ADAPTIVE OBSERVER AND CONTROL DESIGN FOR A CLASS OF NONLINEAR SYSTEMS

Dipl.–Ing. Martin Rau, Prof. Dr.–Ing. Dr.–Ing. h.c. Dierk Schröder
Institute for Electrical Drive Systems—Technical University Munich
Arcisstraße 21, 80333 München, Germany
Phone: +49-89-289-28445 Fax: +49-89-289-28336
email: Martin.Rau@ei.tum.de, Dierk.Schroeder@ei.tum.de

ABSTRACT: In this paper, we will present a general approach for observer and control design for a class of nonlinear systems. We will assume, that we can separate the system into a known linear part and an unknown static nonlinearity. The observer starts from the known linear part and is extended by a neural network to approximate the nonlinearity. This leads to a mathematically proven stable identification and observation method that provides information about all system states and the nonlinearity. The control concept is a nonlinear state feedback law, which, due to the observer, does not require full state measurement. Our approach has shown excellent results for different types of nonlinearities.

KEYWORDS: nonlinear systems, nonlinear control, adaptive control, nonlinear observer, neural networks

1 INTRODUCTION

Many technical plants in the field of motion control and electrical drive systems are modelled by linear differential equations. Nevertheless, most of these systems contain nonlinearities, which are unknown or partially unknown, e.g. friction, spring characteristics, backlash or excentricities. Any plant containing these nonlinear effects cannot be controlled optimally by a linear controller. Hence, it is desirable to identify the nonlinearity for means of improving the controller design. This should be done online to automate initial identification as well as to cope with parameter drift in the plant.

For linear systems, a state feedback controller allows the most flexible adjustment of the desired dynamics. In real applications, only few components of the state vector are available for measurement and therefore a Luenberger observer for state estimation is necessary. An existing nonlinearity will produce an error in the estimated state vector of the Luenberger observer, since the nonlinear effect is not considered. Furthermore, the nonlinearity was not taken into account during the controller design procedure, which will additionally decrease the performance of the controller or even may cause instability. These facts show the necessity to design observers for nonlinear systems and to consider nonlinear effects in the control law.

2 SYSTEM MODEL AND OBSERVER DESIGN

In motion control and electrical drives, we often have plants consisting of a known linear part and one or more nonlinearities. Here, we will restrict ourselves to SISO–systems (single input single output) with one nonlinearity; MIMO–systems (multiple input multiple output) with several nonlinearities can be studied in Hangl (1997). The systems under consideration are of the following form

$$\begin{aligned} \underline{\dot{x}} &= A\underline{x} + \underline{b}u + \underline{k}\cdot\mathcal{NL}(\underline{x}_E,\underline{p}) \\ y &= \underline{c}^T\underline{x} \end{aligned} \tag{1}$$

Equation (1) describes a system with an **isolated nonlinearity**. The system matrices A, $\underline{b}$, $\underline{c}$ and the coupling vector $\underline{k}$ of the nonlinearity are constant and known, and represent the linear part of the system. $\underline{k}$ contains information about the point of entering of the nonlinearity. The nonlinearity $\mathcal{NL}(\underline{x}_E,\underline{p})$ itself is a static function of its input signal $\underline{x}_E$. $\mathcal{NL}$ is unknown, but for control purpose, we will assume that it is sufficiently smooth to calculate partial derivatives of higher order. The vector $\underline{x}_E$, the input signal of the nonlinearity, is a subset of the whole state vector $\underline{x}$. It has to be known in advance, which state variables the nonlinearity depends on. $\underline{p}$ is an arbitrary time variant, but known parameter vector. By introducing this independent variable it is possible to study the effects of external influences like temperature or age of the plant. Figure 1 shows the structure of a system with an isolated nonlinearity. The observer's task is to identify the nonlinearity $\mathcal{NL}$ by means of a neural network and to provide an estimation of the state vector $\underline{x}$. For online use in control systems, this procedure has to be mathematically proven stable. From control theory's point of view, the presented method is an observation of disturbances, which provides an additional functional relation between the disturbances and their input signals. To approximate the nonlinearity $\mathcal{NL}$, we will use a *General Regression Neural Network* (GRNN) which is a normalized radial basis function (RBF) network.

