nonlinear models of computation (nonlinear structures).
5. Elaboration of methods for separating and/or compensating trajectory components in case of systems without steady states.
6. Extension of the available results for the case of multiple-step reconfigurations.
7. Analysis of the computational overhead of reconfiguration and transient management.

References

[AA98] Apkarian, P., R.J. Adams, “Advanced Gain-Scheduling Techniques for Uncertain Systems,” *IEEE Transactions on Control Systems Technology*, Vol. 6, No. 1, pp 21-32, Jan. 1998.

[AW97] Arehart, A.B., W.A. Wolovich, “Bumpless Switching in Hybrid Systems,” in Antsaklis, P., W. Kohn, A. Nerode, S. Sastry, (Eds.), *Hybrid Systems IV, Lecture Notes in Computer Science, no. 1273.* New York: Springer Verlag, 1997.

[BeMo99] Bemporad, A., M. Morari, “Control of systems integrating logic, dynamics and constraints,” Automatica, Vol. 35. pp. 407-427, 1999.

[BG97] Bodson, M., J.E. Groszkiewicz, “Multivariable Adaptive Control for Reconfigurable Flight Control,” *IEEE Trans. on Control Systems Technology*, Vol. 5, No. 2, pp. 217-229, March, 1997.

[BoMe99] Boskovic, J.D., R.K. Mehra, “Stable Multiple Model Adaptive Flight Control for Accommodation of a Large Class of Control Effector Failures,” *Proceedings of the American Control Conference,* San Diego, California, June 1999, pp. 1920-1924.

[CMS99] Carini, A., V.J. Mathews, G.L. Sicuranza, “Sufficient Stability Bounds for Slowly Varying Direct-Form Recursive Linear Filters and Their Applications in Adaptive IIR Filters,” IEEE Trans. on Signal Processing, Vol. 47, No. 9, pp. 2561-2567, Sept. 1999.

[CP96] Chung, J.-G., K.K. Parhi, *Pipelined Lattice and Wave Digital Recursive Filters.* Norwell, MA: Kluwer, 1996.

[Cra89] John. J. Craig, *Introduction to robotics: mechanics and control*. Addison-Wesley Publishing Company, New York, 1989.

[GNRR93] Grossmann, R.L., A. Nerode, A.P. Ravn, and H. Rischel, Eds., *“Hybrid Systems,” Lecture Notes in Computer Science, no. 736,* New-York: Springer-Verlag , 1993.

[HP00] Hiskens, I.A., M.A. Pai, “Trajectory Sensitivities of Hybrid Systems,” *IEEE Transactions on Circuits and Systems - Part I: Fundamental Theory and Applications,* Vol. 47, No.2, pp. 204-220, Febr. 2000.

[Hwa77] Hwang, S.Y., “Minimum uncorrelated unit noise in state-space digital filtering,” *IEEE Trans. Acoustics, Speech & Signal Processing*, Vol. ASSP-25, pp. 273-281, Aug. 1977.

[KPS01] Kovácsházy, T., G. Péceli, Gy. Simon, “Transients in Reconfigurable Signal Processing Channels,” *IEEE Transactions on Instrumentation and Measurement*, Vol. 50, No. 4, pp. 936-939, Aug. 2001.

[LDC99] Lyshevski, S.E., K.R. Dunipace, R.D. Colgren, “Identification and Reconfigurable Control of Multivariable Aircraft,” *Proc. of the American Control Conference*, San Diego, California, June 1999, pp. 2732-2736.

[LM99] Liberzon, D., A.S. Morse, “Basic Problems in Stability and Design of Switched Systems,” *IEEE Control Systems*, Vol. 19, No. 5, pp. 59-70, Oct. 1999.

[LT00] Liu, C-W., J.S. Thorp, “New Methods for Computing Power System Dynamic Response for Real-Time Transient Stability Prediction,” *IEEE Transactions on Circuits and Systems - Part I: Fundamental Theory and Applications,* Vol. 47, No.3, pp. 324-337, March 2000.

[Lue71] Luenberger, D.G., “An Introduction to Observers,” *IEEE Trans. on Automatic Control*, Vol. AC-16, No.6, pp.596-602, Dec. 1971

[MA98] Mayyas, K., T. Aboulnasr, “On Transient Error Surfaces of Output Error IIR Adaptive Filtering,” IEEE Trans. on Signal Processing, Vol. 46, No. 3, pp. 766-771, Mar. 1998.

[Mar00] Marquez, H.J., “An Input-Output Approach to Systems Described Using Multiple Models,” *IEEE Transactions on Circuits and Systems - Part I: Fundamental Theory and Applications*, Vol. 47, No.6, pp. 940-943, June 2000.

[MB99] Mosterman, P.J., G. Biswas, “Diagnosis of Continuous Valued Systems in Transient Operating Regions,” *IEEE Trans. on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 29, No.6, pp554-565, Nov. 1999.

