New Concepts for the Identification of Dynamic Takagi-Sugeno Fuzzy Models

Christoph Hametner
Institute for Mechanics and Mechatronics,
Vienna University of Technology, Vienna, Austria
hametner@impa.tuwien.ac.at

Stefan Jakubek
Institute for Mechanics and Mechatronics,
Vienna University of Technology, Vienna, Austria
jakubek@impa.tuwien.ac.at

***Abstract*— Takagi-Sugeno Fuzzy Models have proved to be a powerful tool for the identification of nonlinear dynamic systems. Recent publications have addressed the problems of local versus global accuracy and the identifiability and interpretability of local models as true linearisations. The latter issue particularly concerns off-equilibrium models. Well-established solution approaches involve techniques like regularisation and multi-objective optimisation. In view of a practical application of these models by inexperienced users this paper addresses the following issues: 1. Unbiased estimation of local model parameters in the presence of input- and output noise. At the same time the dominance of the trend term in off-equilibrium models is balanced. 2. The concept of stationary constraints is introduced. They help to significantly improve the accuracy of equilibrium models during steady-state phases. A simulation model demonstrates the capabilities of the proposed concepts.**

***Keywords*—Takagi-Sugeno Fuzzy Models, Nonlinear System Identification, Identification Algorithms**

I. Introduction

Due to the high complexity of many real systems physical models are only rarely applied. Their high computational demand in connection with the difficulties involved in determining physical parameters often prevent a practical application. In this case the architecture of the local Neuro-Fuzzy networks, also known as *Takaki-Sugeno* models has proved to be reliable. These models interpolate between different local models each valid in a certain region of a partition space. The basic principles of this modeling approach have been more or less independently developed in different disciplines like neural networks, fuzzy logic, statistics and artificial intelligence with different names such as local model networks, Takagi-Sugeno fuzzy models or neuro-fuzzy models, [1], [2], [3], [4], [5], [6], [7].

An important advantage of local model networks is that they can be designed to automatically adapt themselves to the complexity of the problem in a highly efficient way. This is achieved by partitioning those regions of the input space where the structure of the nonlinearity is more complex into many subdomains, [7], [8], [9]. Many developments are focused on the bottleneck of the local model network which is the determination of these subdomains or validity functions, respectively.

It is a well known fact that the identification of dynamic nonlinear Takaki-Sugeno models is very different from their static counterparts. Local linear models are often poorly identifiable as linearisations of the nonlinear process. This is especially the case when the assumptions on persistence of excitation or identifiability are not strongly satisfied. The use of constraints and regularisation are useful to identify interpretable local models, [10]. For dynamic systems, this is typically the case for local models associated with transient operating regimes, which are commonly called *off-equilibrium models*, [8]. For these models the so-called *trend-term* tends to dominate over the other model parameters.

An important problem involved with the identification of dynamic models from real processes is the noise in input- *and* output data. Conventional methods for the minimisation of the prediction error suffer from the drawback that the parameter estimates are biased. This problem becomes particularly serious in situations where the noise level by itself is strong, when there is a significant difference in noise corruption between inputs and output and when there is a strong correlation between input- and output noise.

An alternative that is also applicable to Neuro-Fuzzy networks is the *Total Least Squares* approach (TLS), [11]. In section III this approach will be explained and the reduction of the dominance of the trend term associated with this approach will be highlighted. Also, two important modifications for a practical applicability will be made: On the one hand a data transformation is proposed that takes into account the different noise variances and possible correlations. On the other hand one has to consider that the identification data have to be weighted with respect to their validity to the local model for parameter estimation. Therefore, in analogy to the Weighted Least Squares method a *Weighted Total Least Squares* method will be derived.

The second fundamental problem of dynamic models to be addressed in this paper is their accuracy during steady-state phases. A common way to improve steady-state behaviour is the introduction of stationary phases where inputs and outputs are kept constant, [12] and increased weighting of these phases. This approach again yields a multi-objective parameter estimation as described in e.g. [10] where dynamic and static accuracy have to be traded for.

An alternative and very efficient strategy consists of *enforcing* the compliance to selected stationary operating points besides minimising a performance criterion. This approach

1–4244–0023–6/06/$20.00

ensures a certain steady-state accuracy without the aforementioned tradeoff. Mathematically, this strategy leads away from pure optimisation towards constrained optimisation. Section IV describes how stationary constraints can be enforced in a dynamic Neuro-Fuzzy network.

Altogether the application of the proposed concepts leads to a training algorithm that produces excellent dynamic models without the expense of in-depth mathematical knowledge about regularisation and multi-objective optimisation.

Section V illustrates the effectiveness by means of a simulation example.

