The property exploited by real time recurrent learning is that the expression $\partial\hat{y}(k-1)/\partial\underline{\theta}(k-1)$ is the previous model gradient, which has already been evaluated at time $k-1$ and thus is available. Consequently, (17.35) represents a dynamic system for the gradient calculation; the new gradient is a filtered version of the old gradient:

$$\underline{g}(k) = \alpha + \beta\,\underline{g}(k-1)\,. \tag{17.36}$$

Care has to be taken during learning that $|\partial f(\cdot)/\partial\hat{y}(k-1)| < 1$ because otherwise the gradient update becomes unstable. Equivalently to the BPTT algorithm, the model derivatives are equal to zero for the initial values, i.e., $\partial\hat{y}(0)/\partial\underline{\theta}(0) = 0$. For higher order recurrent models and internal dynamics state space models an expression like the second term in (17.35) appears for each model state $\underline{\hat{x}}$. Thus, the dynamic system of the gradient update possesses the same dynamic order as the model.

Compared with BPTT, real time recurrent learning is much simpler and faster, it requires less memory, and most importantly its complexity does not depend on the size N of the training data set. In comparison with static backpropagation, however, it requires significantly higher effort for both implementation and computational demand.

In the case of complex internal dynamics models, the gradient calculation can become so complicated that it is not worth computing the derivatives at all. Alternatively, zero-th order direct search techniques or global optimization schemes as discussed in Sect. 4.3 and Chap. 5, respectively, can be applied because they do not require gradients.

17.6 Multivariable Systems

The extension of nonlinear dynamic models to the multivariable case is straightforward. A process with multiple outputs is commonly described by a set of models, each with a single output; see the Figs. 16.52 and 16.53 in Sect. 9.1. Multiple inputs can be directly incorporated into the model. In internal dynamics approaches and in external dynamics approaches *without* output feedback no restrictions apply. However, for external dynamics structures *with* output feedback the model flexibility is limited. For example, the NOE model for p physical inputs extends to

$$\begin{aligned}\hat{y}(k) = {} & f\,(u_1(k-1),\ldots,u_1(k-m),\ldots,u_p(k-1),\ldots,u_p(k-m),\\ & \hat{y}(k-1),\ldots,\hat{y}(k-m))\,.\end{aligned} \tag{17.37}$$

As demonstrated in the following, this model possesses identical denominator dynamics for the linearized transfer functions of all inputs to the output. The linearized NOE model is

$$\begin{aligned}\Delta\hat{y}(k) = {} & b_1^{(1)}\Delta u_1(k-1) + \ldots + b_m^{(1)}\Delta u_1(k-m) + \ldots\\ & + b_1^{(p)}\Delta u_p(k-1) + \ldots + b_m^{(p)}\Delta u_p(k-m)\end{aligned} \tag{17.38a}$$

$$- a_1\Delta\hat{y}(k-1) - \ldots - a_m\Delta\hat{y}(k-m)\,.$$

In transfer function form this becomes

$$\Delta\hat{y}(k) = \frac{B^{(1)}(q)}{A(q)}\Delta u_1(k) + \ldots + \frac{B^{(p)}(q)}{A(q)}\Delta u_p(k) \tag{17.39}$$

with $A(q) = 1 + a_1q^{-1} + \ldots + a_mq^{-m}$ and $B^{(i)}(q) = b_1^{(i)}q^{-1} + \ldots + b_m^{(i)}q^{-m}$. In opposition to linear dynamic models, where only equation error models are restricted to identical denominator dynamics (Sect. 16.10), for nonlinear dynamics input/output models this is the case for both equation and output error structures. Thus, in order to avoid restrictions in the model dynamics for multivariable systems, the dynamic order m of the model may have to be chosen higher than the true process order; see Sect. 16.10.2. These difficulties do not arise for internal dynamics models and for external dynamics models without output feedback such as NFIR and NOBF because they possess no common output feedback path.

17.7 Excitation Signals

One of the most crucial tasks in system identification is the design of appropriate excitation signals for gathering identification data. This step is even more decisive for nonlinear than for linear models (see Sect. 16.2) because nonlinear dynamic models are significantly more complex and thus the data must contain considerably more information. Consequently, for identification of nonlinear dynamic systems the requirements on a suitable data set are very high. In many practical situations, even if extreme effort and care has been taken, the gathered data may not be informative enough to identify a black box model that is capable of describing the process in all relevant operating conditions. This fact underlines the important role that prior knowledge (and model architectures that allow its incorporation) play in nonlinear system identification.