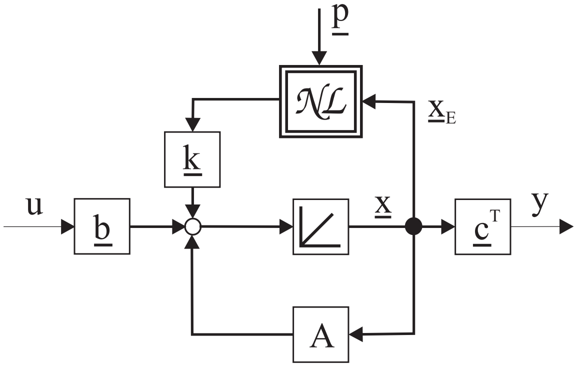

Figure 1: System with isolated nonlinearity

2.1 FUNCTION APPROXIMATION WITH NEURAL NETWORKS

A detailed overview over the variety of neural network types can be seen in Brause (1995). Our intention is only to present the main properties of the GRNN (Specht (1991)). The objective is the approximation of the real nonlinearity $\mathcal{NL}$ by a neural network as a function of its input signal $\underline{x}_E$ and $\underline{p}$ as accurately as possible. The approximation $\widehat{\mathcal{NL}}(\underline{x}_E, \underline{p})$ is the output signal of the neural network. $\mathcal{NL}$ is approximated in the following way. N neurons $\underline{\chi}_i$ with the corresponding weights $\hat{\Theta}_i$ are distributed over the input space of the nonlinearity. The activation vector $\underline{\mathcal{A}}$ is a function of the square of the distance to each neuron. Its components are calculated by the following equation

$$\mathcal{A}_i = \frac{e^{-\frac{(\underline{x}_E-\underline{\chi}_i)^T\cdot(\underline{x}_E-\underline{\chi}_i)}{2\sigma^2}}}{\sum\limits_{l=1}^{N} e^{-\frac{(\underline{x}_E-\underline{\chi}_l)^T\cdot(\underline{x}_E-\underline{\chi}_l)}{2\sigma^2}}} \tag{2}$$

where the smoothing factor σ determines the width of the local activation function. Composing the unknown parameter vector (weights of neural network) $\hat{\underline{\Theta}}^T = \left[\hat{\Theta}_1 \ldots \hat{\Theta}_N\right]^T$, the approximation of the nonlinearity can be expressed as the scalar product of the activation and parameter vector.

$$\widehat{\mathcal{NL}} = \hat{\underline{\Theta}}^T \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) \tag{3}$$

A graphical interpretation of the GRNN is shown in figure 2. The network consists of two main layers, one for the calculation of the activations and one for computing the output signal. The accuracy of the approximation mainly depends on the number and positions of the neurons, but also on the shape of the approximated function. Due to the local activation function of the GRNN, the interpolation and extrapolation properties are definite (figure 3). The real nonlinearity of the plant can be interpreted as a pre–trained network with fixed weights

$$\mathcal{NL} = \underline{\Theta}^T \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) + e_i \tag{4}$$

For all further mathematical considerations, the **inherent approximation error** e_i will be neglected. It can be reduced arbitrarily by increasing the number of neurons.

2.2 ADAPTIVE OBSERVER DESIGN

The observer design is based on the following idea. On the one hand, the Luenberger observer is a well established tool for estimating unknown state variables of systems with known linear dynamics. On the other hand, the

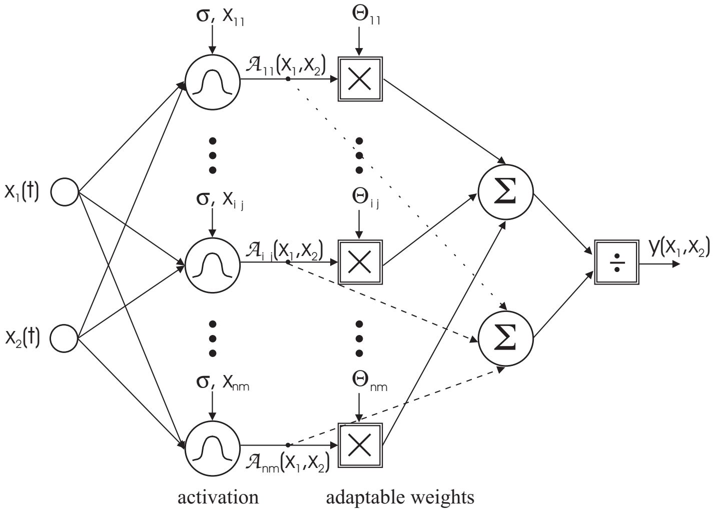

Figure 2: General Regression Neural Network with two–dimensional input

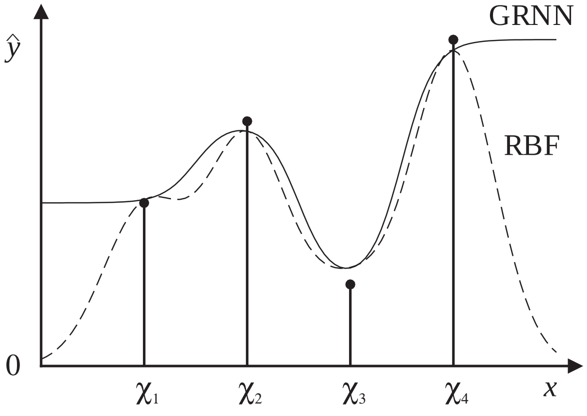

Figure 3: Approximation of a static function by a GRNN and RBF network

GRNN has excellent properties for approximating static nonlinear functions. The combination of both leads to the **neural observer** for systems with an isolated nonlinearity. The plant is described exactly by equation (1). The neural network has to adapt to the plant's nonlinearity $\mathcal{NL}$ mathematically proven stable. To apply the observer design, the following requirements have to be fulfilled (identity matrix I)

