

# Online Prediction of Influent Characteristics for Wastewater Treatment Plants Management Using Adaptive Recursive NNARMAX Model

Vincent A. Akpan<sup>1,\*</sup>, Reginald A. O. Osakwe<sup>2</sup>

<sup>1</sup>Department of Physics Electronics, The Federal University of Technology, P.M.B. 704, Akure, Ondo State, Nigeria

<sup>2</sup>Department of Physics, The Federal University of Petroleum Resources, P.M.B. 1221 Effurun, Delta State, Nigeria

**Abstract** A novel technique for the nonlinear modeling and online prediction of incoming influent characteristics of an activated sludge wastewater treatment (AS-WWTP) is presented in this paper. The nonlinear modelling and online prediction in the presence of disturbances is achieved using an online adaptive recursive least squares (ARLS) algorithm to the nonlinear model identification formulated in this paper. The performance of the proposed ARLS algorithm is compared with the so-called incremental backpropagation (INCBP) which is also an online identification. These two algorithms are validated by one-step, five-step ahead prediction methods as well as the Akaike's method to estimate the final prediction error (AFPE) of the regularized criterion. Furthermore, the validation results show the superior performance of the proposed ARLS algorithm in terms of much smaller prediction errors when compared to the INCBP algorithm. The results from the incoming influent characteristics predictions show three scenarios, namely: high toxic, low toxic and acceptable toxic levels of the incoming influent. The proposed techniques and algorithms can be adapted and deployed for the modeling and prediction of an incoming influent (sewage) for industrial WWTP management systems.

**Keywords** ARLS, Artificial neural network (ANN), AS-WWTP, Benchmark simulation model No. 1 (BSM #1), INCBP, Influent (sewage) characteristics

## 1. Introduction

The wastewater treatment is extremely important for humans as well as animals and plants. Generally, the wastewater is exposed to different processes which can remove most of the pollutants such as organic substances, ammonium, phosphorus, nitrogen and other residuals from industrial environment and urban or rural communities. Excess nitrogen and phosphorus in surface waters and nitrogen in groundwater causes eutrophication (i.e. excess algae growth) in surface waters and causes health related problems in humans and livestock as a result of high intake of nitrogen in its nitrate form. Also the effluent quality from industrial wastewater treatment plants are now subjected to tighter regulation as a result of these nutrients as well as nitrogen and phosphorus in both public and receiving waters.

Wastewater treatment processes are very complex, strongly nonlinear and characterized by uncertainties regarding its parameters [1]. In previous studies, the complete modelling of the WWTP process has been a

challenging problem and thus researchers decomposes the WWTP process into two parts namely; the biological reactors and the settler. Also, attempts have been made to use the well-celebrated neural network due to its approximating capability for the complete modeling of the WWTP process as a single multivariable system i.e. combining the biological reactors and the settler as one entity. In this study, attempt is made to obtain a neural network model of an incoming influent (sewage) that will be used to predict the influent characteristics in order to take decisions on the channelling of the incoming influent to the appropriate channel for the proper management of an AS-WWTP process.

The activated sludge process was developed in England in 1914 by Arden and Lockett [2] and was so named because it involved the production of an activated mass of microorganisms capable of aerobically stabilizing a waste. The activated sludge process has been utilized for treatment of both domestic and industrial wastewaters for over half a century. This process originated from the observation made a long time ago that whenever wastewater, either domestic or industrial, is aerated for a period of time, the content of organic matter is reduced, and at the same time a flocculent sludge is formed. Microscopic examination of this sludge reveals that it is formed by a heterogeneous population of microorganisms, which changes continually in nature in

\* Corresponding author:

avincent@ieee.org (Vincent A. Akpan)

Published online at <http://journal.sapub.org/ajis>

Copyright © 2014 Scientific & Academic Publishing. All Rights Reserved

response to variation in the composition of the wastewater and environmental conditions. Microorganisms present are unicellular bacteria, fungi, algae, protozoa, and rotifers.