Independently of the chosen model architecture and structure, the quality of the identification signal determines an upper bound on the accuracy that in the best case can be achieved by the model. For linear systems, guidelines for the design of excitation signals are presented in Sect. 16.2. Quite frequently, the so-called pseudo random binary signal (PRBS) is applied; see e.g., [171, 233, 360]. The parameters of this signal, whose spectrum can be easily derived, are chosen according to the dynamics of the process. For nonlinear systems, however, besides the frequency properties of the excitation signal, the amplitudes have to be chosen properly to cover all operating conditions of interest. Therefore, the synthesis of the excitation signal cannot be carried out as mechanistically as for linear processes (although even for linear systems an individual design can yield significant improvements); each process requires an individual design. Nevertheless, the aspects discussed in

the following should always be considered because they are of quite general interest.

First, it will be illustrated why a PRBS is inappropriate for nonlinear dynamic systems. Consider a first order system of Hammerstein structure whose input lies in the interval $[-4, 4]$ with a time constant of 16 s following the nonlinear difference equation

$$y(k) = 0.06 \arctan(u(k-1)) + 0.94\, y(k-1) \tag{17.40}$$

when sampled with $T_0 = 1$ s. Figure 17.12a shows a PRBS in the time domain and the resulting input space (spanned by $u(k-1)$ and $y(k-1)$)[1] data distribution. Clearly, this data would be well suited for estimating a plane, which is the task for linear system identification. Note that, to estimate the parameters b_1 and a_1 of a linear model $y(k) = b_1 u(k-1) - a_1 y(k-1)$, the data should stretch as widely as possible in the $u(k-1)$ and $y(k-1)$ directions. Such a distribution yields the smallest possible parameter variance in both estimates for b_1 and a_1. Exactly this property is achieved by the PRBS since it alternates between the minimum and maximum value (here -4 and 4) in $u(k-1)$ and also covers the full range for $y(k-1)$ between -1.4 and 1.4. Although a PRBS is well suited for linear system identification, i.e., if the one-step prediction function is known to be a plane, it is inappropriate for nonlinear systems. No information about the system behavior for input amplitudes other than -4 and 4 is gathered.

An obvious solution to this problem is to extend the PRBS to different amplitudes. The arguably simplest approach proposed in [284, 285] is to give each step in the PRBS a different amplitude, leading to an amplitude modulated PRBS (APRBS). First, a standard PRBS is generated. Then, the number of steps are counted and the interval from the minimal to the maximum input is divided into as many levels. Finally, each step in the PRBS is given one of these levels by random. Such an APRBS and the resulting input space data distribution are illustrated in Fig. 17.12b. In general the input space is well covered with data. Some "holes" may exist, however, and their location depends on the random assignment of the amplitude level to the PRBS step. Clearly, here is some room for improvements. Nevertheless, the holes disappear or at least become smaller as the length of the signal increases.

Besides the minimal and maximum amplitudes and the length of the signal (controlled by the number of registers; see [171, 233, 360]) one additional design parameter exists: the minimum hold time, i.e., the shortest period of time for which the signal stays constant. Given the length of the signal, the minimum hold time determines the number of steps in the signal and thus

[1] This analysis is carried out for the external dynamics approach because it allows us to gain some important insights about the desirable properties of the excitation signals. Although with the internal dynamics approach the one-step prediction function is not explicitly approximated, this analysis based on information content considerations is also valid for this class of approaches.