1. The nonlinearity is effective at the system's output, such that the transfer function $H_{NL} = \underline{c}^T(sI - A)^{-1}\underline{k} \neq 0$ (except for its zeros).
2. The linear part of the system is completely observable, i.e. the observability matrix $Q_{obs} = \left[\begin{array}{cccc} \underline{c} & A^T\underline{c} & \ldots & (A^T)^{n-1}\underline{c} \end{array}\right]$ has to be regular.
3. The nonlinear plant is either asymptotically stable or a stabilizing controller is known. This controller does not have to be optimized to fulfill certain criterions, but only has to keep the system states bounded in the desired region of operation.

The observer is described in state space notation similarly to an ordinary Luenberger observer.

$$\begin{aligned} \dot{\underline{\hat{x}}} &= A\underline{\hat{x}} + \underline{b}u + \underline{k}\cdot\widehat{\mathcal{NL}}(\underline{x}_E, \underline{p}) + \underline{l}(\hat{y} - y) \\ \hat{y} &= \underline{c}^T\underline{\hat{x}} \end{aligned} \tag{5}$$

The observer error is defined as

$$e = \hat{y} - y \tag{6}$$

With the state error $\underline{e}_S = \underline{\hat{x}} - \underline{x}$ and equations (1) and (5), the error differential equation can be derived as

$$\dot{\underline{e}}_S = (A + \underline{l}\,\underline{c}^T)\underline{e}_S + \underline{k}\left(\widehat{\mathcal{NL}} - \mathcal{NL}\right) \tag{7}$$

Applying equation (7) and using mixed Laplacian and time space notation, the observer error is

$$e = \underline{c}^T \underline{e}_S = \underbrace{\underline{c}^T (sI - A - \underline{l}\,\underline{c}^T)^{-1} \underline{k}}_{H_E} \left(\widehat{\mathcal{NL}} - \mathcal{NL}\right) \tag{8}$$

Equation (8) shows that there is a linear transfer function between the current estimation error $\widehat{\mathcal{NL}} - \mathcal{NL}$ and the observer error e. Depending on the error transfer function H_E we want to derive stable learning laws for the neural network.

The estimation error can be expressed in the form

$$\begin{aligned} \widehat{\mathcal{NL}} - \mathcal{NL} &= \hat{\underline{\Theta}}^T \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) - \underline{\Theta}^T \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) \\ &= \left(\hat{\underline{\Theta}}^T - \underline{\Theta}^T\right) \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) = \underline{\Phi}^T \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) \end{aligned} \tag{9}$$

with the parameter error $\underline{\Phi} = \hat{\underline{\Theta}} - \underline{\Theta}$. Thus, the observer error e has the following final form

$$e = H_E(s) \underline{\Phi}^T \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) \tag{10}$$

If $H_E(s)$ is SPR (strictly positive real), equations (10) and (11) form a globally stable error model (Narendra (1989)). The adaptation law is

$$\dot{\hat{\underline{\Theta}}} = \dot{\underline{\Phi}} = -\eta e \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) \tag{11}$$

With a learning rate $\eta > 0$ and a persistently exciting activation vector $\underline{\mathcal{A}}$ as defined in Narendra (1989), error convergence and parameter convergence are mathematically proven.

$$\lim_{t \to \infty} e(t) = 0 \qquad \lim_{t \to \infty} \hat{\underline{\Theta}} = \underline{\Theta} \qquad \forall \ \eta > 0 \tag{12}$$

The learning law of the GRNN has to manage the general problem of adapting several parameters $\hat{\Theta}_i$ with only one error signal. Compared to a multi–layer perceptron network, the GRNN has local activation functions. That means, for a given input signal only few components of $\underline{\mathcal{A}}$ are activated. Furthermore the activation functions are linearly independent and the condition of persistent excitation according to Narendra (1989) is fulfilled (Lenz (1998)).