II. Architecture of dynamic Neuro-Fuzzy networks

A. General

Fig. 1 shows the architecture of a dynamic Neuro-Fuzzy network. Every local model (indicated as LM_j) maps past inputs and outputs in $\boldsymbol{u}$ according to

$$\boldsymbol{u} = [\boldsymbol{u}_1(k)\ ,\ \boldsymbol{u}_2(k)\ ,\ \ldots\ ,\ \boldsymbol{u}_r(k)\ ,\ \hat{\boldsymbol{y}}(k)]^T \tag{1}$$

to a local estimation $\hat{y}_j$ of $y(k)$. Here $\boldsymbol{u}_i(k)$ contains past values of the i-th input according to

$$\boldsymbol{u}_i(k) = [u_i(k-d_i-1),\ u_i(k-d_i-2)\ ,\ \ldots$$
$$\ldots\ ,\ u_i(k-d_i-m_i)]$$

and $\hat{\boldsymbol{y}}(k)$ contains past network outputs:

$$\hat{\boldsymbol{y}}(k) = [\hat{y}(k-1)\ ,\ \hat{y}(k-2)\ ,\ \ldots\ ,\ \hat{y}(k-n)].$$

In the above equations m_i $(i = 1, \ldots, r)$ denotes the order of the numerator of the i-th input, d_i is the associated dead time and n is the denominator order. Using the validity function Φ_j all local estimations $\hat{y}_j$ are used to form the global model output $\hat{y}(k)$ by weighted aggregation:

$$\hat{y}(k) = \sum_{j=1}^{m} \Phi_j(\boldsymbol{u}_\Phi)\hat{y}_j(\boldsymbol{u}, \boldsymbol{\theta}_j). \tag{2}$$

Here $\boldsymbol{\theta}_j$ is a vector containing the parameters of the j-th local model. Typically, a *local linear* model structure is implemented:

$$\hat{y}_j(\boldsymbol{u}, \boldsymbol{\theta}_j) = \boldsymbol{u}^T \cdot \boldsymbol{\theta}_j \tag{3}$$

In addition to the terms in $\boldsymbol{u}$ one often adds a constant offset or so-called *trend-term* which yields a *local linear affine* model, cf. [8]:

$$\hat{y}_j = [\ \boldsymbol{u}^T\ ,\ 1\ ] \cdot \boldsymbol{\theta}_j. \tag{4}$$

The local model structures (3) and (4) have the advantage that they can be analysed with the vast fundus of linear system theory. Many model-based applications of automation theory like controller design or fault-diagnosis also profit from a linear system structure.

From Fig. 1 and equation (2) it becomes apparent that the local models and their validity functions have different arguments. While the input to local models (also known as consequents) contains all dynamic inputs and outputs given in (1) the input vector to the validity functions $\boldsymbol{u}_\Phi$ (also known as premises) contains only those quantities that are necessary for proper partitioning. It should be noted that proper selection of the partition space can be very different for static and dynamic models, [13], [8].

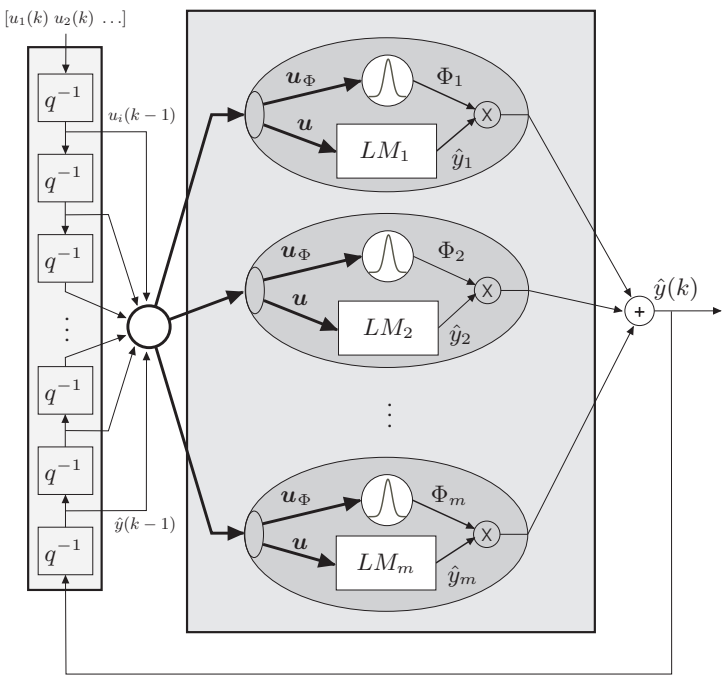

Fig. 1. Architecture of a dynamic Neuro-Fuzzy network

The spatial distribution of local models in the partition space via their validity functions $\Phi_j(\boldsymbol{u}_\Phi)$ must be designed such that they provide sufficient accuracy.

While in many special applications their location is fixed, general local model network construction algorithms consist of an outer loop that determines the location, extent and orientation of each local model by means of its validity function Φ_j and a nested inner loop that optimises the parameters $\boldsymbol{\theta}_j$ of the local model. Recent developments can be found in e.g. [7], [14].