Wastewater normally contains thousands of different organics and a measurement of each individual organic matter would be impossible rather a different collective analyses are used which comprise a greater or minor part of the organics. Activated sludge wastewater treatment processes are difficult to control because of their complexity; nonlinear behaviour; large uncertainty in uncontrolled inputs and in the model parameters and structures, and multiple time scales of the dynamics and multivariable input-output structure. The activated sludge process aims to achieve, at minimum costs, a sufficiently low concentration of biodegradable in the effluent together with minimal sludge production; and this is achieved through efficient control of the process. The first control opportunity in ASWWTP is in regulating the influent flow-rate which implies that control issues in wastewater treatment facilities pertain primarily to aeration control for energy usage and satisfying process demands [3].

While the dissolve oxygen concentration is considered as the most important control parameter in activated sludge process (ASP), the control of dissolved oxygen level in the ASWWTP reactors plays an important role in the operation of the plant. DO concentration control of the ASP has been recognized as a rewarding and meaningful control, both from economical and biological point of view [1]. Several research papers have been published on how to model and control WWTPs [4]–[8]. The review of most research papers in this area considers modeling and control issues in wastewater treatment facilities in terms of aeration control for energy usage and satisfying process demands. Unlike [3], the view of this paper, however, is that the first control opportunity in ASWWTP should be on the prediction of the characteristics of the incoming influent before the modeling and control of the complete WWTP to determine the toxic content level of the incoming influent.

The sensors and actuators employed for parameter measurements and control of WWTP processes are very expensive; and many researchers have been searching for alternative ways to carry out these measurements [9]. The life span the sensors are usually shortened due to the characteristics and quality of the incoming influent (sewage). Thus, it becomes imperative to know in advance the characteristic of the incoming influent to determine the degree of pretreatment or disposal of the incoming influent while bearing in mind the international regulations on the disposal of waste in receiving waters.

The objective of this paper is directed towards the online prediction of influent characteristics of the incoming sewage for the proper management of WWTPs. The paper is organized as follows. Section 2 presents the WWTP process description. The neural network-based adaptive recursive

least squares (ARLS) algorithm is summarized in Section 3. The online prediction problem of the influent characteristics is formulated and integrated with the proposed neural network identification scheme is given in Section 4. The implementation of the online prediction of the influent characteristics and the simulation results are presented in Section 5. Section 6 concludes the paper with some discussions and future directions.

## 2. The AS-WWTP Process Description

With the tight effluent requirements defined by the European Union and to increase the acceptability of the results from wastewater treatment analysis, the generally accepted COST Actions 624 and 682 benchmark simulation model no. 1 (BSM1) model is considered [1], [4]–[8]. The BSM1 model uses eight basic different processes to describe the biological behaviour of the AS-WWTP processes. The combinations of the eight basic processes results in thirteen different observed conversion rates as described in Appendix A. These components are classified into soluble components ( $S$ ) and particulate components ( $X$ ). The nomenclatures and parameter definitions used for describing the AS-WWTP in this work are given in Table 1. Moreover, four fundamental processes are considered: the growth and decay of biomass (heterotrophic and autotrophic), ammonification of organic nitrogen and the hydrolysis of particulate organics.

The reference model of biochemical reactions in the bioreactors is the activated sludge model no.1 (ASM1) [1], [4]–[8]. The success of this model has prompted widespread interest in biochemical modeling of wastewater in both academia and industry. The overall WWTP model consists of two main parts: the hydraulic model, which represents reactor behaviour, flow rates and recirculation; and the second primary component of WWTP model, is the activated sludge model, which portrays microbial growth, death and nutrient consumption. These models are necessarily approximations to the vast number of biological processes occurring in each bioreactor. Selection of the proper model allows adequate description of those processes most relevant to a particular WWTP. The development of accurate models is a prerequisite for applying model predictive control techniques for the whole process control and dynamic optimization.

