Linear Identification of a Steam Generation Plant

Magdi S. Mahmoud *Member, IAENG*

***Abstract*—The paper examines the development of models of steam generation plant using linear identification techniques. The identification process is carried out using experimental data of a multi-input multi-output MIMO system representation. Various techniques of modeling and identification are applied considering the complete system as a MIMO model and studying the effect of all the inputs on individual in different cases. The model will also be studied as a SISO system considering one input and one output at a time.**

***Index Terms*—ARMAX model, Linear identification, State-space model, Steam generation plant.**

I. Introduction

THERE are two types of configurations [1] in the electricity generation using drum boilers and steam turbines:

(1) A single boiler is used to generate steam that is directly fed to a single turbine. This configuration is usually referred to as a boiler-turbine unit.

(2) A header is used to accommodate all the steam produced from several boilers, and the steam is then distributed to several turbines through the header.

The steam can be used to generate electricity as well as other purposes [8]. This configuration is commonly used in industrial utility plants. Boiler-turbine units are nowadays preferred over header systems, because they can achieve quick response to electricity demands from a power grid or network. It is generally accepted that a boiler-turbine unit is a highly nonlinear and strongly coupled complex system.

The objective of this paper is to develop a class of models for steam generation plant using linear identification techniques. The identification process is based on real data collected multi-input multi-output MIMO pilot system. Standard methods of modeling and identification are applied considering the complete system as a MIMO model and studying the effect of all the inputs on individual in different cases. The model will also be studied as a SISO system considering one input and one output at a time. Simulation studies are performed and the ensuing results are evaluated.

II. System Description and Data Analysis

For the system considered in this paper, the input/output experimental data has been obtained from [6] based on real measurements of a steam generator at Abbot Power plant in Champaign IL. The data comes from a model of this steam generator. The inputs are listed as follows.

U1: Fuel scaled 0-1

Manuscript received July 22, 2010. This work is supported by the deanship for scientific research (DSR) at King Fahd University for Petroleum and Minerals (KFUPM) through project **IN100018**.

Magdi S. Mahmoud is with the Systems Engineering Department, King Fahd University for Petroleum and Minerals (KFUPM), PO Box 5067, Dhahran 31261, Saudi Arabia. (Phone: 00-966-54201-9258; Fax: 00-9663-860-2965; e-mail: msmahmoud@ kfupm.edu.sa).

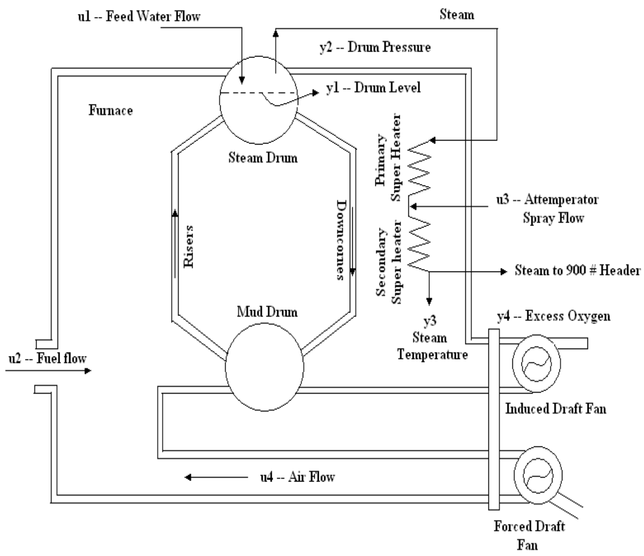

Fig. 1. Steam Generation Plant Diagram

U2: Air scaled 0-1
U3: Reference level
U4: Disturbance defined by the load level

The outputs are

Y1: Drum pressure
Y2: Excess oxygen in exhaust gases
Y3: Level of oxygen in the drum
Y4: Steam flow

Fig. 1 shows the schematic diagram of the Steam generator model. The simulation data constitutes 9600 samples at a sampling rate of 3s. This process is a MIMO one. A set of 4000 samples (5000:9000) are used for testing, another set of 4000 samples (2500:6500) for validation purpose. The important statistical parameters of all inputs and outputs are listed in Table III.

III. Linear System Identification

It is known that system identification deals with the problem of fitting mathematical models to time series of input-output data. The system identification problem addressed hereafter is to estimate a dynamical model of a system based on observed data of a steam generator unit. Several methods to describe a system and to estimate such descriptions exist [1] , [4]. The underlying procedure to determine such a model involves three basic ingredients:

- The input-output data.
- A set of candidate models (the model structure)
- A criterion to select a particular model in the set, based on the information in the data (the identification method).

TABLE I
IMPORTANT STATISTICAL PARAMETERS OF DATABASE

INPUTS/OUTPUTS OF SYSTEM	MEAN	STANDARD DEVIATION	MIN	MAX
INPUT 1: FUEL SCALED 0-1	0.504	0.229	0.000	1.07
INPUT 2: AIR SCALED 0-1	0.528	0.295	0.000	1.07
INPUT 3: REFERENCE LEVEL	0.554	2.460	-4.00	4.53
INPUT 4: DISTURBANCE	0.004	0.010	-0.015	0.023
OUTPUT 1: DRUM PRESSURE	329.4	85.94	154	534
OUTPUT 2: EXCESS OXYGEN IN AIR	4.544	6.157	-0.069	21
OUTPUT 3: DRUM OXYGEN LEVEL	0.552	2.849	-9.55	12.3
OUTPUT 4: STEAM FLOW	14.85	7.571	1.99	34.6

Thus the system identification procedure has a natural logical flow: first *collect data*, then choose a *model set*, then pick *the best model* in this set.

After having settled on the preceding three choices, we have, at least implicitly arrived at a particular model: the one that best describes the data according to the chosen criteria. It then remains to test whether this model is good enough, that is, whether it is valid for its purpose. Such tests are known as model validation. They involve various procedures to assess how the model relates to observed data, to prior knowledge and to its intended use.