If H_E is not SPR, the modified learning law

$$\dot{\hat{\underline{\Theta}}} = \dot{\underline{\Phi}} = -\eta \epsilon H_E \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) \tag{13}$$

with the augmented error ϵ has to be applied. With $\eta > 0$, error and parameter convergence is guaranteed again.

$$\epsilon = e - H_E \hat{\underline{\Theta}}^T \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) + \hat{\underline{\Theta}}^T H_E \underline{\mathcal{A}}(\underline{x}_E, \underline{p}) \tag{14}$$

With an adaptive observer design like this, we succeeded in identifying an isolated nonlinearity and observing all non–measurable states of the system. This observer is similar to a disturbance observer where the nonlinearity is interpreted as disturbance. This approach has the advantage, that the disturbance (nonlinearity) is known at any time after identification.

For the mathematical proof of stability it was necessary to know the input signal $\underline{x}_E$ of the nonlinearity. If $\underline{x}_E$ cannot be measured, the proof is possible with help of **virtual network weights** as it is explained in Lenz (1998) in detail. This is similar to adapting to a time variant nonlinearity without considering time to be an independent input signal of the plant. This observer design is the basis for the following control concept.

3 ADAPTIVE NONLINEAR STATE SPACE CONTROL

In this section we will present an approach which employs the results of the adaptive neural observer. This leads to the advantage that all system states and the nonlinearity are available for the improvement of the controller's performance. This controller will also be adaptive, since the information from the observer is provided during online operation and both, observer and controller are started simultaneously. The controller design is based on a controllable canonical form which has to be examined first.

3.1 NONLINEAR CONTROLLABLE CANONICAL FORM

For linear systems, there are many canonical forms with different properties for special applications. Here, we want to extend the linear controllable canonical form to nonlinear systems. The linear form for SISO systems is described by equation (15), where the parameters α_i and c_i are the coefficients of the corresponding transfer function.

$$\begin{aligned} \dot{\bar{\underline{x}}} &= \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & & & \ddots & & \vdots \\ 0 & \dots & & & 1 & 0 \\ 0 & \dots & & & 0 & 1 \\ -\alpha_0 & -\alpha_1 & \dots & & & -\alpha_{n-1} \end{bmatrix} \cdot \bar{\underline{x}} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \cdot u \\ y &= \begin{bmatrix} c_1 & c_2 & \dots & c_{n-1} \end{bmatrix} \cdot \underline{x} \end{aligned} \tag{15}$$

To transform a system from an arbitrary state space description to the controllable canonical form, a linear transformation $\bar{\underline{x}} = T \cdot \underline{x}$ is necessary. The transformation is possible, if the controllability matrix has full rank. The advantage of this special state space description is that linear state space controllers can be designed easily, e.g. by pole placement.

An extension to nonlinear systems is the nonlinear controllable canonical form (NCCF). Its equations for a system of order n are

$$\begin{aligned} \dot{\bar{\underline{x}}} &= \bar{\underline{f}}(\bar{\underline{x}}) + \bar{\underline{g}}(\bar{\underline{x}}) \cdot u \\ y &= \bar{h}(\bar{\underline{x}}) \end{aligned} \tag{16}$$

with

$$\bar{\underline{f}}(\bar{\underline{x}}) = \begin{bmatrix} \bar{x}_2 \\ \bar{x}_3 \\ \vdots \\ \bar{x}_n \\ \bar{\mathcal{F}}(\bar{\underline{x}}) \end{bmatrix} \qquad \bar{\underline{g}}(\bar{\underline{x}}) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \bar{\mathcal{G}}(\bar{\underline{x}}) \end{bmatrix} \tag{17}$$

The transformation from an arbitrary nonlinear state space equation with state vector $\underline{x}$ can be achieved by a nonlinear state transformation which has to be a **diffeomorphism in the whole range of operation** of the system.

$$\bar{\underline{x}} = \underline{v}(\underline{x}) \qquad \underline{x} = \underline{v}^{-1}(\bar{\underline{x}}) \tag{18}$$

The nonlinear system in its original coordinates is given by the functions $\underline{f}(\underline{x})$, $\underline{g}(\underline{x})$ and $h(\underline{x})$.

$$\begin{aligned} \dot{\underline{x}} &= \underline{f}(\underline{x}) + \underline{g}(\underline{x}) \cdot u \\ y &= h(\underline{x}) \end{aligned} \tag{19}$$

The transformation $\underline{v}$ is defined by the conditions in equation (20).

$$\bar{\underline{f}}(\bar{\underline{x}}) = \underline{f}\left(\underline{v}^{-1}(\bar{\underline{x}})\right) \quad \text{and} \quad \bar{\underline{g}}(\bar{\underline{x}}) = \underline{g}\left(\underline{v}^{-1}(\bar{\underline{x}})\right) \tag{20}$$