The schematic of a BNR-ASWWTP design with basic control strategies is shown in Fig.1 using the *Johannesburg* configuration [7], [8] which consists of anaerobic, anoxic and aerobic zones and a secondary settler in a back-to-back scheme with multiple recycle streams [8]. To ensure that plug flow conditions prevail in the bioreactors, the basins are usually partitioned such that back-mixing is minimized. The constructional features and nomenclature of the process is detailed in Appendix C of [8].

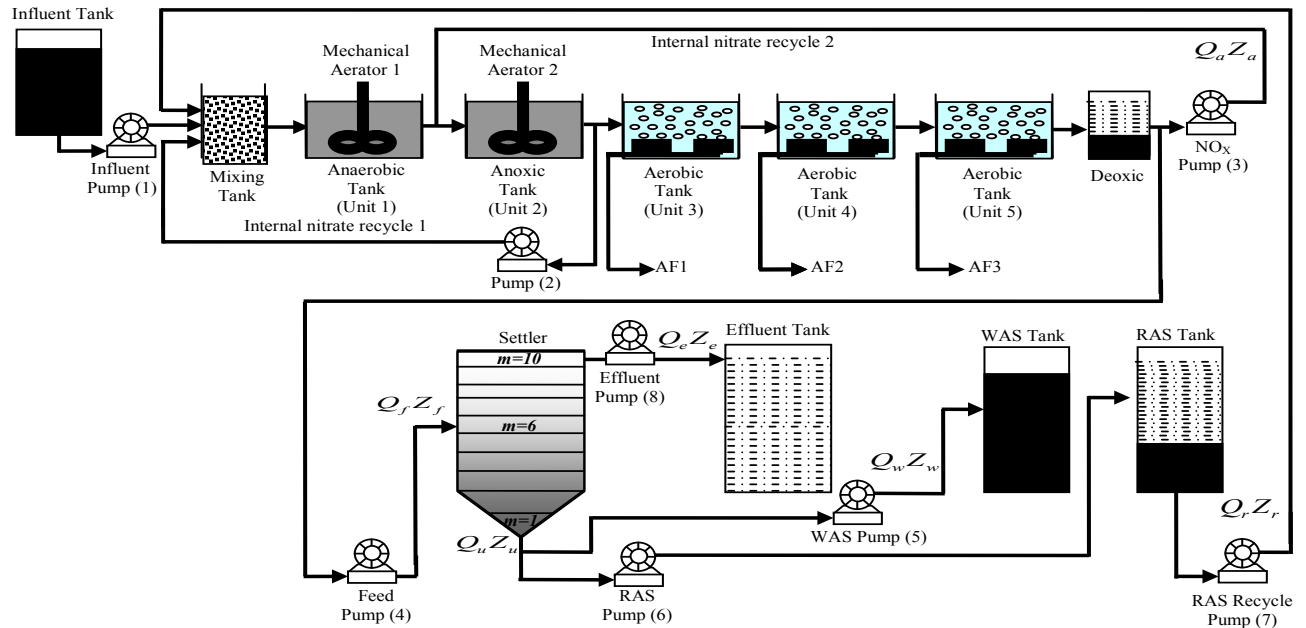


Figure 1. The schematic of the AS-WWTP process

Table 1. The AS-WWTP Nomenclatures and Parameter Definitions

Parameters	Definition	Parameters	Definition
$S_I^*$	Soluble inert organic matter	COD	Chemical oxygen demand
$S_S^*$	Readily biodegradable substrate	BOD	Biochemical oxygen demand
$X_I^*$	Particulate inert organic matter	IQ	Influent (inf) quality
$X_S^*$	Slowly biodegradable substrate	EQ	Effluent (e) quality
$X_{BH}^*$	Active heterotrophic biomass	QIN	Influent flow rate
$X_{BA}^*$	Active autotrophic biomass	F_M_R	Food-to-microorganisms ratio
$X_P^*$	Particulate products arising from biomass decay	Ntotal	Total nitrogen
$S_O^*$	Soluble oxygen	AF1,AF2,AF3	Aeration control points for the aerated reactors
$S_{NO}^*$	Nitrate and nitrite nitrogen	Qa1	Internal recycled nitrate (IRN) flow rates
$S_{NH}^*$	Ammonia and ammonium nitrogen	Qa2	External recycled nitrate (ERN) flow rates
$S_{ND}^*$	Soluble biodegradable organic nitrogen	Qf	Feed flow rates
$X_{ND}^*$	Particulate biodegradable organic nitrogen	Qw	Waste activated sludge (WAS) flow rate
$S_{ALK}^*$	Alkalinity	Qr	Recycled activated sludge (RAS) flow rates
TSS	Total soluble solids	Qe	Effluent flow rate
MA1, MA2	Mechanical aerators of the anaerobic and anoxic reactors	Qu	Sludge under flow rates
MLVSS	Mixed liquor volatile suspended solids	KLa	Mass transfer coefficient of the aerated reactors
IRN	Internal recycled nitrate	ERN	External recycled nitrate
Zf	Feed concentration	Ze	Effluent concentration
Zu	Settler underflow concentration	Zw	Waste activated sludge (WAS) concentration
Zr	Recycled activated sludge (RAS) concentration	PE	Pumping energy
AE	Aeration energy	DO	Dissolved oxygen
Za1	Internal recycled nitrate (IRN) concentration	Za2	External recycled nitrate (ERN) concentration

**Note:** (i) The numerical values of 1, 2, 3, 4, and 5 in front of each parameter correspond to the parameter description in the anaerobic, anoxic and the three aerated reactors respectively.  
(ii) The *inf* and *E* (and sometimes *e*) refers to influent and effluent respectively.  