III. Total Least Squares

A. General

The estimation of the local model parameters $\boldsymbol{\theta}_j$ in (2) by Least-Squares is based on the minimisation of the prediction error at the training data:

$$J = \frac{1}{2N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2. \tag{5}$$

Here N denotes the number of data records used for the local model. In the case that only target data are affected by noise the minimisation of (5) yields a bias-free estimation of $\boldsymbol{\theta}_j$. In dynamic identification problems, however, the regressors in (1) contain past inputs and output of the process which means that

noise affects both the target $y(k)$ and the regressor itself. It is well-known that the resulting statistical correlation between target and regressor causes a bias in the parameter estimates.

In order to obtain a bias-free parameter estimation in the case of noisy inputs and outputs it is necessary to reconstruct both outputs *and* inputs. For a simple map $x \mapsto y$ this means that instead of (5) the following criterion has to be minimised:

$$J = \frac{1}{2N} \left\{ \sum_{i=1}^{N} (x_i - \hat{x}_i)^2 + \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \right\} \tag{6}$$

Since (6) entails that both inputs and outputs have to be reconstructed the underlying optimisation is called *Total Least Squares* (TLS). From a geometric point of view the optimisation of (6) requires that the euclidean distances between data points (x_i, y_i) and the model $(\hat{x}_i, \hat{y}_i)$ are minimised. It can be shown that for $N \to \infty$ TLS delivers bias-free parameter estimates, cf. [15].

B. Methodology

The application of Total Least Squares for parameter estimation from noisy inputs and outputs has been suggested repeatedly in recent years, [11], [15], [16], [17]. Implicitly, the Gustafson-Kessel-algorithm [18] generates clusters of local linear models with TLS parameter estimates. However, this method is confined to the product space which severely limits its applicability to high-dimensional problems.

This section provides an exemplary explanation of Total Least-Squares for a dynamic SISO-system, followed by sec. III-C where the case is treated when inputs and outputs are subject to different noise variances or, more generally, when there are even statistical correlations between the different noise signals.

Let $\boldsymbol{X} \in \mathbb{R}^{N \times M}$ be the regressor matrix and $\boldsymbol{y} \in \mathbb{R}^{N \times 1}$ be the target vector in a SISO identification problem. The row vectors $\boldsymbol{x}^T$ in $\boldsymbol{X}$ and the elements of $\boldsymbol{y}$ are built according to the following scheme:

$$\left.\begin{array}{rcc} \boldsymbol{x}^T(k) & = & [\, y(k-1)\,,\, y(k-2)\,,\, \ldots \mid \\ & & \mid u(k-d-1)\,,\, u(k-d-2)\,,\, \ldots] \\ \boldsymbol{y}(k) & = & y(k) \end{array}\right\}. \tag{7}$$

Total Least Squares now aims at modifying both $\boldsymbol{y}$ and $\boldsymbol{X}$ in such a way that the following conditions are satisfied:

$$\begin{gathered} \boldsymbol{y}, \boldsymbol{X} \;\; \Rightarrow \;\; \hat{\boldsymbol{y}}, \hat{\boldsymbol{X}}, \;\; \text{where} \\ \hat{\boldsymbol{y}} \in \text{Image}(\hat{\boldsymbol{X}}) \;\; \text{and} \;\; || \, \boldsymbol{y} - \hat{\boldsymbol{y}} \mid \boldsymbol{X} - \hat{\boldsymbol{X}} \, ||_F = \min . \end{gathered} \tag{8}$$

The reconstructions $\hat{\boldsymbol{y}}, \hat{\boldsymbol{X}}$ can be arbitrary, however in the present application a *linear* model structure was chosen. For that purpose an augmented regressor matrix is defined:

$$\boldsymbol{W} = [\, \boldsymbol{y} \mid \boldsymbol{X} \,] \tag{9}$$

The k-th row vector $\boldsymbol{w}^T(k)$ from $\boldsymbol{W}$ is given as $\boldsymbol{w}^T(k) = [\, \boldsymbol{y}(k) \mid \boldsymbol{x}^T(k) \,]$. A linear reconstruction of $\boldsymbol{w}^T$ is obtained from

$$\hat{\boldsymbol{w}}^T = \boldsymbol{w}^T - \left[(\boldsymbol{w} - \boldsymbol{m})^T \boldsymbol{b} \right] \boldsymbol{b}^T \;\; \text{with} \; ||\boldsymbol{b}||_2 = 1 \tag{10}$$

Here $\boldsymbol{b}$ denotes the unit normal vector to the optimal (hyper)plane and $\boldsymbol{m}$ is any vector in that (hyper)plane. This linear model structure also ensures that $\hat{\boldsymbol{y}} \in \text{Image}(\hat{\boldsymbol{X}})$ holds. A reconstruction of all N data records yields

$$\hat{\boldsymbol{W}} = \boldsymbol{W} - \left[(\boldsymbol{W} - \boldsymbol{1}\boldsymbol{m}^T)\boldsymbol{b} \right] \boldsymbol{b}^T \tag{11}$$

with the $N \times 1$-vector $\boldsymbol{1} = [\, 1\,,\, 1\,,\, \ldots\,,\, 1]^T$. The Frobenius norm becomes

$$||\boldsymbol{W} - \hat{\boldsymbol{W}}||_F = \boldsymbol{b}^T (\boldsymbol{W}^T - \boldsymbol{m}\boldsymbol{1}^T)(\boldsymbol{W} - \boldsymbol{1}\boldsymbol{m}^T)\boldsymbol{b}. \tag{12}$$