IV. PARAMETRIC MODEL STRUCTURES

Parametric models describe dynamical systems in terms of differential equations (time-domain) and transfer functions (frequency-domain). This provides insight into the system physics and compact model structure. Generally, you can describe a system using an equation, which is known as the general-linear polynomial model or the general-linear model, see Fig.2.

The linear model structure provides flexibility for both the system dynamics and stochastic dynamics. However, a nonlinear optimization method is needed to carry-out the estimation of the general-linear model. This method requires intensive computation with no guarantee of global convergence [2]. Simpler models that are a subset of the general linear model structure shown in Fig. 2 are possible. By setting one or more of the polynomials $A(q), B(q), C(q)$ or $D(q)$ equals to 1 one can create several simpler models including AR, ARX, ARMAX, Box-Jenkins, and output-error structures [2]. Each of these methods has their own advantages and disadvantages. Experience has shown that [4] for any particular problem the choice of the model structure to use depends on the dynamics and the noise characteristics of the system. Using a model with more freedom or parameters is not always better as it can result in the modeling of nonexistent dynamics and noise characteristics. This is where physical insight into a system is helpful.

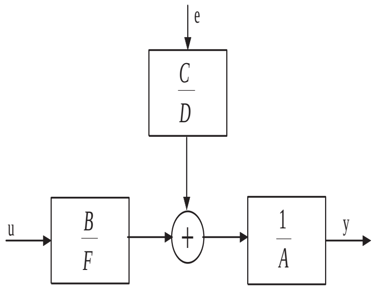

Fig. 2. General Polynomial Model

In what follows, we will summarize the essential features of some the linear identification methods.

A. ARX model

The use of equation error models, also denoted by linear regression models, is widespread in the modeling and identification of dynamical systems. The essential characteristic of the linear regression model is that a residual component is defined which is a linear function of the unknown model coefficients. In the SISO (single input single output) situation we can write:

$$\begin{aligned} y(t) \quad &+ \quad a_1 y(t-1) + \cdots + a_{n_a}(t-n_a) \\ &= \quad b_1 u(t-1) + \cdots + b_{n_b}(t-n_b) + e(t) \end{aligned} \quad (1)$$

with $y(t)$ the output signal, $u(t)$ the input signal of the model, and $a_1 \ldots a_{n_a}$ $b_1 \ldots b_{n_b}$ unknown parameters. The use of these kinds of models in estimation and identification problems is essentially based on the argument that *a least squares identification criterion is an optimization problem that is analytically solvable*.

Since the white noise term $e(t)$ here enters as a direct error in the difference equation, the model is often called *an equation error model*. The adjustable parameters in this case are

$$\theta = [a_1 \ldots a_{n_a} \; b_1 \ldots b_{n_b}] \quad (2)$$

If we introduce

$$\begin{aligned} A(q) &= 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}, \\ B(q) &= b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b} \end{aligned} \quad (3)$$

We see that the model corresponds to

$$G(q,\theta) = \frac{B(q)}{A(q)} \quad H(q,\theta) = \frac{1}{A(q)} \quad (4)$$

In the ARX model the AR refers to the auto-regressive part $A(q)y(t)$ and X refers to the extra input $B(q)u(t)$ (called the exogenous variable). In special case where $n_a = 0, y(t)$ is modeled as a finite impulse response (FIR). Such model sets are particularly common in signal processing applications. The signal flow of ARX model can be depicted as in Fig.3. The signal flow can be depicted as in the Fig.3. From the picture we see that the model is not the most natural one from a physical point of view: the white noise is assumed to go through the denominator dynamics of the system before being added to the output. However, the predictor defines linear regression.

ISBN: 978-988-19251-5-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

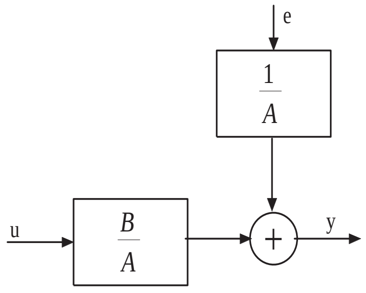

Fig. 3. ARX model structure

Computing the predictor for the system above we get

$$\hat{y}(t|\theta) = B(q)u(t) + [1 - A(q)]y(t) \tag{5}$$

Now introduce the vector

$$\varphi(t) = [-y(t-1) \cdots - y(t-n_a)u(t-1) \dots u(t-n_b)]$$

Then we can write the equation above as

$$\hat{y}(t|\theta) = \theta^T.\varphi(t) = \varphi^T(t).\theta \tag{6}$$

The predictor is a scalar product between a known data vector $\varphi(t)$ and a parameter vector θ. Such a model is called a linear regression in statistics and the vector $\varphi(t)$ is called regression vector. It is of importance since the simple and powerful estimation methods can be applied for the determination of θ. In case some coefficients of the polynomials A and B are known, we arrive at linear regression of the form

$$\hat{y}(t|\theta) = \varphi(t).\theta + \mu(t) \tag{7}$$

where $\mu(t)$ is a known term.