(iii) Other parameters are introduced and defined as they are needed.  
(iv) Notations with asterisk (\*) are the state variables

The basic mode of operation of the AS-WWTP process is as follows. Organic wastewater is introduced into a reactor where an aerobic bacterial culture is maintained in suspension. The reactor contents are referred to as the mixed liquor. In the reactor, the bacterial culture converts the organic content of the wastewater into cell tissue. The aerobic environment in the reactor is achieved by the use of diffused or mechanical aeration, which also serve to maintain the liquor in a completely mixed regime. After a specified period of time, the mixture of new cells and old cells is passed into a settling tank where the cells are separated from the treated wastewater. A portion of the settled cells is recycled to maintain the desired concentration of organisms in the reactor, and a portion is wasted.

### 3. The Neural Network Identification Scheme and Validation Algorithms

#### 3.1. Formulation of the Neural Network Model Identification Problem

The method of representing dynamical systems by vector difference or differential equations is well established in systems and control theories [7], [8], [10], [11]. Assuming that a  $p$ -input  $q$ -output discrete-time nonlinear multivariable system at time  $k$  with disturbance  $\tilde{d}(k)$  can be represented by the following Nonlinear AutoRegressive Moving Average with eXogenous inputs (NARMAX) model:

$$\left. \begin{aligned} Y(k) &= J[Y(k-1), \dots, Y(k-n), \\ &U(k-d), \dots, U(k-d-m)] + \tilde{d}(k) \end{aligned} \right\} \quad (1)$$

where  $J(\bullet, \bullet)$  is a nonlinear function of its arguments, and  $[U(k-d), \dots, U(k-d-m)]$  are the past input vector,  $[Y(k-1), \dots, Y(k-n)]$  are the past output vector,  $Y(k)$  is the current output,  $m$  and  $n$  are the number of past inputs and outputs respectively that define the order of the system, and  $d$  is time delay. The predictor form of (1) based on the information up to time  $k-1$  can be expressed in the following compact form as:

$$\hat{Y}(k | k-1, \theta(k)) = J[\varphi(k, \theta(k)), \theta^T(k)] \quad (2)$$

where

$\varphi(k, \theta(k)) = [Y(k-1), \dots, Y(k-n), U(k-d), \dots, U(k-d-m), \varepsilon(k-1, \theta(k)), \dots, \varepsilon(k-n, \theta(k))]^T$  is the regression (state) vector,  $\theta(k)$  is an unknown