For the determination of $\boldsymbol{b}$ and $\boldsymbol{m}$ the centroid vector $\boldsymbol{\mu}_W$ of all data records is defined:

$$\boldsymbol{\mu}_W = \frac{1}{N} \boldsymbol{W}^T \boldsymbol{1} \tag{13}$$

Referencing all data records to the centroid according to $\tilde{\boldsymbol{W}} = \boldsymbol{W} - \boldsymbol{1}\boldsymbol{\mu}_W^T$ leads to

$$\begin{aligned} ||\boldsymbol{W} - \hat{\boldsymbol{W}}||_F &= \boldsymbol{b}^T (\tilde{\boldsymbol{W}}^T \tilde{\boldsymbol{W}}) \boldsymbol{b} + \\ &+ N[(\boldsymbol{\mu}_W - \boldsymbol{m})^T \boldsymbol{b}]^2 = \min . \end{aligned} \tag{14}$$

The optimal choice for $\boldsymbol{m}$ is apparently $\boldsymbol{m} = \boldsymbol{\mu}_W$. The matrix $(\tilde{\boldsymbol{W}}^T \tilde{\boldsymbol{W}})$ is symmetric and positive semidefinite, the unit normal vector $\boldsymbol{b}$ is consequently obtained as the eigenvector associated to the smallest eigenvalue of $(\tilde{\boldsymbol{W}}^T \tilde{\boldsymbol{W}})$.

In addition to the regressors in (7) the vector $\boldsymbol{m}$ now delivers a bias-term or trend-term ([8]).

After partitioning of $\boldsymbol{b}$ according to $\boldsymbol{b} = [b_1, \boldsymbol{\beta}]^T$ TLS offers the following difference equation using the regressors in (7):

$$\hat{y}(k) = \frac{1}{b_1} \left[-\boldsymbol{x}^T(k)\boldsymbol{\beta} + \boldsymbol{m}^T \boldsymbol{b} \right]. \tag{15}$$

The way the trend term is obtained in TLS parameter estimation offers another important advantage over LS when off-equilibrium models are considered. As outlined in cf. [8] these models suffer from the problem that the trend term often dominates over the remaining parameters. An obvious solution is given by regularisation. The separation of the bias from the remaining parameters in TLS provides an automatic means of regularisation that improves the model performance during transients. This will be highlighted in sec. V.

Remark: It can be easily shown that referencing to the centroid does not change the parameter estimation results when applying LS.

C. Decorrelation of identification data

In the minimisation of (6) it is assumed that all measurements in $\boldsymbol{w}^T$ are equally corrupted with noise and that the individual noise sources are uncorrelated. In practical applications these prerequisites are almost never fulfilled. In many cases an input signal is generated by a controller (external excitation) which processes noisy output data such that a correlation over time is generated. In these cases the identification data must be decorrelated prior to parameter identification.

Let $\nu(k)$ denote the noise signal that is superimposed to the true output $y_0(k)$ and let $\mu(k)$ be the noise signal associated

to the true input $u_0(k)$. The data record $\boldsymbol{w}(k)$ is then obtained from its unperturbed equivalent $\boldsymbol{w}_0(k)$ from

$$\boldsymbol{w}^T(k) = \boldsymbol{w}_0^T(k) + \boldsymbol{n}^T(k),$$

with the noise vector

$$\boldsymbol{n}(k) = [\,\nu(k)\,,\,\nu(k-1)\,,\,\ldots$$
$$|\,\mu(k-d-1)\,,\,\mu(k-d-2)\,,\,\ldots]^T. \qquad (16)$$

The covariance matrix $\boldsymbol{R}_n = E\{\boldsymbol{n}(k)\boldsymbol{n}^T(k)\}$ contains all the abovementioned correlations and is assumed to be known. In practical applications it can be determined from data records from stationary phases according to

$$\boldsymbol{R}_n \approx \frac{1}{N-1}(\tilde{\boldsymbol{W}}^T\tilde{\boldsymbol{W}}). \qquad (17)$$

For $N \to \infty$ the approximation (17) converges to the expectation $E\{\boldsymbol{n}(k)\boldsymbol{n}^T(k)\}$. In the practical tests conducted in connection with the presented method it turned out that (17) always led to excellent results.