B. *Least square method*

The linear regression employs predictor (6) where φ is defined by

$$\varphi(t) = [-y(t-1) \cdots - y(t-n_a)u(t-1) \dots u(t-n_b)]$$

Using the least squares (LS) criterion for linear regression [4], the prediction error becomes

$$\varepsilon(t,\theta) = y(t) - \varphi^T(t).\theta \tag{8}$$

and the criterion function with $L(q) = 1$ and $l(\varepsilon) = \frac{1}{2}\varepsilon^2$ is

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}\frac{1}{2}[y(t) - \varphi^T(t).\theta]^2 \tag{9}$$

A unique feature of this criterion is that it is a quadratic function in θ. Therefore, it can be minimized analytically, which gives, provided the indicated inverse exists, the least square estimate (LES):

$$\begin{aligned} \hat{\theta}_N^{LS} &= argmin V_N(\theta, Z^N) \\ &= \Big[\frac{1}{N}\sum_{t=1}^{N}\varphi(t)\varphi^T(t)\Big]^{-1}\frac{1}{N}\sum_{t=1}^{N}\varphi(t)y(t) \\ &= [R(N)]^{-1}f(N) \end{aligned} \tag{10}$$

where

$$R(N) = \frac{1}{N}\sum_{t=1}^{N}\varphi(t)\varphi^T(t),\ f(N) = \frac{1}{N}\sum_{t=1}^{N}\varphi(t)y(t)$$

are $d \times d$ matrix and $d \times n$ vector, respectively

If the regression vector $\varphi(t)$ contains lagged input and output variables, the entries of the above equation will be of the form

$$[R(N)]_{ij} = \frac{1}{N}\sum_{t=1}^{N} y(t-i)y(t-j) \quad 1 \le i,j \le n_a$$

Similar sums of $u(t-r).u(t-s)$ or $u(t-r).y(t-s)$ for entries of $R(N)$. They will consist of estimates of the covariance functions of $\{y(t)\}$ and $\{u(t)\}$. The LSE can thus be computed using only such estimates and is therefore related to correlation analysis.

C. *Recursive algorithm*

Suppose that the weighting sequence has the following property:

$$\begin{aligned} \beta(t,k) &= \lambda(t)\beta(t-1,k) \quad 0 \le k \le t-1 \\ \beta(t,t) &= 1 \end{aligned} \tag{11}$$

This means that we may write

$$\beta(t,k) = \Pi_{k+1}^{t}\lambda(j)$$

We note that the following assumption implies that

$$\begin{aligned} \bar{R}(t) &= \lambda(t)\bar{R}(t-1) + \varphi(t)\varphi^T(t) \\ f(t) &= \lambda(t)f(t-1) + \varphi(t)y(t) \end{aligned} \tag{12}$$

Now

$$\begin{aligned} \hat{\theta}_t &= \bar{R}^{-1}(t)f(t) = \bar{R}^{-1}(t)[\lambda(t)f(t-1) + \varphi(t)y(t)] \\ &= \bar{R}^{-1}(t)[\lambda(t)\bar{R}(t-1)\hat{\theta}_{t-1} + \varphi(t)\varphi^T(t)] \\ &= \bar{R}^{-1}(t)\{[\bar{R}(t) - \varphi(t)\varphi^T(t)]\hat{\theta}_{t-1} + \varphi(t)y(t)\} \\ &= \hat{\theta}_{t-1} + \bar{R}^{-1}(t)\varphi(t)[y(t) - \varphi^T(t)\hat{\theta}_{t-1}] \end{aligned} \tag{13}$$

We thus have

$$\begin{aligned} \hat{\theta}_t &= \hat{\theta}_{t-1} + \bar{R}^{-1}(t)\varphi(t)[y(t) - \varphi^T(t)\hat{\theta}_{t-1}] \\ \bar{R}(t) &= \lambda(t)\bar{R}(t-1) + \varphi(t)\varphi^T(t) \end{aligned} \tag{14}$$

which describes a recursive estimation algorithm. At time $t-1$ we store only the finite dimensional information vector

$$X(t-1) = [\hat{\theta}_{t-1}, \bar{R}(t-1)]$$

Since $\bar{R}$ is symmetric, the dimension of X is $d + d(d+1)/2$. At time t this vector is updated using the above equations, which is done with a given, fixed amount of operations.

V. Multivariable ARX Model

Considering the input $u(t)$ to be an m-dimensional vector and the output $y(t)$ to be and p-dimensional vector, we get the generalization of the equation error model as

$$\begin{aligned} y(t) &+ A_1y(t-1) + \cdots + A_{n_a}(t-n_a) \\ &= B_1u(t-1) + \cdots + B_{n_b}(t-n_b) + e(t) \end{aligned} \tag{15}$$

where A_i are $p \times p$ matrices and B_i are $p \times m$ matrices.

We may introduce the polynomials

$$\begin{aligned} A(q) &= I + A_1q^{-1} + \cdots + A_{n_a}q^{-n_a}, \\ B(q) &= B_1q^{-1} + \cdots + B_{n_b}q^{-n_b} \end{aligned} \tag{16}$$

ISBN: 978-988-19251-5-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

These are now matrix polynomials in the backward shift operator q^{-1}.The system here is given by

$$y(t) = G(q,\theta)u(t) + H(q,\theta)e(t)$$

with

$$G(q,\theta) = A^{-1}(q)B(q) \quad H(q,\theta) = A^{-1}(q) \tag{17}$$

The inverse $A^{-1}(q)$ of the matrix polynomial is interpreted and calculated in a straightforward way. Clearly, $G(q,\theta)$ will be a $p \times m$ matrix whose entries are rational functions of q^{-1}. The factorization in terms of two matrix polynomials is also called a (left) matrix fraction description (MFD).

The parameter vector in this case is given by

$$\begin{aligned} \theta &= [A_1 \dots A_{n_a} \; B_1 \dots B_{n_b}] \\ \varphi(t) &= [-y(t-1) \cdots - y(t-n_a)u(t-1) \dots u(t-n_b)] \end{aligned}$$

We get the output equation as

$$y(t) = \theta^T.\varphi(t) + e(t) \tag{18}$$

This is in obvious analogy with linear regression. This can be seen as p different linear regressions, written on top of each other, all with the same regression vector.