The existence of $\underline{v}$ is not guaranteed for all nonlinear systems. Nonlinear controllability conditions, as proven in Slotine (1991), have to be fulfilled. They include Lie–brackets and the involutivity condition. The transformation $\underline{v}$ exists only if the following conditions hold in the operating range Ω of the system (Slotine (1991)):

- the vector fields $\left\{\underline{g}, ad_f\underline{g}, \dots, ad_f^{n-1}\underline{g}\right\}$ are linearly independent in Ω
- the set $\left\{\underline{g}, ad_f\underline{g}, \dots, ad_f^{n-1}\underline{g}\right\}$ is involutive in Ω

If these controllability conditions hold, then an output function $h_t(\underline{x})$ exists, such that the system with this output function has relative degree n. Note, that h_t is not a new system output, but is only required for the calculation of $\underline{v}$. Of course, all state variables must be accessible to perform the transformation. How to compute the necessary output function h_t is quite lengthy and will not be shown here. For detailed information see Slotine (1991) and Sommer (1979). Nevertheless, h_t can be calculated symbolically with standard algebra software (e.g. MAPLE V) and is assumed to be known for the future considerations.

Thus, the system under consideration is of the new form

$$\begin{aligned} \dot{\underline{x}} &= \underline{f}(\underline{x}) + \underline{g}(\underline{x}) \cdot u \\ y_t &= h_t(\underline{x}) \end{aligned} \tag{21}$$

Input–output linearization of this system always results in a relative degree of n, and therefore is equal to input–state linearization. This fact guarantees, that there is **no zero dynamics** and no examinations concerning their stability are necessary. The output's derivatives are

$$\begin{aligned} y_t &= h_t(\underline{x}) \\ \dot{y}_t &= L_f h_t(\underline{x}) \\ \ddot{y}_t &= L_f^2 h_t(\underline{x}) \\ &\vdots \\ y_t^{n-1} &= L_f^{n-1} h_t(\underline{x}) \\ y_t^{n} &= L_f^{n} h_t(\underline{x}) + L_g L_f^{n-1} h_t(\underline{x}) \cdot u \end{aligned} \tag{22}$$

$$\tag{23}$$

These equations directly lead to the transformed state vector $\underline{\bar{x}}$ and the diffeomorphism $\underline{v}(\underline{x})$. The new state vector and the transformation $\underline{v}$ are defined in equation (24).

$$\underline{\bar{x}} = \begin{bmatrix} y_t \\ \dot{y}_t \\ \vdots \\ y_t^{n-1} \end{bmatrix} \qquad \underline{v} = \begin{bmatrix} h_t(\underline{x}) \\ L_f h_t(\underline{x}) \\ \vdots \\ L_f^{n-2} h_t(\underline{x}) \\ L_f^{n-1} h_t(\underline{x}) \end{bmatrix} \tag{24}$$

The functions $\bar{\mathcal{F}}(\underline{\bar{x}})$ and $\bar{\mathcal{G}}(\underline{\bar{x}})$ in equation (17) therefore are

$$\bar{\mathcal{F}}(\underline{\bar{x}}) = L_f^n h_t \left(\underline{v}^{-1}(\underline{\bar{x}})\right) \qquad \bar{\mathcal{G}}(\underline{\bar{x}}) = L_g L_f^{n-1} h_t \left(\underline{v}^{-1}(\underline{\bar{x}})\right) \tag{25}$$

or respectively in the system's original coordinates

$$\mathcal{F}(\underline{x}) = \bar{\mathcal{F}}\left(\underline{v}(\underline{x})\right) = L_f^n h_t \left(\underline{x}\right) \qquad \mathcal{G}(\underline{x}) = \bar{\mathcal{G}}\left(\underline{v}(\underline{x})\right) = L_g L_f^{n-1} h_t \left(\underline{x}\right) \tag{26}$$

The transformation of the system into state coordinates $\underline{\bar{x}}$ is complete now. The system shows the same behaviour as before, but has the controllable canonical form, which offers easier controller design. For all later considerations we assume that the system's equations are in NCCF already. The corresponding real output function consequently is

$$y = h(\underline{x}) = h\left(\underline{v}^{-1}(\underline{\bar{x}})\right) = \bar{h}(\underline{\bar{x}}) \tag{27}$$

The nonlinear state space controller design will be applicable like pole placement for linear systems. We want to specify a linear reference system, which has the desired pole locations. The nonlinear system will be input–state linearized with the same poles. The chosen control law (28) allows pole placement for the nonlinear system, i.e. its dynamics can fully be influenced.