parameter vector which must be selected such that  $\hat{Y}(k | \theta(k)) \approx Y(k)$ ,  $\varepsilon(k, \theta(k))$  is the error between (1) and (2) defined as

$$\varepsilon(k, \theta(k)) = Y(k) - \hat{Y}(k | \theta(k)) \quad (3)$$

where  $k-1$  in  $\hat{Y}(k | k-1, \theta(k))$  of (2) is henceforth omitted for notational convenience. Note that  $\varepsilon(k, \theta(k))$  is the same order and dimension as  $\hat{Y}(k | \theta(k))$ .

Now, let  $\Theta$  be a set of parameter vectors which contain a set of vectors such that:

$$\Theta: \theta(k) \in \mathbb{Q}_\theta \subset \mathbb{R}^V \rightarrow \hat{\theta}(k) \quad (4)$$

where  $\mathbb{Q}_\theta$  is some subset of  $\mathbb{R}^V$  where the search for  $\hat{\theta}(k)$  is carried out;  $V$  is the dimension of  $\theta(k)$ ;  $\hat{\theta}(k)$  is the desired vector which minimizes the error in (3) and is contained in the set of vectors  $\Theta = \{\theta_1(k), \dots, \theta_\tau(k)\}$ ;  $\theta_1(k), \dots, \theta_\tau(k)$  are distinct values of  $\theta(k)$ ; and  $\tau = 1, 2, \dots, \max_{iter}$  is the number of iterations required to determine the  $\hat{\theta}(k)$  from the vectors in  $\Theta$ .

Let a set of  $N$  input-output data pair obtained from prior system operation over  $NT$  period of time be defined:

$$Z^N = \{U(1), Y(1), \dots, U(N), Y(N)\}, \quad N = 1, 2, \dots \quad (5)$$

where  $T$  is the sampling time of the system outputs. Then, the minimization of (3) can be stated as follows:

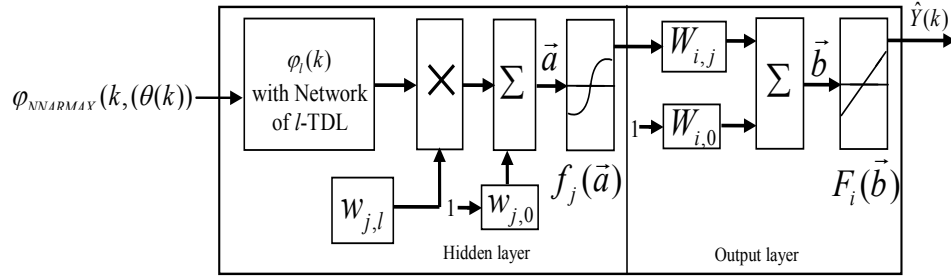
$$\hat{\theta}(k) = \arg \min_{\theta(k)} J(Z^N, \varphi(k, \theta(k)), \theta(k)) \quad (6)$$

where  $J(Z^N, \varphi(k, \theta(k)), \theta(k))$  is formulated as a total square error (TSE) type cost function which can be stated as:

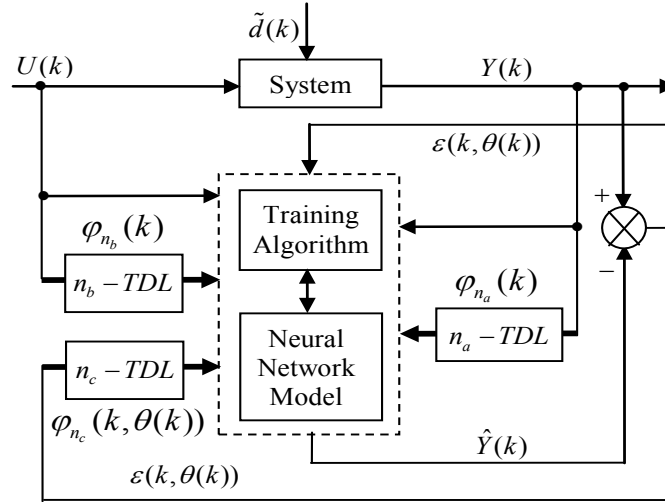
$$J(Z^N, \varphi(k, \theta(k)), \theta(k)) = \frac{1}{2N} \sum_{l=1}^N [\varepsilon(l, \theta(k))]^2 \quad (7)$$