When additional structure information is imposed on the parameterization, it is normally no longer possible to use the above equation, since the different output components will not employ identical regression vectors. Then a d-dimensional column vector θ and a $p \times d$ matrix $\varphi^T(t)$ has to be formed such that

$$y(t) = \varphi^T(t)\theta + e(t) \tag{19}$$

A. Least Square Estimation

If the output $y(t)$ is a p-vector and the norm

$$l(\varepsilon) = \frac{1}{2}\varepsilon^T \Lambda^{-1} \varepsilon \tag{20}$$

is used, the LS criterion takes the form

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N} \frac{1}{2}[y(t) - \varphi^T(t).\theta]^T \Lambda^{-1}[y(t) - \varphi^T(t).\theta]$$

This gives the estimate

$$\hat{\theta}_N^{LS} = [\frac{1}{N}\sum_{t=1}^{N} \varphi(t)\Lambda^{-1}\varphi^T(t)]^{-1} \frac{1}{N}\sum_{t=1}^{N}\varphi(t)\Lambda^{-1}y(t)$$

In case we use the particular parameterization with θ as an $r \times p$ matrix,

$$\hat{y}(t|\theta) = \theta^T.\varphi(t)$$

the LS criterion becomes

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N} \|y(t) - \theta.\varphi(t)\|^2$$

with the estimate

$$\hat{\theta}_N^{LS} = [\frac{1}{N}\sum_{t=1}^{N}\varphi(t)\varphi^T(t)]^{-1}\frac{1}{N}\sum_{t=1}^{N}\varphi(t)y^T(t)$$

The above expression brings out the advantages of the structure

$$\hat{y}(t|\theta) = \theta^T.\varphi(t)$$

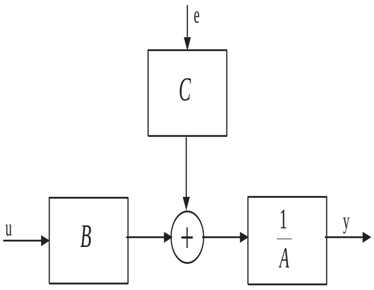

Fig. 4. ARMAX Model Structure

To determine the $r \times p$ estimate $\hat{\theta}_N$, it is sufficient to invert an $r \times r$ matrix.

The recursive algorithm however follows the same procedure as in the single input single output case described above.

VI. ARMAX Model

The basic problem with the ARX model is the lack of adequate freedom in describing the properties of the disturbance term. We could add flexibility to that by describing the equation error as a moving average of white noise. This gives the model:

$$\begin{aligned} y(t) &+ a_1 y(t-1) + \cdots + a_{n_a}(t-n_a) \\ &= b_1 u(t-1) + \cdots + b_{n_b}(t-n_b) + e(t) \\ &+ e(t) + c_1 e(t-1) + \cdots + c_{n_c} e(t-n_c) \end{aligned} \tag{21}$$

It can be rewritten as

$$A(q)y(t) = B(q)u(t) + C(q)e(t) \tag{22}$$

where

$$\begin{aligned} A(q) &= 1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a} \\ B(q) &= b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b} \\ C(q) &= 1 + c_1 q^{-1} + \cdots + c_{n_c} q^{-n_c} \end{aligned} \tag{23}$$

and

$$G(q,\theta) = \frac{B(q)}{A(q)} \quad H(q,\theta) = \frac{C(q)}{A(q)} \tag{24}$$

the parameter vector is given by

$$\theta = [a_1 \dots a_{n_a} \; b_1 \dots b_{n_b} \; c_1 \dots c_{n_c}]$$

In the view of the moving average (MA) part $C(q)e(t)$ the model is called ARMAX model. The ARMAX model has become a standard tool in control and econometrics for both system description and control design.

A version with an enforced integration in the system description is the ARIMA(X) model which is useful to describe the systems with slow disturbances. The signal flow in ARMAX Model can be depicted as in Fig.4 The predictor for the ARMAX model can be obtained as

$$\hat{y}(t|\theta) = \frac{B(q)}{C(q)}u(t) + \Big[1 - \frac{A(q)}{C(q)}\Big]y(t) \tag{25}$$

or

$$C(q)\hat{y}(t|\theta) = B(q)u(t) + [C(q) - A(q)]y(t) \tag{26}$$

ISBN: 978-988-19251-5-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

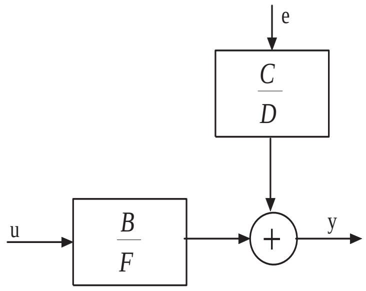

Fig. 5. Box-Jenkins Model Structure

This means that the prediction is obtained by filtering u and y through a filter with denominator dynamics determined by $C(q)$. The predictor can be rewritten in formal analogy as follows.

$$\begin{aligned} \hat{y}(t|\theta) &= B(q)u(t) + [1 - A(q)]y(t) \\ &+ [C(q) - 1][y(t) - \hat{y}(t|\theta)] \end{aligned} \tag{27}$$

Introducing the prediction error

$$\varepsilon(t, \theta) = y(t) - \hat{y}(t|\theta)$$

and the vector

$$\varphi(t) = [-y(t-1) \cdots - y(t-n_a)u(t-1)\ldots u(t-n_b) \\ \varepsilon(t-1,\theta)\ldots\varepsilon(t-n_c,\theta)]$$

Then the predicted output can be written

$$\hat{y}(t|\theta) = \varphi^T(t,\theta).\theta \tag{28}$$

Due to the non linear effect of θ in the vector $\varphi(t,\theta)$ it is called pseudo-linear regression.

VII. Box-Jenkins Model

A natural development of the output error model is to further the properties of the output error. This can be done by assuming that the true process is

$$y(t) = \frac{B(q)}{F(q)}u(t) + \frac{C(q)}{D(q)}e(t) \tag{29}$$

Where

$$\begin{aligned} B(q) &= b_1q^{-1} + \cdots + b_{n_b}q^{-n_b} \\ F(q) &= 1 + f_1q^{-1} + \cdots + f_{n_f}q^{-n_f} \\ C(q) &= 1 + c_1q^{-1} + \cdots + c_{n_c}q^{-n_c} \\ D(q) &= 1 + d_1q^{-1} + \cdots + d_{n_d}q^{-n_d} \end{aligned} \tag{30}$$

In a practical sense, this would seem the most natural finite-dimensional parameterization and the transfer functions $G(.)$ and $H(.)$ are independently parameterized as rational functions. This model, see Fig.5, gives us a family of output error related models.