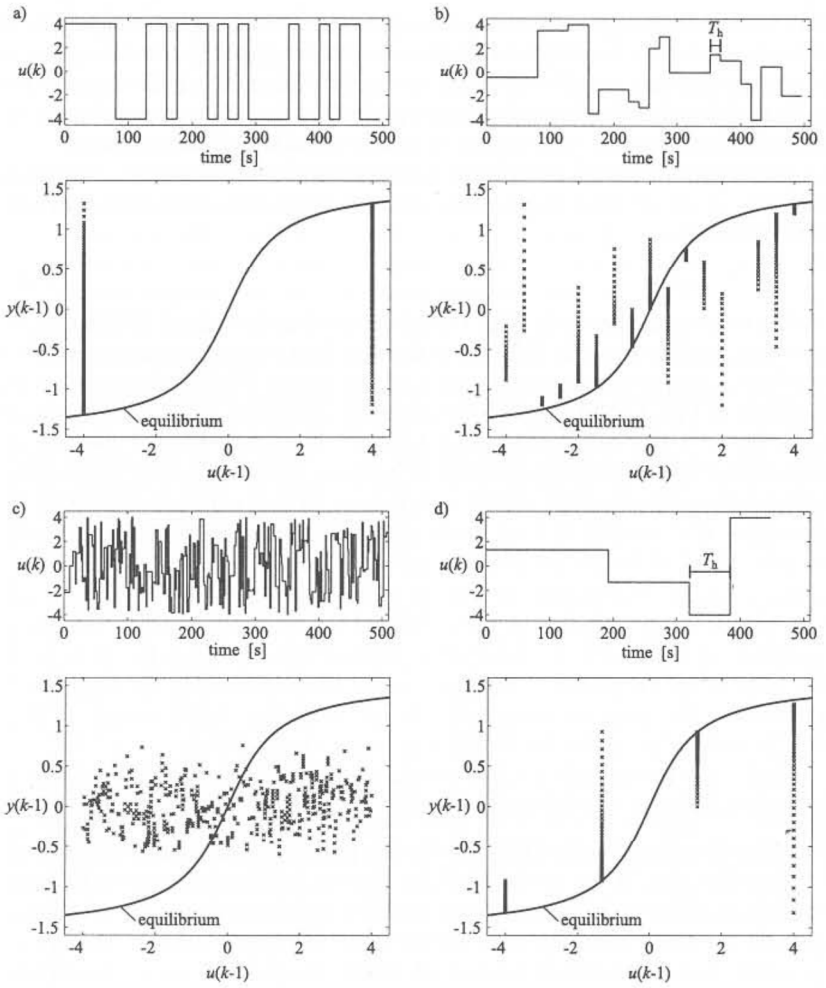

Fig. 17.12. Excitation signals for nonlinear dynamic systems: a) binary PRBS, b) APRBS with appropriate minimum hold time, c) APRBS with too short a minimum hold time, d) APRBS with too large a minimum hold time

it influences the frequency characteristics. In linear system identification the minimum hold time is typically chosen equal to the sampling time [171]. For nonlinear system identification it should be chosen neither too small nor too large. On the one hand, if it is too small the process will have no time to settle and only operating conditions around $y_0 \approx (u_{\max} + u_{\min})/2$ will be covered; see Fig. 17.12c. A model identified from such data would not be able to describe the static process behavior well. On the other hand, if the minimum

hold time is too large only a very few operating points can be covered for a given signal length; see Fig. 17.12d. This would overemphasize low frequencies but, much worse, it would leave large areas of the input space uncovered with data, and thus the model could not properly capture the process behavior in these regions since the data simply contains no information on them.

In the experience of the author it is reasonable to choose the minimum hold time of the APRBS about equal to the dominant (largest) time constant of the process:

$$T_{\mathrm{h}} \approx T_{\max} \,. \tag{17.41}$$

A similar situation as in Fig. 17.12d occurs if multi-valued PRBS signals according to [60, 72, 172, 383] are designed. These signals retain some nice correlation properties similar to those of the binary PRBS. However, as illustrated in Fig. 17.12d, they cover only a small number of amplitude levels if the signal is required to be relatively short. Note that besides the lack of information about the process behavior between these amplitude levels some model architectures are not robust with respect to such a data distribution. In particular, the training of local modeling approaches such as RBF networks or neuro-fuzzy models can easily become ill-conditioned for such data distributions. Global approximators such as polynomials and MLP networks are more robust in this respect but nevertheless they will show a strong tendency to overfitting in the $u(k-i)$-dimensions. From this discussion it becomes clear that besides the properties of the process, the properties of the applied model architecture also play an important role for excitation signal design. In [95] some guidelines are given for local linear neuro-fuzzy models.