$$u = -\frac{1}{\bar{\mathcal{G}}(\underline{\bar{x}})} \left(\bar{\mathcal{F}}(\underline{\bar{x}}) + \alpha_0 \bar{x}_1 + \alpha_1 \bar{x}_2 + \ldots + \alpha_{n-1} \bar{x}_n - \gamma w\right) = r(\underline{\bar{x}}) + m(\underline{\bar{x}}) \cdot w \tag{28}$$

By applying the transformation $\underline{\bar{x}} = \underline{v}(\underline{x})$, the control law can also be written in the system's original coordinates. The resulting linear system has the linear controllable canonical form with the coefficients α_i in the last row of the system matrix and the coupling vector $\underline{\bar{b}} = [0 \; 0 \; \ldots \; 0 \; \gamma]^T$. The dynamics is adjusted via α_i, stationary accuracy is achieved by the choice of an appropriate γ. The control goal is to make y equal to w for a constant reference value w. For $\dot{\underline{\bar{x}}} = 0$, the last row of the controlled system is

$$0 = -\alpha_0 y + \gamma w \tag{29}$$

To meet $y = w$ in the stationary point, $\gamma = \alpha_0$ has to be chosen.

The proposed control law can be applied to arbitrary nonlinear systems of the form (16), but of course also to systems with isolated nonlinearity, which are included in equation (16). The control law (28) is not simplified significantly, but it is applicable to a wider class of real applications now. As we have seen, we need the whole state vector for free pole placement. For systems with an isolated nonlinearity the adaptive observer of section 2 can be used to estimate the system's states. When calculating the control law symbolically for such a system, partial derivatives of the unknown nonlinear function $\mathcal{NL}$ of order up to the system's order may appear. The controller is of the general form

$$u = r\left(\underline{x}, \mathcal{NL}, \frac{d\mathcal{NL}}{dx_E}, \ldots\right) + m\left(\underline{x}, \mathcal{NL}, \frac{d\mathcal{NL}}{dx_E}, \ldots\right) \cdot w \tag{30}$$

The derivatives are also unknown and therefore have to be identified as well as the nonlinearity itself.

3.2 ADAPTIVE CONTROL LAW FOR SYSTEMS WITH ISOLATED NONLINEARITY

The next step of the adaptive controller design is the online identification of derivatives of the nonlinear function included in the plant with respect to its input signal x_E. In general, the nonlinearity may depend on several state variables. To provide a clear approach, we will show the identification procedure for a one–dimensional function. Nevertheless, an extension to multi–dimensional functions is possible. Additionally, we assume that the adaptive observer is well designed, so that its output equals the nonlinearity at any time. The identification scheme is based on the chain rule of differential calculus

$$\frac{d\mathcal{NL}(x_E)}{dt} = \frac{\partial \mathcal{NL}}{\partial x_E} \cdot \frac{dx_E}{dt} = \mathcal{NL}' \cdot \dot{x}_E \tag{31}$$

This equation is implemented in an identification structure (figure 4) where $\mathcal{NL}'$ is replaced by a neural network (GRNN). The extension by an integrator avoids explicit differentiation of the output signal. Instead of differentiating x_E, the signal $\dot{x}_E$ can also be composed of the states and the nonlinearity of the system's observer. The calculation of the error signal e is similar to section 2, so we skip the details. The result is

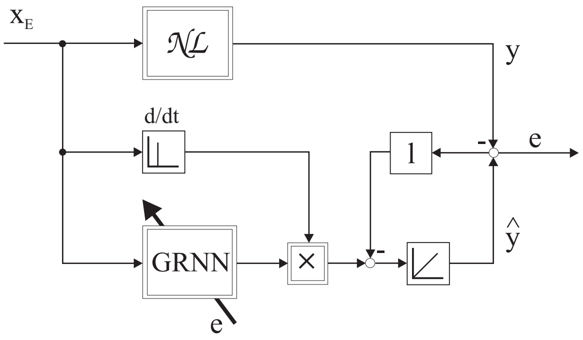

Figure 4: Identificator for the partial derivative of $\mathcal{NL}$

$$e = H_d(s) \cdot \dot{x}_E \cdot \underline{\mathcal{A}}(x_E) \cdot \underline{\Phi} \qquad \text{with} \qquad H_d(s) = \frac{1}{s+l} \tag{32}$$

$\underline{\Phi}$ is the parameter error vector of the corresponding neural network. The error transfer function $H_d(s)$ is always SPR, so the simplified learning law for the network weights $\hat{\underline{\Theta}}$ guarantees global stability for all $\eta, l > 0$.

$$\dot{\underline{\Phi}} = \dot{\hat{\underline{\Theta}}} = -\eta e \underline{\mathcal{A}}(x_E)\dot{x}_E \tag{33}$$

For derivatives of higher order, this procedure has to be repeated in the same manner. To achieve convergent results, the nonlinearity has to be differentiable to the desired order. It should be mentioned, that the identification of the nonlinear function and of its derivative with respect to its input signal can operate simultaneously, which means all identificators can be active at the same time.