The parameter vector in this case is given by

$$\theta = [b_1 \ldots b_{n_b}\ f_1 \ldots f_{n_f}\ c_1 \ldots c_{n_c}\ d_1 \ldots d_{n_d}]$$

VIII. State-Space Model

In state-space models, the relationship between the input, noise and output signals is written as a system of first order differential or difference equations using an auxiliary state vector $x(t)$. The description of linear dynamical systems is especially useful in that it insights into physical mechanisms of the system can usually be more easily incorporated into state-space models than into the polynomial models. For most physical systems it is easier to construct models with physical insight in continuous time than in discrete time, simply because most laws of physics are expressed in continuous time. This means that the modeling normally leads to a representation

$$\dot{x}(t) = F(\theta)x(t) + G(\theta)u(t) \tag{31}$$

Here for an n-dimensional system and an m-dimensional input, the matrices F and G are matrices of dimensions $n \times n$ and $n \times m$, respectively. Moreover θ is a vector of parameters that correspond to the unknown values of physical coefficients, material constants, and the like. The modeling is usually carried out in terms of the state variables x that have physical significance and then the measured outputs will be known combinations of the states.

Let $\eta(t)$ be the measurements that would be obtained with ideal, noise-free sensors:

$$\eta(t) = Hx(t) \tag{32}$$

Using p as the differential operator the above state representation can be written as

$$[pI - F(\theta)]x(t) = G(\theta)u(t) \tag{33}$$

Which means that the transfer function from u to η is

$$\begin{aligned} \eta(t) &= G_c(p,\theta)u(t) \\ G_c(p,\theta) &= H[pI - F(\theta)]^{-1}G(\theta) \end{aligned} \tag{34}$$

Let the measurements be sampled at time instants $t = kT, k = 1, 2, \ldots$ and the disturbance effects at those time instants be $v_T(kT)$. Hence the measured output is

$$\begin{aligned} y(kT) &= Hx(kT) + v_T(kT) \\ &= G_c(p,\theta)u(t) + v_T(kT) \end{aligned} \tag{35}$$

There are several ways of transporting $G_c(p,\theta)$ to a representation that is explicitly discrete time. Suppose that the input is constant over the sampling interval T

$$u(t) = u_k = u(kT), \quad kT \le t \le (k+1)T \tag{36}$$

Then (31) can be easily solved from $t = kT$ to $t = kT + T$, yielding

$$x(kT + T) = A_T(\theta)x(kT) + B_T(\theta)u(kT) \tag{37}$$

where

$$A_T(\theta) = e^{F(\theta)\tau}$$

$$B_T(\theta) = \int_{\tau=0}^{\tau} e^{F(\theta)\tau}G(\theta)d\tau \tag{38}$$

Introducing q as the forward shift of T time units, we can write

$$[qI - A_T(\theta)]x(kT) = B_T(\theta)u(kT)$$

ISBN: 978-988-19251-5-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Or

$$\eta(kT) = G_T(q,\theta)u(kT)$$

$$G_T(q,\theta) = H[qI - A_T(\theta)]^{-1}B_T(\theta)$$

Hence the output from the sampled data can be got as

$$y(t) = G_T(q,\theta)u(t) + v_T(t), \quad t = T, 2T, 3T \ldots$$

IX. MIMO Simulation Studies

Multivariable systems are often more challenging to model. In particular, systems with several outputs could be difficult. A basic reason for the difficulties is that the coupling between several inputs and outputs leads to more complex models. The structures involved are richer and more parameters will be required to obtain a good fit.

However, models for prediction and control will be able to produce better results if constructed for all outputs simultaneously. Therefore, in this section, the complete steam generator system will be treated using MIMO ARX model and state-space model.

A. MIMO ARX model

For simulation using the MIMO ARX Model all the four inputs and outputs of the system were considered. For simplicity in exposition and to avoid bias, the values of the system matrices were specified as

$$n_a = \begin{bmatrix} 228 & 8 & 8 & 8 \\ 8 & 8 & 8 & 8 \\ 8 & 8 & 8 & 8 \\ 8 & 8 & 8 & 8 \end{bmatrix}, \quad n_b = \begin{bmatrix} 6 & 9 & 6 & 9 \\ 6 & 9 & 6 & 9 \\ 9 & 9 & 6 & 9 \\ 9 & 6 & 6 & 9 \end{bmatrix}$$

$$n_c = \begin{bmatrix} 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 \\ 3 & 3 & 3 & 3 \end{bmatrix}$$

It must be noted that the coefficients n_a, n_b and n_c were selected to yield the best fitness levels. The delay coefficients were however not considered as optimum results were available without delay in the system. The model was constructed using samples over the range $[5000 - 9000]$. The validation of the model so obtained was carried out on samples over the range $[2500 - 6500]$.

The results of the simulation are plotted in Fig.6 along with the percentage fitness of the various modeled outputs with respect to the measured outputs for the MIMO ARX model.

