16th Triennial World Congress, Prague, Czech Republic

ELSEVIER
IFAC
PUBLICATIONS

INCREMENTAL IDENTIFICATION OF NARX MODELS BY SPARSE GRID APPROXIMATION

Olaf Kahrs, **Marc Brendel**, **Wolfgang Marquardt** [1]

Lehrstuhl für Prozesstechnik, RWTH Aachen University, D-52064 Aachen, Germany

Abstract: Nonlinear empirical models are used in various applications. During model-building, five major steps usually have to be carried out: model structure selection, determination of input variables, complexity adjustment of the model, parameter estimation and model validation. These steps have to be repeated until a satisfactory model is found, which can be very time consuming and may require user interaction. This paper proposes an algorithm based on sparse grid function approximation to incrementally build a nonlinear empirical model. The algorithm exhibits good performance in terms of manual effort and computation time. The method is illustrated by a case study on the identification of a NARX model.

Keywords: Nonlinear models, sparse grids, input selection, NARX

1. INTRODUCTION

Many important industrial processes exhibit nonlinear behavior. For model-based control and optimization a process model is required. Physically-based process models are often not available due to development and maintenance cost. Therefore an empirical process model may be identified from experimental or plant data as described by e.g. Henson and Seborg (1997).

As a typical example, the identification of discrete-time, nonlinear, auto-regressive models with exogeneous inputs (NARX models) is considered in this paper, although the methodology presented can easily be applied to other model structures as well. The general formulation of a NARX model for a process with a single input $u \in \mathbb{R}$ and a single output $y \in \mathbb{R}$ is $y_k = f(\mathbf{x}_k)$, where f is a nonlinear function. The input variables of the NARX model

$$\mathbf{x}_k = [u_{k-1}, \ldots, u_{k-p}, y_{k-1}, \ldots, y_{k-q}] \quad (1)$$

are delayed samples of the process input and output, using the notation $y_k = y(t = t_k)$. The model order parameters p and q determine how many delayed samples are included in $\mathbf{x}_k$.

Given a measurement data set, the NARX model is usually built in an iterative process that consists of five major steps:

(1) Select a model structure: A suitable model structure for the function f has to be selected from the many alternatives that are proposed in literature, such as artificial neural networks or polynomial models (Henson and Seborg, 1997). The selection is usually strongly influenced by personal preferences and the available software but should be based on the intended application. For instance, Pearson (2003) states four important measures for model utility in process control: approximation accuracy, physical interpretability, ease of controller design and ease of model development.

[1] Corresponding author. E-mail: marquardt@lpt.rwth-aachen.de

(2) Determine input variables: The input variables of the NARX model, i.e. the model order, are often assumed to be known. Alternatively, they are determined by building multiple model candidates with different sets of input variables and selecting the best one. This time-consuming search strategy can be avoided by algorithms that suggest a model order prior to the choice of a model structure as was presented by e.g. Feil *et al.* (2004).

(3) Adjust the model complexity: The concepts of stepwise model refinement, model pruning, i.e. deleting dispensable parameters, and regularization are frequently employed when model complexity is adjusted, see e.g. Bishop (1995) for artificial neural networks. Model complexity is often expressed by the number of model parameters and is determined by the grade of nonlinearity in the data set as well as the desired model accuracy.

(4) Estimate the model parameters: A parameter estimation problem, which is sometimes also combined with step (3), is set up and solved by an algorithm that is usually tailored to the model structure for numerical efficiency.

(5) Validate the identified model: After a solution has been obtained, the model should be validated, in order to decide if some of the preceding steps have to be repeated to improve the model.

The model-building process can be very time-consuming if many model candidates with different sets of input variables and model complexities have to be considered. The goal of this paper is to reduce the repetitions of the model-building steps by a systematic model-building strategy that combines all steps into one algorithm. Brendel and Marquardt (2003) noted that sparse grid approximation may find use for structure selection of nonlinear discrete-time models. This is demonstrated in this work.

Yserentant (1986) proposed the sparse grid approach for multilevel-splitting of finite-element spaces. It was mainly used for the solution of PDEs (Griebel *et al.*, 1992). Recently the sparse grid approach has been introduced to the field of function approximation by Garcke *et al.* (2001) and Brendel and Marquardt (2003). In this paper the stepwise model refinement algorithm of Brendel and Marquardt (2003) is modified to systematically select input variables and adjust model complexity. Furthermore, an alternative regularization term is proposed.

Sparse grid approximation is described briefly in Section 2, which is the basis of the systematic model-building strategy presented in Section 3. This strategy is exemplified in a case study on NARX model identification and the resulting NARX model orders are compared to those found by Feil *et al.* (2004). Finally, some conclusion are given in Section 5.

2. SPARSE GRID FUNCTION APPROXIMATION

Two basic statistical assumptions are commonly made if an output error model $\tilde{y}_k = f(\mathbf{x}_k) + \epsilon_k$ is used in function approximation: (i) the model structure f is reasonably correct and (ii) the measurements of the input variables $\mathbf{x}_k$ in the k-th experiment contain negligible errors. All measurement errors and the model mismatch are described by the error term ϵ_k, which is assumed to have a normal distribution with zero mean and variance σ^2.

The objective is to identify the function $f : \mathbb{R}^d \to \mathbb{R}$ of some function space V from measurement data of the output variable $\tilde{y}$ and the input variables $\mathbf{x}$. This inverse problem may be ill-posed in the sense of Hadamard and therefore requires regularization (Engl *et al.*, 1996).

The function f is the solution of the regularized least-squares estimation problem

$$\min_{f \in V} \frac{1}{M} \sum_{k=1}^{M} \left(f(\mathbf{x}_k) - \tilde{y}_k\right)^2 + \lambda \Phi(f). \qquad (2)$$

The first term measures the average approximation error of the function f to the measured values $\tilde{y}_k$ and the second term represents a regularization term to handle the ill-posedness.

The following subsections briefly describe how the function f is defined in the sparse grid approach. Since the sparse grid solution is composed of multiple so-called full grid solutions, the function approximation on a full grid is presented first.

2.1 Function approximation on a full grid

The function f is represented in the full grid approach by a truncated basis function expansion

$$f_{\mathbf{l}}(\mathbf{x}) := \sum_{\mathbf{j} \in \Lambda} \theta_{\mathbf{l},\mathbf{j}} \phi_{\mathbf{l},\mathbf{j}}(\mathbf{x}) \qquad (3)$$

where $\theta_{\mathbf{l},\mathbf{j}}$ are parameters to be estimated and $\phi_{\mathbf{l},\mathbf{j}}(\mathbf{x})$ are basis functions that are local with finite support in d-dimensional space. The basis functions $\phi_{\mathbf{l},\mathbf{j}}(\mathbf{x})$ are parameterized by the level multi-index $\mathbf{l}$ and the position multi-index $\mathbf{j}$, which have d entries each, where d is the number of variables in $\mathbf{x}$. The level multi-index $\mathbf{l}$ defines the mesh size $h_{\mathbf{l}} = (h_{l_1}, ..., h_{l_d}) = (2^{-l_1}, ..., 2^{-l_d})$ of the full grid in each dimension. The position multi-index $\mathbf{j}$ translates the basis function in each dimension to the coordinates $\mathbf{x}_{\mathbf{l},\mathbf{j}} = (j_1 2^{-l_1}, \ldots, j_d 2^{-l_d})$ with $\mathbf{j} \in \Lambda := \{\mathbf{j} \mid j_i \in \{0, \ldots, 2^{l_i}\}, i = 1 \ldots d\}$. For