For a correct application of TLS the individual elements of the noise vector (16) must be uncorrelated. If this is not the case the regressors in (9) are transformed by replacing $\boldsymbol{b}$ by $\boldsymbol{b} = \boldsymbol{T}\tilde{\boldsymbol{b}}$. The relevant noise vector $\tilde{\boldsymbol{n}}$ for the determination of $\tilde{\boldsymbol{b}}$ is obtained by $\tilde{\boldsymbol{n}}^T(k) = \boldsymbol{n}^T(k)\boldsymbol{T}$. A correct optimisation of $\tilde{\boldsymbol{b}}$ through TLS can thus be assured if the transformation matrix $\boldsymbol{T}$ is chosen such that the covariance matrix $\boldsymbol{R}_{\tilde{n}} = E\{\tilde{\boldsymbol{n}}(k)\tilde{\boldsymbol{n}}^T(k)\}$ becomes the identity matrix:

$$\boldsymbol{R}_{\tilde{n}} = \boldsymbol{T}^T\boldsymbol{R}_n\boldsymbol{T} = \boldsymbol{I}. \qquad (18)$$

In the case of pure measurement noise $\boldsymbol{R}_n$ is a diagonal matrix such that $\boldsymbol{T} = \boldsymbol{R}_n^{-1/2}$ is a solution to (18). In more complex cases an eigenvector/eigenvalue decomposition of $\boldsymbol{R}_n$ according to $\boldsymbol{R}_n = \boldsymbol{S}\boldsymbol{D}\boldsymbol{S}^T$ must be carried out. A solution of (18) is then given by $\boldsymbol{T} = \boldsymbol{S}\boldsymbol{D}^{-1/2}$. Finally, the vector $\tilde{\boldsymbol{b}}$ is obtained as the eigenvector corresponding to the smallest eigenvalue of $\boldsymbol{T}^T\tilde{\boldsymbol{W}}^T\tilde{\boldsymbol{W}}\boldsymbol{T}$.

D. Weighted TLS

For the parameter estimation of local models estimation errors are weighted through the validity function Φ_j. Let $\boldsymbol{Q}_j$ denote a diagonal weighting matrix for the j-th local model. Its diagonal elements q_{ji} represent the values of the validity function $\Phi_j(\boldsymbol{u}_\Phi(i))$ at the training data points. Compared to (6) one now defines a modified criterion:

$$J_j = \frac{1}{2N}\left\{\sum_{i=1}^{N} q_{ji}[x_i - \hat{x}_i]^2 + \sum_{i=1}^{N} q_{ji}[y_i - \hat{y}_i]^2\right\} \qquad (19)$$

Minimisation of (19) corresponds to a weighted version of TLS which will be named WTLS. Instead of the Frobenius norm (12) one now has to minimise the following norm:

$$||\boldsymbol{Q}_j^{1/2}(\boldsymbol{W} - \hat{\boldsymbol{W}})||_F =$$
$$= \boldsymbol{b}^T(\boldsymbol{W}^T - \boldsymbol{m}\boldsymbol{1}^T)\boldsymbol{Q}_j(\boldsymbol{W} - \boldsymbol{1}\boldsymbol{m}^T)\boldsymbol{b}. \qquad (20)$$

The centroid $\boldsymbol{\mu}_W$ in (13) is also replaced by its weighted equivalent:

$$\boldsymbol{\mu}_{Wj} = \boldsymbol{W}^T\boldsymbol{q}_j/s_q. \qquad (21)$$

Here $\boldsymbol{q}_j$ denotes the vector composed from the main diagonal of $\boldsymbol{Q}_j$ and s_q is the sum of its elements: $s_q = \boldsymbol{1}^T\boldsymbol{q}_j$. In analogy to (14) one obtains

$$||\boldsymbol{Q}_j^{1/2}(\boldsymbol{W} - \hat{\boldsymbol{W}})||_F = \boldsymbol{b}^T(\tilde{\boldsymbol{W}}^T\boldsymbol{Q}_j\tilde{\boldsymbol{W}})\boldsymbol{b} +$$
$$+ s_q[(\boldsymbol{\mu}_{Wj} - \boldsymbol{m})^T\boldsymbol{b}]^2 = \min. \qquad (22)$$

A solution for the vector $\boldsymbol{m}$ is now $\boldsymbol{m} = \boldsymbol{\mu}_{Wj}$ while the unit normal $\boldsymbol{b}$ (or $\tilde{\boldsymbol{b}}$) is computed from $\tilde{\boldsymbol{W}}^T\boldsymbol{Q}_j\tilde{\boldsymbol{W}}$ instead of $\tilde{\boldsymbol{W}}^T\tilde{\boldsymbol{W}}$.

IV. Stationary constraints

A powerful strategy to improve the steady-state performance of dynamic models consists in *enforcing* the adherence of the model to predefined stationary points. The remaining degrees of freedom of the parameter vector are then used to optimise the model for its dynamic accuracy. This approach has the advantage that there is an exact separation between static and dynamic mapping. Basically one can enforce *one* stationary constraint per local model and input. During stationary phases the regressor (7) becomes

$$\boldsymbol{x}_s^T(i) = [y_s(i)\; y_s(i)\; \ldots | u_s(i)\; u_s(i)\; \ldots]. \qquad (23)$$