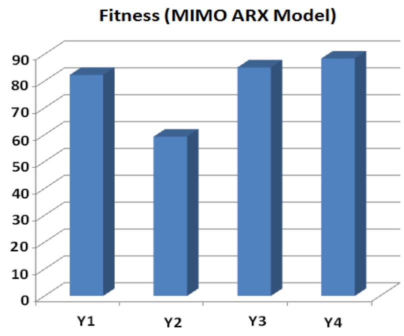

Fig. 6. Fitness Levels for MIMO ARX Model

B. State-Space model

The result of several simulation experiments indicated that a state space model of order 6 yields optimum results. Further increase in the order yielded only a negligible increase in the fitness of the models. With the model order equal to 9, lower fitness levels were obtained, but it showed better results in the residual analysis. Therefore, to strike a balance between the complexity of the model, the fitness and residuals the order of the state space model was selected as 6. The system matrices so obtained are

$$A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} 0.9801 & -0.0075 & 0.0112 \\ 0.0119 & 0.8827 & -0.0618 \\ -0.0093 & 0.0371 & 0.9315 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} -0.0027 & -0.0058 & 0.0000 \\ 0.0389 & -0.1092 & -0.0363 \\ -0.1377 & 0.2576 & -0.0077 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0.1283 & -0.2289 & -0.1460 \\ 0.1213 & -0.1286 & 0.0468 \\ -0.0705 & 0.0478 & 0.1061 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} -0.3670 & -0.1585 & 0.2250 \\ 0.4288 & 0.0213 & 0.2496 \\ 0.2198 & -0.5664 & 0.9134 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0010 & -0.0003 & -0.0001 & -0.0240 \\ -0.0154 & -0.0012 & -0.0004 & -0.3559 \\ 0.0145 & -0.0100 & 0.0004 & -0.1065 \\ -0.0884 & -0.0730 & 0.0042 & -0.1648 \\ -0.0982 & 0.0605 & -0.0002 & 0.3970 \\ -0.0087 & 0.0055 & 0.0003 & 0.0463 \end{bmatrix}$$

$$C = \begin{bmatrix} C_1 & C_2 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 4.0849 & -0.2798 & 0.4308 \\ -0.0376 & 0.0352 & 0.0228 \\ -0.0353 & -0.0904 & 0.0802 \\ 0.1698 & -0.1087 & -0.1254 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 0.0194 & 0.1252 & 0.0915 \\ -0.0172 & 0.0147 & -0.0826 \\ 0.0025 & 0.0185 & 0.0108 \\ 0.0108 & -0.0120 & -0.0061 \end{bmatrix}$$

$$D = 0$$

Just as in the earlier case, The model was constructed using samples over the range $[5000 - 9000]$. The validation of the model so obtained was carried out on samples over the range $[2500 - 6500]$. The results of the simulation are plotted in Fig.7 along with the percentage fitness of the various modeled outputs with respect to the measured outputs for the state space model.

C. Comparison of MIMO Models

A comparison between the fitness percentages between the MIMO ARX model and the state space Model is shown in Table IX-C. Generally speaking, it is better to work with state space models in the multivariable case, since the model structure complexity is easier to deal with. It is essentially a matter of choosing the model order. However we observe that the fitness levels for

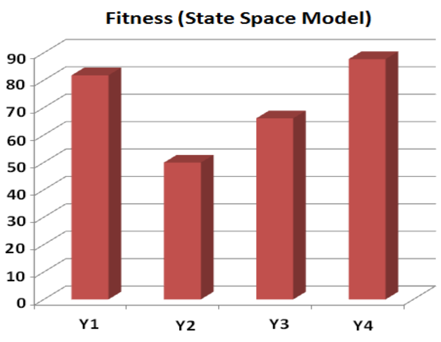

Fig. 7. Fitness Levels for State-Space Models

TABLE II
COMPARISON OF FITNESS PERCENTAGES

Outputs	Percentage Fitness	
	MIMO ARX Model	State Space Model
y1	82.27	81.79
y2	49.59	49.93
y3	84.54	66.1
y4	88.21	87.7

the MIMO ARX model are relatively greater than the state space model.

The ARX model is the simplest model incorporating the stimulus signal. The estimation of the ARX model is the most efficient of the polynomial estimation methods because it is the result of solving linear regression equations in analytic form. Moreover, the solution is unique. In other words, the solution always satisfies the global minimum of the loss function. The ARX model therefore is preferable, especially when the model order is high.

D. Residuals analysis

The leftovers from the modeling process - the part of the data that the model could not reproduce are the residuals

$$\varepsilon(t) = \varepsilon(t, \hat{\theta}_N) = y(t) - \widehat{y(t|\hat{\theta}_N)}$$

It is clear that these bear information about the quality of the model. If we have a data set Z^N, be it estimation or validation or a nominal model m. We wish to know the quality of the model, which in a sense is a statement about how it will be able to reproduce new data sets.

A simple and pragmatic approach is to compute the basic

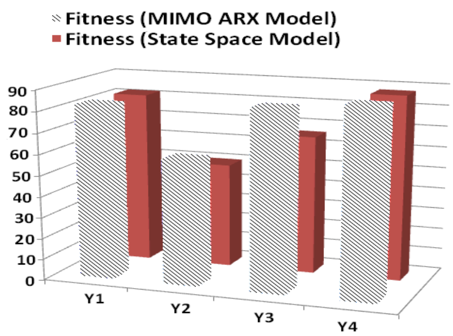

Fig. 8. Comparison of Fitness Levels of MIMO Models

statistics for the residuals from the model:

$$S_1 = max|\varepsilon(t)|, \quad S_2^2 = \frac{1}{N}\sum_{t=1}^{N} \varepsilon^2(t)$$

This means that the model m has never produced a residual than S_1 (or an average error of S_2) for all data we have seen. It's likely that such a bound will hold also for future data.

The statistics in the above equation have an implicit invariance assumption: The *residuals do not depend on something that is likely to change*. Of special importance is that they do not depend on the particular input used in Z^N. If they did, the values of S_1 and S_2 would be limited, since the model should work for a range of possible inputs. To check this it is reasonable to study the covariance between residuals and past inputs:

$$\hat{R}_{\varepsilon u}^N = \frac{1}{N}\sum_{t=1}^{N} \varepsilon(t)u(t-\tau)$$

If these numbers are small we have some reason to believe that the measures S_1 and S_2 could have relevance also when the model is applied to other inputs.
Another way to express the importance of $\hat{R}_{\varepsilon u}^N$ being small is as follows: If there are traces of past inputs in the residuals, then there is a part of $y(t)$ that originates from the past input and that has not been properly picked up by the model m. Hence the model could be improved.