With this step, all components for the control law for systems with isolated nonlinearity are available. Note, that this control concept assumes an unknown nonlinearity and therefore differs from ordinary nonlinear control rules significantly. The control concept in equation (30) requires

- information about the nonlinearity and its derivatives at any time (the order of the derivatives is specified by calculating the control law symbolically) which is provided by the adaptive observer
- all state variables of the system which are also provided by the adaptive observer
- a symbolically calculated control law with specified linear dynamics (α_i)

This control method is a straightforward extension of the linear state space control theory. If the nonlinearity is zero, the controller (28) results in a linear state feedback law, as if the nonlinearity were not present, which can be verified for any system by symbolic calculation. This result leads to the following important property: As long as all neural networks are untrained (e.g. at the beginning of operation) their output signals are zero and the controller is a linear state feedback law. If this controller can stabilize the system (well defined dynamics is not required), no pre–training of the networks is necessary and operation can start without prior knowledge. This is important especially for unstable systems, where uncontrolled excitation of the system is not possible. The structure of the presented concept is depicted in figure 5 for the case, that derivatives of order two of the nonlinear function are necessary. The adaptive component of the controller results from the adaptation of the observer (and the identificators for the derivatives) to the unknown nonlinearity, which, in turn provides this information to the control law. To show the practical benefit, we will present an application example in the next section.

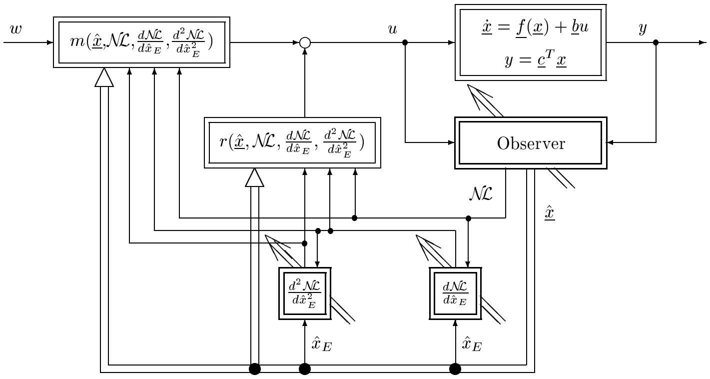

Figure 5: Nonlinear adaptive state space control

4 APPLICATION EXAMPLE

Many subsystems in electro–mechanical drives can be modelled as multi–body systems. Our sample system is a two–body system coupled by a damped elastic spring. The first body is the rotor of a motor, the second body includes all rotating parts of the load. The objective of the controller design is the speed control of the second mass. As a nonlinearity, we use a speed dependent torque at the second body, e.g. a friction torque. In this example, the nonlinearity would lead to an unstable system, so we can show the advantage of starting without prior knowledge. The system's state space description is

$$\begin{aligned} \underline{\dot{x}} &= \underbrace{\begin{bmatrix} -\frac{d}{J_1} & -\frac{c}{J_1} & \frac{d}{J_1} \\ 1 & 0 & -1 \\ \frac{d}{J_2} & \frac{c}{J_2} & -\frac{d}{J_2} \end{bmatrix}}_{A} \cdot \underline{x} + \underbrace{\begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix}}_{\underline{b}} \cdot u + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \frac{1}{J_2} \end{bmatrix}}_{\underline{k}} \cdot \underbrace{\frac{x_3^3}{100}}_{\mathcal{NL}(x_3)} \\ y &= \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{\underline{c}^T} \cdot \underline{x} \end{aligned} \tag{34}$$

with the state vector

$$\underline{x} = \begin{bmatrix} N_1 \\ \Delta\alpha \\ N_2 \end{bmatrix} \tag{35}$$

where N_1 is the speed of the motor, N_2 the speed of the load and $\Delta\alpha$ the torsion angle between the two bodies. The input signal u is the active motor torque (torque at the first body). The control law was calculated according to the previous sections and resulted in a controller depending on all state variables, the nonlinearity and the first two derivatives of the nonlinear function.

To check whether the controller could be started without prior knowledge, a linear state space controller was designed without taking into account the nonlinearity. Nevertheless, it was able to stabilize the system. Therefore, the neural networks were all initialized to zero and the controller was started without prior knowledge about the nonlinearity. The learning phase is depicted in figure 6. At the beginning of operation great deviations from the reference value are obvious. These result from the untrained networks. With increasing time, these differences disappear. The identified nonlinearity is shown in figure 7. Its derivatives were also learned by neural networks.