The augmented TLS-regressor becomes

$$\boldsymbol{w}_s^T(i) = [\,y_s(i)\,|\,\boldsymbol{x}_s^T(i)\,]. \qquad (24)$$

and after referencing to the centroid $\boldsymbol{m}$

$$\tilde{\boldsymbol{w}}_s^T(i) = \boldsymbol{w}_s^T(i) - \boldsymbol{m}^T. \qquad (25)$$

The subscript s indicates the stationary phase and the index i denotes the number of the stationary phase if there is more than one. If the i-th stationary phase is to be mapped correctly by a TLS model the vector $\boldsymbol{b}$ (or $\tilde{\boldsymbol{b}}$) has to satisfy the following condition:

$$\tilde{\boldsymbol{w}}_s^T(i)\boldsymbol{b} = 0. \qquad (26)$$

If multiple stationary constraints $1, 2, \ldots$ have to be enforced (26) is replaced by

$$\tilde{\boldsymbol{S}}_s\boldsymbol{b} = 0 \quad \text{with} \quad \tilde{\boldsymbol{S}}_s = \begin{bmatrix} \tilde{\boldsymbol{w}}_s^T(1) \\ \tilde{\boldsymbol{w}}_s^T(2) \\ \vdots \end{bmatrix}. \qquad (27)$$

The compliancy of (27) in the optimisation of (14) can be ensured if $\boldsymbol{b}$ (or $\tilde{\boldsymbol{b}}$) is constrained to the null-space of $\tilde{\boldsymbol{S}}_s$:

$$\boldsymbol{b} \in \mathcal{N}(\tilde{\boldsymbol{S}}_s) \quad \text{or} \quad \boldsymbol{b} = \boldsymbol{N}(\tilde{\boldsymbol{S}}_s)\boldsymbol{\xi}. \qquad (28)$$

Here, the column vectors of $\boldsymbol{N}(\tilde{\boldsymbol{S}}_s)$ span the null-space of $\tilde{\boldsymbol{S}}_s$. Inserting $\boldsymbol{b} = \boldsymbol{N}(\tilde{\boldsymbol{S}}_s)\boldsymbol{\xi}$ in (14) or (22) shows that $\boldsymbol{\xi}$ is now the eigenvector associated to the smallest eigenvalue of $\boldsymbol{N}^T\tilde{\boldsymbol{W}}^T\tilde{\boldsymbol{W}}\boldsymbol{N}$ or $\boldsymbol{N}^T\tilde{\boldsymbol{W}}^T\boldsymbol{Q}_j\tilde{\boldsymbol{W}}\boldsymbol{N}$, respectively. After the computation of $\boldsymbol{b}$ or $\tilde{\boldsymbol{b}}$ from $\boldsymbol{b} = \boldsymbol{N}(\tilde{\boldsymbol{S}}_s)\boldsymbol{\xi}$ the transfer function can be determined from (15).

V. Simulation Results

As an example a Wiener model was chosen for illustrative purposes. It consists of a dynamic linear block with a normalized transfer function $G_L(z)$ in cascade with a static nonlinearity $f(v)$ at the output with v as the intermediate variable at the output of the linear block. For the present simulation results $G_L(z)$ and $f(v)$ were chosen as

$$G_L(z) = \frac{V(z)}{U(z)} = \frac{0.01(1.867z^{-1} + 1.746z^{-2})}{1 - 1.7826z^{-1} + 0.8187z^{-2}} \tag{29}$$

$$y(k) = f(v(k)) = \arctan(v(k)). \tag{30}$$

Despite their simple structure the identification of Wiener systems can become challenging, in particular when the nonlinearity $f(v)$ has a saturation character.

To demonstrate the capability of the proposed algorithm the identification data were corrupted with input- and output noise ($\sigma_y = 0.01$, $\sigma_u = 0.01$). The identification data were collected using an APRB-Signal for excitation in $u(k)$. The bandwidth and maximum amplitude were chosen such that the system was sufficiently excited where special attention must paid to the saturation zones ($|u| > 4$). Also, it must be ensured that the system is excited to such an extent that off-equilibrium models (local models with no equilibrium in their region of validity) can be built. For the augmentation of models that contain equilibria six stationary operating points were recorded to serve as potential stationary constraints.

Figure 2 depicts the partition space, which was determined by the training algorithm presented in [14]. The dashed line represents the stationary equilibrium, the aforementioned stationary operating points are shown as circles. The projections of the dynamic identification data are depicted as grey dots. Their distribution in the partition space clearly shows the extent of the dynamic excitation of the process. Local models are represented by contour lines of their validity functions. Those models that are intersected by the equilibrium line are equilibrium models whereas all other models are off-equilibrium models.

The superior performance of the TLS parameter estimation is demonstrated in Tables I and II and through Figures 3 and 4, respectively. The validation results given in the tables and in Figure 3 were generated with noise-free data in order to clearly demonstrate the effects of TLS whereas the validation data shown in Fig. 4 were generated with the noise corruption given above.