[MGWD97] Musgrave, J.L., T-H. Guo, E. Wong, A. Duyar, “Real-Time Accommodation of Actuator Faults on a Reusable Rocket Engine,” *IEEE Transactions on Control Systems Technology*, Vol. 5, No. 1, pp 100-109, Jan. 1997.

[Mou97] Moustakides, G.V., “Study of the Transient Phase of the Forgetting Factor RLS,” IEEE Trans. on Signal Processing, Vol. 45, No. 10, pp. 2468-2476, Oct. 1997.

[MR76] Mullis, C.T., R.A. Roberts, “Synthesis of minimum roundoff noise fixed point digital filters,” *IEEE Trans. Circuits & Systems*, Vol. CAS-23, pp. 551-562, Sept. 1976.

[NB97] Narendra, K.S., J. Balakrishnan, “Adaptive Control Using Multiple Models,” *IEEE Trans. on Automatic Control*, Vol. 42, No. 2, pp. 171-187, Febr. 1997.

[OI01] O’Brien, R.T., P.A. Iglesias, “On the Poles and Zeros of Linear, Time-Varying Systems,” *IEEE Trans. on Circuits and Systems-I*, Vol. 48, No.5, May 2001, pp. 565-577.

[PK99] Péceli, G., T. Kovácsházy, “Transients in Reconfigurable Digital Signal Processing Systems,” *IEEE Transactions on Instrumentation and Measurement*, Vol. 48, No. 5, pp. 986-989, Oct. 1999.

[PMP96] Padmanabhan, M., K. Martin, and G. Péceli, *Feedback-based Orthogonal Filters: Theory, Applications, and Implementation.* Kluwer Academic Publishers, Boston-Dordrecht-London, 1996. 265 p.

[RG75] Rabiner, L.R., B. Gold, *Theory and Application of Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.

[RM87] Roberts, R.A., C.T. Mullis, *Digital Signal Processing.* New York: Addison-Wesley, 1987.

[S&al93] Sztipanovits, J., D.M. Wilkes, G. Karsai, C. Biegl, L.E. Lynd, “The Multigraph and Structural Adaptivity,” *IEEE Trans. on Signal Processing*, Vol. 41, No. 8, pp. 2695-2716, Aug. 1993.

[Shi88] Shinners, S.M., *Advanced Modern Control System Theory and Design*. John Wiley & Sons, Inc, New York, 1988.

[SK97] Sztipanovits, J., G. Karsai, “Model-Integrated Computing,” *IEEE Computer*, pp. 110-112, April, 1997.

[SKP00] Simon, Gy., T. Kovácsházy, G. Péceli, "Transients in Reconfigurable Control Loops," *IEEE Instrumentation and Measurement Technology Conference, IMTC/2000,* Baltimore, Maryland, USA, May 1-4, 2000, Vol. 3. pp. 1333-1337.

[Solo97] Solo, V., “The Stability of LMS,” IEEE Trans. on Signal Processing, Vol. 45, No. 12, pp. 3017-3026, Dec. 1997.

[SPK01] Simon, Gy., G. Péceli, T. Kovácsházy, "Transient Reduction in Control Loops in Case of Joint Plant-Controller Reconfiguration," *IEEE Instrumentation and Measurement Technology Conference IMTC/2001*, May 21-23, Budapest, Hungary, 2001, Vol. 2, pp. 1172-1176.

[Sun93] Sun, J., “A Modified Model Reference Adaptive Control Scheme for Improved Transient Performance,” IEEE Trans. on Automatic Control, Vol. 38, No. 8, pp. 1255-1259, Aug. 1993.

[TVSS98] Torfs, D.E., R. Vuerinckx, J. Swevers, J. Schoukens, “Comparison of Two Feedforward Design Methods Aiming at Accurate Trajectory Tracking of the End Point of a Flexible Robot Arm,” *IEEE Transactions on Control Systems Technology*, Vol. 6, No. 1, pp 2-14, Jan. 1998.

[VL98] Välimäki, V., T.I. Laakso, “ Suppression of Transients in Variable Recursive Digital Filters with Novel and Efficient Cancellation Method,” *IEEE Trans. on Signal Processing*, Vol. 46, No. 12, pp. 3408-3414, Dec. 1998.

[ZJ99] Zhang, Y., J. Jiang, “Design of Integrated Fault Detection, Diagnosis and Reconfigurable Control Systems,” *Proc. of the 38th Conf. on Decision & Control*, Phoenix, Arizona USA, December 1999, pp. 3587-3592.

[ZZ88] Zetterberg, L.H., Q. Zang, “Elimination of transients in Adaptive Filters with Application to Speech Coding,” *Signal Processing*, Vol.15, No.4, pp. 419-428, 1988.

Table of Contents