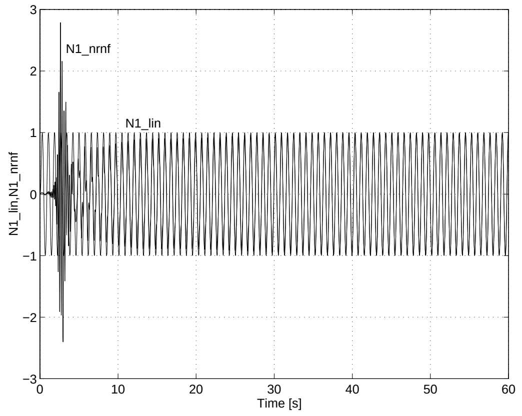

Figure 6: Learning phase

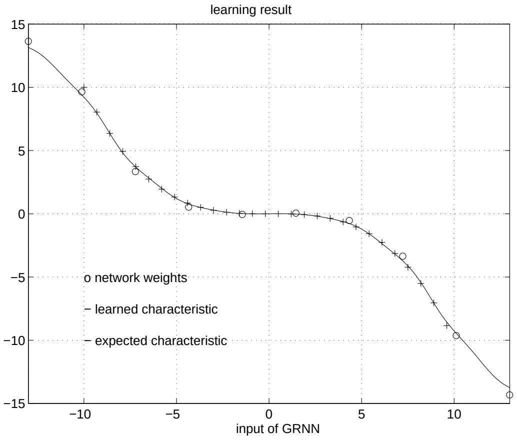

Figure 7: Identified nonlinearity

Figure 8 shows a comparison between different step responses.

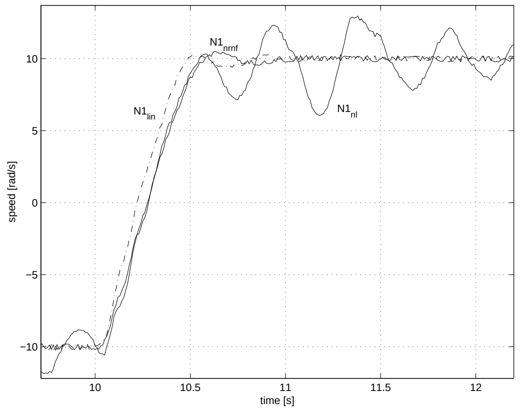

Figure 8: Comparison of step responses

- $N1_{nl}$: nonlinear system is controlled by a linear state space controller without considering the nonlinearity in the controller.
- $N1_{nrnf}$: nonlinear system is controlled by the proposed nonlinear state space controller.
- $N1_{lin}$: linear reference system without nonlinearity

It can be seen, that the linear controller's performance is not satisfying, whereas the proposed controller is able to compensate the nonlinearity completely. Its dynamics is identical to the reference system which was the aim of the design procedure. A similar controller has already been implemented in continuous processing plants for the compensation of the effects of unwinder excentricities.

5 CONCLUSION

In this paper we presented a method for identification and control of a special class of nonlinear systems. These systems with an isolated nonlinearity often appear in electromechanical drive systems. The plant is separated into a known linear part and an unknown nonlinear part, the isolated nonlinearity. The nonlinear part is identified globally stable by a *General Regression Neural Network*, a special kind of RBF network. The controller relies on the output of the adaptive observer which provides estimates of all system states and information about the nonlinearity. Thus, full state measurement is not required. To perform the proposed design procedure, the system under consideration must be input–state linearizable. The controller compensates all nonlinear effects in the plant and allows pole placement of the remaining linearized system. The benefit of this controller is that it takes into account the nonlinearity already during the design procedure and does not treat the nonlinearity as a disturbance.

REFERENCES

Brause, R.: *Neuronale Netze.* Teubner Verlag Stuttgart, 1995.

Hangl, F. D.: *Theorie des systematischen Entwurfs lernfähiger Beobachter für eine Klasse nichtlinearer Strecken.* GMA–Workshop Interlaken, 1997.

Lenz, U.: *Lernfähige neuronale Beobachter für eine Klasse nichtlinearer dynamischer Systeme und ihre Anwendung zur intelligenten Regelung von Verbrennungsmotoren.* Dissertation, TU München, 1998.

Narendra, K. S.: *Stable Adaptive Systems.* Prentice–Hall, 1989.

Slotine, J., Weiping, L.: *Applied Nonlinear Control.* Prentice–Hall, 1991.

Sommer, R.: *Entwurf nichtlinearer, zeitvarianter Systeme durch Polvorgabe.* Regelungstechnik Vol. 12, Oldenbourg Verlag, 1979, pp. 393–399.

Specht, D.: *A General Regression Neural Network.* IEEE Transactions on Neural Networks, Vol. 2, 6. November 1991, pp. 568–576.