The comparison of Tables I and II reveals two phenomena: Looking at the RMSE computed with *identification* data it is visible that the performance of LS seems to be superior. This is due to the fact that LS aims at minimising the prediction error in the identification data whereas TLS minimises the *generalisation error*. Looking at the validation the situation is vice versa and TLS outperforms LS. Figure 3 compares the autocorrelation functions of the prediction errors from validation data. It is clearly visible that TLS also outperforms LS in this context.

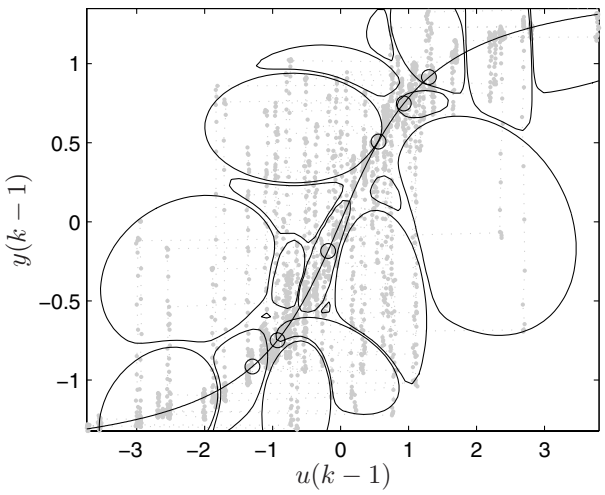

Fig. 2. Wiener model: partition space with dynamic data and stationary points

TABLE I
Wiener model: Root Mean Squared Error (RMSE) for TLS-parameter optimisation

Mode	Data from: identification	validation
prediction	0.0219	$4.7322 \cdot 10^{-3}$
simulation	0.0458	0.0424

TABLE II
Wiener model: Root Mean Squared Error (RMSE) for LS-parameter optimisation

Mode	Data from: identification	validation
prediction	0.0186	$9.3274 \cdot 10^{-3}$
simulation	0.0693	0.0574

As outlined earlier TLS parameter estimation computes the trend term separately from the other parameters. For off-equilibrium local affine models this means that the parameter estimation is not dominated by the trend term (cf. [8]) which results in better accuracy during transients. This fact is reflected in Figure 4 where noisy validation data are compared to simulation results from LS and TLS models respectively.

Finally, Figure 5 highlights the effect of the inclusion of stationary constraints in equilibrium models. The actual constraints are not only met exactly but they also have a positive side-effect on transient states.

VI. Conclusion and Outlook

In this article new concepts for the identification of dynamic Neuro-Fuzzy networks were proposed. First, the Total Least Squares method was introduced as a means of bias-free parameter estimation for local models. Second, a constrained optimisation concept was proposed that allows the incorporation of stationary constraints into the parameter estimation process. A simulation example highlights the effectiveness of the proposed concepts.

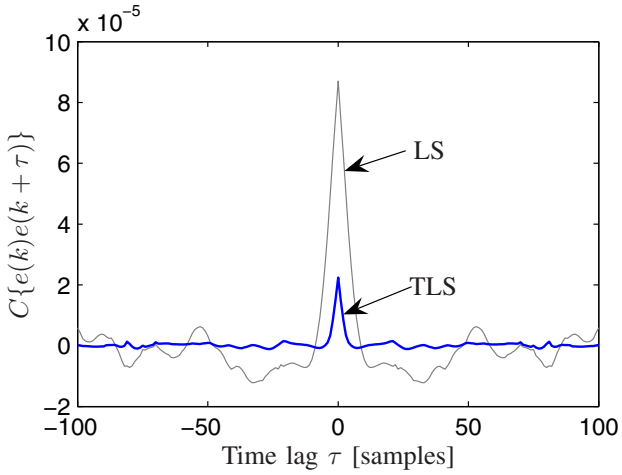

Fig. 3. Wiener model: Autocorrelation function of the prediction error

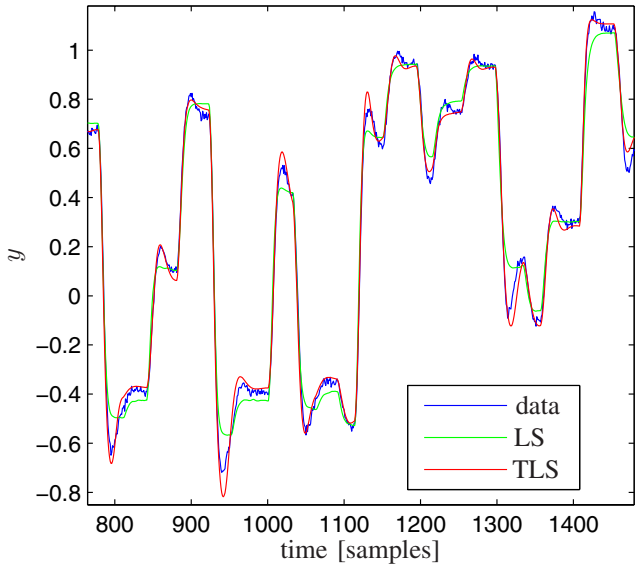

Fig. 4. Wiener model identification: Increased accuracy of TLS-models during strong transients