In addition to the more general guidelines given above, the following issues influence the design of excitation signals:

1. *Purpose of modeling:* First of all, the purpose of modeling should be specified, e.g., is the model used for control, for fault diagnosis, for prediction, or for optimization. Thereby, the required model precision for the different operating conditions and frequency ranges is determined. For example, a model utilized for control should be most accurate around the crossover frequency, while errors at low frequencies may be attenuated by the integral action of the controller and errors at high frequencies are beyond the actuator and closed-loop bandwidth anyway.
2. *Maximum length of the training data set:* The more training data can be measured the more precise the model will be if a reasonable data distribution is assumed. However, in industrial applications the length of the signal depends on the availability of the process. Usually, the time for configuration experiments is limited. Furthermore, the maximum length of the signal might be given by memory restrictions during signal processing and/or model building.
3. *Characteristics of different input signals:* For each input of the system, it must be checked whether dynamic excitation is necessary (e.g., for the manipulated variables in control systems) or if a static signal is sufficient (e.g., slowly changing measurable disturbances in control systems).
4. *Range of input signals:* The process should be driven through all operating regimes that might occur in real operation of the plant. Unrealistic operating conditions need not be considered. It is important that the data covers the limits of the input range because model extrapolation is much more inaccurate than interpolation.
5. *Equal data distribution:* In particular for control purposes, the data at the process output should be equally distributed in order to contain the same amount of information about each setpoint.
6. *Dynamic properties:* Dynamic signals must be designed in a way that they properly excite variant dynamics in different operating points.

From this list of general ideas, it follows that prior knowledge about the process is required for the design of an identification signal. In fact, some basic properties of the plant to be identified are usually known, namely the static mapping from the system's inputs to the output, at least qualitatively, as well as the major time constants. If the system behavior is completely unknown some experiments such as recording of step responses can provide the desired information.

In practice, the operator of a process restricts the period of time and the kind of measurements that can be taken. Often one will be allowed only to observe the process in normal operation without any possibility of actively gathering information. In this case, extreme care has to be taken to make the system robust against model extrapolation, which is almost unavoidable when available data is so limited. Several strategies for that purpose can be pursued: incorporation of prior knowledge into the model (e.g., a rough first principles model or qualitative knowledge in the form of rules to describe the extrapolation behavior), detection of extrapolation, and switching to a robust backup model, etc.

Finally, it is certainly a good idea to gather more information (i.e., collect more data) in operating regimes that are assumed (i) to behave more complex and/or (ii) to be more relevant than others. The reason for (i) is that the less smooth the behavior is in some region, the more complex the model has to become there and thus the more parameters have to be estimated requiring more data. The reason for (ii) is that more relevant operating conditions should be modeled with higher accuracy than others.

It is important to understand that a high data density in one region forces a flexible model to "spend" a great part of its complexity on describing this region. As this effect is desirable owing to reasons (i) and (ii) it can also be undesirable whenever the high data density was not generated on purpose but just accidentally exists. The latter situation almost always occurs if the data was not actively gathered by exciting the process but rather was observed during normal process operation. Then, rarely occurring operating conditions are under-represented in the data set although they will be given

as much importance as the standard situations. In such a case, the data can be weighted in the loss function in order to force the model to describe these effects accurately and to prevent the model from spending almost all degrees of freedom on the regimes that are densely covered with data; see (2.2) in Sect. 2.3.

17.8 Determination of Dynamic Orders

If the external dynamics approach is taken, the problem of order determination is basically equivalent to the determination of the relevant inputs for the function $f(\cdot)$ in (17.18). Thus, order determination is actually an input selection problem, and the algorithm given below can equally well be applied for input selection of static or dynamic systems. It is important to understand that although the previous inputs $u(k-i)$ and outputs $y(k-i)$ can formally be considered as separate inputs for the function $f(\cdot)$, they possess certain properties that make order determination in fact a much harder problem than the selection of physically distinct inputs. For example, $y(k-1)$ and $y(k-2)$ are typically highly correlated (indicating redundancy) but nevertheless may both be relevant. Up to now no order determination method has been developed that fully takes into account the special properties arising from the external dynamics approach.