Similarly if we can find correlation among the residuals themselves, that is, if the numbers

$$\hat{R}_{\varepsilon}^N(\tau) = \frac{1}{N}\sum_{t=1}^{N} \varepsilon(t)\varepsilon(t-\tau)$$

are not small for $\tau \neq 0$, then part of $\varepsilon(t)$ could have been predicted from past data. This means that $y(t)$ could have been predicted, which again is a sign of deficiency in the model.

Residual analysis was carried out on the MIMO ARX and MIMO state space models and it is clearly seen that the ARX model is able to reproduce new data sets better than the state space model.

Next we look at the simulation of multi-input single-output (MISO) cases.

X. MISO SIMULATION RESULTS

In the process of identifying good models of a system it is often useful to select subsets of the input and output channels. Partial models of the systems behavior will then be constructed. It might not for example, be clear if all the measured inputs have a significant influence on the outputs.

That is most easily tested by removing an input channel from the data, building a model for how the outputs depend on the remaining input channels and checking if there is a significant deterioration in the model output's fit to the measured one.

Generally speaking, the fit gets better when more inputs are included and worse when more outputs are included. To understand the latter fact, it should be realized that a model that has to explain the behavior of several outputs has a tougher job than one that simply must account for a single output.

If there are difficulties to obtain good models for a multi-output system, it might thus be wise to model one output at a time, to find out which are the difficult ones to handle. Therefore we have divided our system into partial MISO systems considering the effect of all the inputs on each output individually.

A. MISO ARX model

For simulation of the system using the MISO ARX model four separate cases were considered in which the effect of all the four inputs was studied with respect to each output Y_1, Y_2, Y_3 and Y_4 taken one at a time.

ISBN: 978-988-19251-5-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

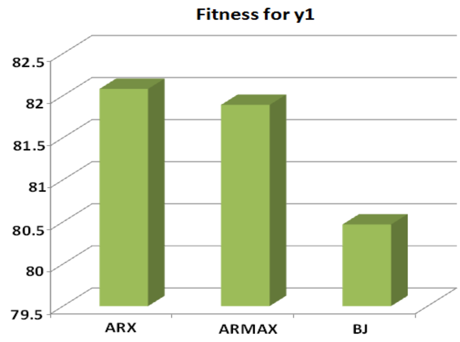

Fig. 9. Fitness of Various MISO Models for Y_1

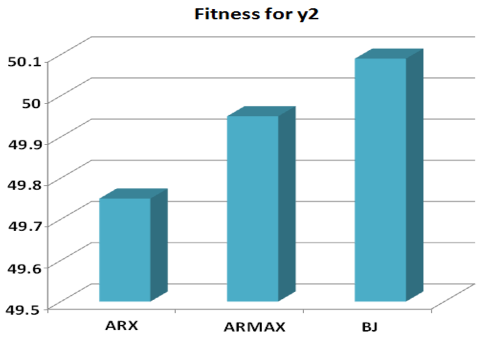

Fig. 10. Fitness of Various MISO Models for Y_2

Considering the MISO ARX model for the output Y_1 the selected orders of the model were:

$$\begin{aligned} n_a &= 20 \\ n_b &= \begin{bmatrix} 25 & 25 & 25 & 25 \end{bmatrix} \\ n_k &= \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned}$$

The delay coefficients were not considered since satisfactory results were obtained even without them. Once again, the model was constructed using samples over the range $[5000 - 9000]$. The validation of the model so obtained was carried out on samples over the range $[2500 - 6500]$.

B. MISO ARMAX model

For the MISO ARMAX model for the output the selected orders of the model were:

$$\begin{aligned} n_a &= 15 \quad n_b = \begin{bmatrix} 15 & 15 & 15 & 15 \end{bmatrix} \\ n_c &= 1 \quad n_k = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned}$$

The orders of the MISO Box Jenkins model used for output Y_1 were:

$$\begin{aligned} n_b &= \begin{bmatrix} 5 & 5 & 5 & 5 \end{bmatrix} \\ n_f &= \begin{bmatrix} 15 & 15 & 15 & 15 \end{bmatrix} \\ n_c &= 5 \quad n_d = 5 \\ n_k &= \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned}$$

The MISO ARX, ARMAX, BJ models were generated for all the four outputs Y_1, Y_2, Y_3 and Y_4. The fitness of various MISO models for the different output is shown in Figs 9 through 12. The fitness levels in each modeling case are now combined together in Fig.13 to facilitate the analysis of the results and is shown below:

It is quite evident from the simulation results, fitness plots and comparison of prediction errors that the ARMAX model has shown better results in the simulation of the multi-input single output systems.

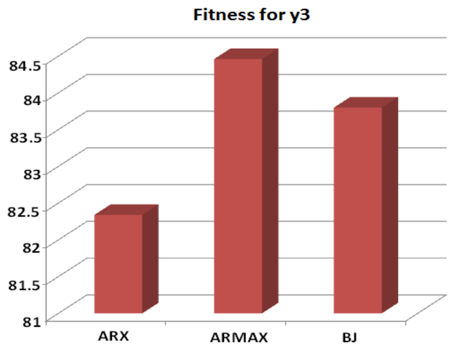

Fig. 11. Fitness of Various MISO Models for Y_3

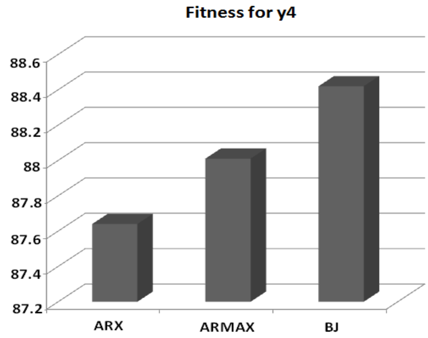

Fig. 12. Fitness of Various MISO Models for Y_4

Selection of one output at a time has also helped in deciding which output is difficult to model. It is very clear from the above results that the second output, namely Oxygen level in flue gases is difficult to model. A main reason for the same, as is visible in the plot of the measured output is that the level of Oxygen in flue gases remains at zero for indefinite intervals of time. It is also desirable that the level of Oxygen in flue gases remains at zero.