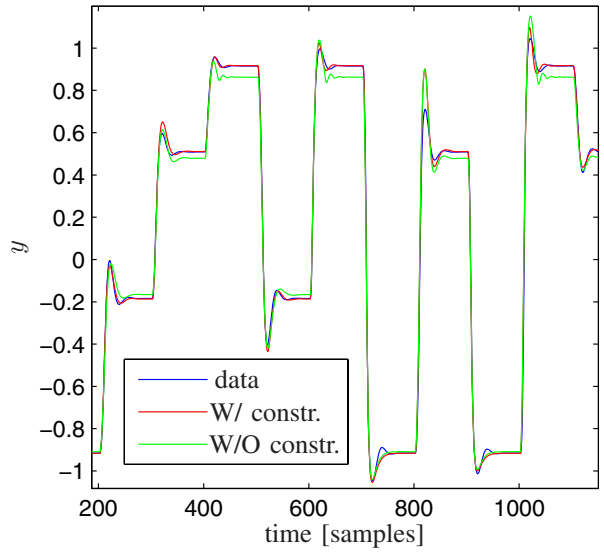

Fig. 5. Wiener model identification: Effect of stationary constraints

REFERENCES

[1] B. A. Foss and T. A. Johansen, "On local and fuzzy modeling," in *In Proceedings of the 3rd Int. Conf. Industrial Fuzzy Control and Intelligent Systems*, 1993, pp. 80 – 87.

[2] T. Johansen and F. B.A., "Identification of non-linear system structure and parameters using regime decomposition," *Automatica*, vol. 31, no. 2, pp. 321 – 326, 1995.

[3] R. Babuska and H. Verbruggen, "An overview of fuzzy modeling for control," *control Engineering Practice*, vol. 4, no. 11, pp. 1593 – 1606, 1996.

[4] S. Ernst, "Hinging hyperplane trees for approximation and identification," in *Proceedings of the 37th IEEE Conference on Decicion & Control 1998*, vol. 2. IEEE, 1998, pp. 1266 – 1271.

[5] R. Babuska, *Recent Advances in Intelligent Paradigms and Applications*. Springer-Verlag, Heidelberg, 2002.

[6] X. Ren, A. Rad, P. Chan, and L. Wai, "Identification and control of continuous-time nonlinear systems via dynamic neural networks," *Industrial Electronics, IEEE Transactions on*, vol. 50, no. 3, pp. 478–486, June 2003.

[7] O. Nelles, *Nonlinear System Identification*, 1st ed. Springer Verlag, 2002.

[8] T. A. Johansen, R. Shorten, and R. Murray-Smith, "On the interpreatation and identification of dynamic takagi-sugeno fuzzy models," *IEEE Transactions on Fuzzy Systems*, vol. 8, no. 3, pp. 297–313, 2000.

[9] D. Kukolj and E. Levi, "Identification of complex systems based on neural and takagi-sugeno fuzzy model," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 34, pp. 272 – 282, Feb. 2004.

[10] T. A. Johansen and R. Babuska, "Multiobjective Identification of Takagi Sugeno Fuzzy Models," *IEEE Transactions on Fuzzy Systems*, vol. 11, pp. 847–860, 2003.

[11] P. P. N. de Groen, "An introduction to total least squares," *Nieuw Archief Voor Wiskunde*, vol. 14, p. 237, 1996, http://www.citebase.org/cgi-bin/citations?id=oai:arXiv.org:math/9805076.

[12] R. Zimmerschied, M. Weber, and I. R., "Static and Dynamic Measurements of Combustion Engines for Optimisation of Control Mappings - a Brief Survey," *Automatisierungstechnik - at*, vol. 53, pp. 87–94, Februar 2005.

[13] T. A. Johansen, H. K.J., and R. Murray-Smith, "Off-equilibrium linearisation and design of gain-scheduled control with application to vehicle speed control," *Control Engineering Practice*, vol. 6, pp. 167–180, 1998.

[14] S. Jakubek, N. Keuth, "A local neuro-fuzzy network for high-dimensional models and optimisation," *Engineering Applications of Artificial Intelligence*, to appear 2006, available online for review only: http://www.impa.tuwien.ac.at/2001-.110.0.html.

[15] S. W. Heij C., "Consistency of system identification by global total least squares," *Automatica*, vol. 35, pp. 993–1008, 1999.

[16] R. B., "Algorithms for global total least squares modelling of finite multivariable time series," *Automatica*, vol. 31, no. 3, pp. 391 – 404, 1995.

[17] I. Markovsky, J. Willems, S. Van Huffel, B. De Moor, and R. Pintelon, "Application of structured total least squares for system identification," in *43rd IEEE Conference on Decision and Control*, vol. 4. IEEE, 2004, pp. 3382–3387.

[18] D. E. Gustafson and W. C. Kessel, "Fuzzy Clustering With the Fuzzy Covariance Matrix," in *Proceedings of the IEEE CDC*. IEEE, 1979, pp. 761–766.