The problem of order determination for nonlinear dynamic systems is still not satisfactorily solved. Surprisingly, very little research seems to be devoted to this important area. It is common practice to select the dynamic order of the model by a combination of trial and error and prior knowledge about the process (when available). Some basic observations can support this procedure. Obviously, if oscillatory behavior is observed the process must be at least of second order. Step responses at some operating points can be investigated and linear order determination methods can be applied; see Sect. 16.9. By these means an approximate order determination of the nonlinear process may be possible. This is, however, a tedious procedure, and a reliable automatic data-based determination method would certainly be desirable. In [31, 33] methods based on higher order correlations are proposed. But these approaches are merely model validation tools that require building a model with a specific order first and then indicating which information may be missing.

He and Asada [142] proposed a strategy which is based directly on measurement data and does not make any assumptions about the intended model architecture or structure. It requires only that the process behavior can be described by a smooth function, which is an assumption that has to be made anyway in black box nonlinear system identification. The main idea of this strategy is illustrated in Fig. 17.13a, and is explained in the following. In the general case, the task is to determine the relevant inputs of the function

$$y = f(\varphi_1, \varphi_2, \ldots, \varphi_n) \tag{17.42}$$

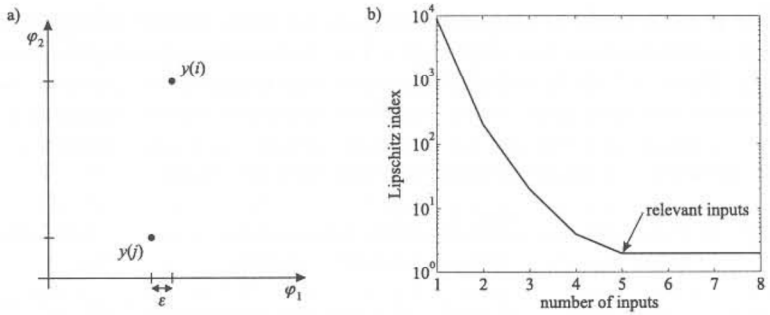

Fig. 17.13. a) Two data points that are close in φ_1 but distant in φ_2 can have very different output values $y(i)$ and $y(j)$ if the function depends on both inputs, while $y(i) \approx y(j)$ if the function only depends on φ_1. b) The Lipschitz indices indicate the case where all relevant inputs are included

from a set of potential inputs $\varphi_1, \varphi_2, \ldots, \varphi_o$ $(o > n)$ that is given. If the φ_i are distinct physical inputs, (17.42) describes a static function approximation problem; if they are delayed inputs and outputs it describes an external dynamics model; see Sect. 17.2.

The idea is as follows. If the function in (17.42) is assumed to depend on only $n-1$ inputs although it actually depends on n inputs, the data set may contain two (or more) points that are very close (in the extreme case they can be identical) in the space spanned by the $n-1$ inputs but differ significantly in the nth input. This situation is shown in Fig. 17.13a for the case $n = 1$. The two points i and j are close in the input space spanned by φ_1 alone but they are distant in the φ_1-φ_2-input space. Because these points are very close in the space spanned by the $n-1$ inputs (φ_1) it can be expected that the associated process outputs $y(i)$ and $y(j)$ are also close (assuming that the function $f(\cdot)$ is smooth). If one (or several) relevant inputs are missing then obviously $y(i)$ and $y(j)$ are expected to take totally different values. In this case, it is possible to conclude that the $n-1$ inputs are not sufficient. Thus, the nth input should be included and the investigation can start again.

In [142] an index is defined based on so-called Lipschitz quotients, which is large if one or several inputs are missing (the larger the more inputs are missing) and is small otherwise. Using this Lipschitz index a curve as shown in Fig. 17.13b may result for $n = 5$ and $o = 8$ when the following input spaces are checked: 1. φ_1, 2. φ_1-φ_2, ..., 8. φ_1-φ_2-...-φ_8. Thus, the correct inputs $(n = 5)$ can be detected at the point where the Lipschitz index stops to decrease.

The Lipschitz quotients in the one-dimensional case (input φ) are defined as

$$l_{ij} = \frac{|y(i) - y(j)|}{|\varphi(i) - \varphi(j)|} \quad \text{for } i = 1, \ldots, N,\ j = 1, \ldots, N \text{ and } i \neq j\,, \tag{17.43}$$