XI. SISO Simulation Studies

Models used for simulations could be very well built for single output, considering single input. These models cannot however be used for control purposes and are intended solely for analysis of the behavior of individual inputs and outputs of the system.

Considering all the inputs to the steam generation plant, it is found that the most significant input is the fuel flow rate and the most significant output is the drum pressure. Therefore simulation of the SISO model was carried out taking into account the above stated input and output.

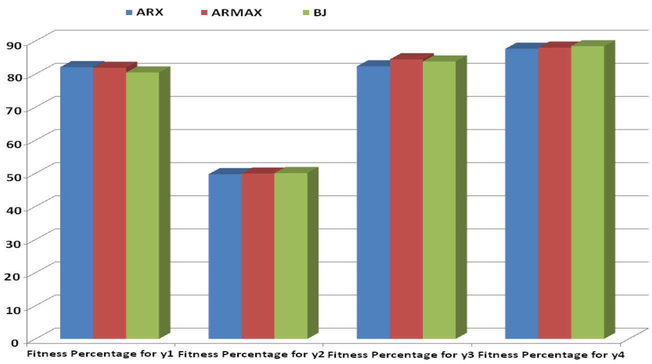

Fig. 13. Comparison of Fitness Levels of MISO Models

ISBN: 978-988-19251-5-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

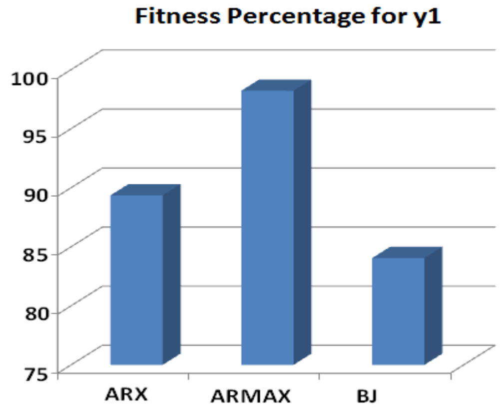

Fig. 14. Fitness of Various SISO Models for Y_1

The sample ranges were selected very carefully and the sets of samples used for modeling and validation were small to ensure better fitness levels.

In the different cases, the samples from 1000 - 1200 were used for creating the ARX model and the samples from 500 - 700 were used for validation.

A. SISO ARX model

The orders of the ARX model were selected as

$$n_a = 20 \quad n_b = 25 \quad n_k = 0$$

B. SISO ARMAX Model

The number of coefficients of the ARMAX model was specified as

$$n_a = 15 \quad n_b = 15 \quad n_c = 1 \quad n_k = 0$$

C. SISO Box-Jenkins model

The selected order of the BJ model was

$$\begin{aligned} n_a &= 15 \quad n_f = 20 \quad n_c = 5 \\ n_d &= 5 \quad n_k = 0 \end{aligned}$$

The fitness percentages of the three models are compared and the results are shown in Fig.14 The fitness level of the ARMAX modeled data was found to be the best. One possible reason is the influence of disturbance. Unlike the ARX model, the ARMAX model structure includes disturbance dynamics. ARMAX models are useful when you have dominating disturbances that enter early in the process, such as at the input. The ARMAX model has more flexibility in the handling of disturbance modeling than the ARX model.

The Box-Jenkins (BJ) structure provides a complete model with disturbance properties modeled separately from system dynamics. The Box-Jenkins model is useful when you have disturbances that enter late in the process. For example, measurement noise on the output is a disturbance late in the process.

XII. Conclusions

There are a variety of model structures available to assist in modeling the steam generator system. The choice of model structure is based upon an understanding of the system identification method and insight and understanding into the system undergoing identification. The characteristics of both system and disturbance dynamics play a role is the proper selection of the model. These system identification methods can handle a wide range of system dynamics without knowledge of the actual system physics, thereby reducing the engineering effort required to develop models.

With respect to the complexity of the model, the fitness levels and the residual analysis it is concluded that the ARX model suits the given system best for the data history provided. However it is noted that the ARX model may not be as good if disturbance dynamics are considered.

Acknowledgment

This work is supported by the deanship for scientific research (DSR) at KFUPM under project no. **IN100018**.

References

[1] W. Tan., H. J Marquez, T. Chen and J. Liu, ”Analysis and control of a nonlinear boiler-turbine unit”, Journal of Process Control, vol. 15, 2005, pp. 883-891.
[2] Ljung, L., ”System Identification - Theory for use”, Prentice-Hall, Second edition, 1999.
[3] M. E. Flynn and M. J. 0’ Malley, ”A Drum Boiler Model for Long Term Power System Dynamic Simulation”, IEEE Trans. Power Systems, vol. 14, o. 1, 1999, pp. .
[4] Y. Zhu, ”Multivariable System Identification for Process Control”, 2001.
[5] S. V. Dudul, ”Identification of a Steam Generator using Focused Time Lagged Recurrent Neural Network Model with Gamma Memory”, IEI Journal.
[6] B. L. R. De Moor, (ed.), ”DaISy: Database for the Identification of Systems”, Dept. Electrical Engineering, ESAT/SISTA, K. U. Leuven, Belgium, URL: http://www.esat.kuleuven.ac.be/sista/daisy /
[7] V. Verhaegen, and V Verdult, ”Filtering and System Identification - A Least Square Approach”, 2007.
[8] W.H. Kwon, S.W. Kim, P.G. Park, ”On the Multivariable Robust Control of a Boiler-Turbine System”, in: IFAC Symposium on Power Systems and Power Plant Control, Seoul, Korea, 1989, pp. 219-223.