The inclusion of  $\theta(k)$  as an argument in  $\varphi(k, \theta(k))$  is to account for the desired model  $\hat{\theta}(k)$  dependency on  $\tilde{d}(k)$ . Thus, given as initial random value of  $\theta(k)$ ,  $m$ ,  $n$  and (5), the system identification problem reduces to the minimization of (6) to obtain  $\hat{\theta}(k)$ . For notational convenience,  $J(\theta(k))$  shall henceforth be used instead of  $J(Z^N, \varphi(k, \theta(k)), \theta(k))$ .

### 3.2. Neural Network Identification Scheme



**Figure 2.** Architecture of the dynamic feedforward NN (DFNN) model



**Figure 3.** NN model identification based on the teacher-forcing method

The minimization of (6) is approached here by considering  $\hat{\theta}(k)$  as the desired model of network and having the DFNN architecture shown in Fig. 2. The proposed NN model identification scheme based on the teacher-forcing method is illustrated in Fig. 3. Note that the “Neural Network Model” shown in Fig. 3 is the DFNN shown in Fig. 2. The inputs to NN of Fig. 3 are  $\phi_{l_m}(k) = [U(k-d), \dots, U(k-d-m)]$ ,  $\phi_{l_n}(k) = [Y(k-1), \dots, Y(k-n)]^T$  and  $\phi_{l_e}(k, \theta(k)) = [\varepsilon(k-1, \theta(k)), \dots, \varepsilon(k-n, \theta(k))]^T$  which are concatenated into  $\phi_l(k, \theta(k))$  as shown in Fig. 2. The output of the NN model of Fig. 3 in terms of the network parameters of Fig. 2 is given as:

$$\hat{Y}(k | \hat{\theta}(k)) = F_i \left( \sum_{j=1}^{n_h} W_{i,j} f_j(\vec{a}) + W_{i,0} \right) \quad (8)$$

$$\vec{a} = \sum_{l=1}^{n_\phi} w_{j,l} \phi_l(k, \theta(k)) + w_{j,0}$$

where  $n_h$  and  $n_\phi$  are the number of hidden neurons and number of regressors respectively;  $i$  is the number of outputs,  $w_{j,l}$  and  $W_{i,j}$  are the hidden and output weights respectively;  $w_{j,0}$  and  $W_{i,0}$  are the hidden and output

biases;  $F_i(\vec{b})$  is a linear activation function for the output layer and  $f_j(\vec{a})$  is an hyperbolic tangent activation function for the hidden layer defined here as:

$$f_j(\vec{a}) = 1 - \frac{2}{e^{2\vec{a}} + 1} \quad (9)$$

Bias is a weight acting on the input and clamped to 1. Here,  $\hat{\theta}(k)$  is a collection of all network weights and biases in (8) in term of the matrices  $\mathbf{W} = \{w_{j,l} \ w_{j,0}\}$  and  $\mathbf{W} = \{W_{i,j} \ W_{i,0}\}$ . Equation (8) is here referred to as NN NARMAX (NNARMAX) model predictor for simplicity.

Note that  $\tilde{d}(k)$  in (1) is unknown but is estimated here as a covariance noise matrix,  $\Gamma[\theta(k)] = \mathbf{E}[\tilde{d}^T(k)\tilde{d}(k)]$ . Using  $\Gamma[\theta(k)]$ , Equation (7) can be rewritten as [5], [6]:

$$J(\theta(k)) = \frac{1}{2N} \left( \sum_{l=1}^N \varepsilon[l, \theta(k)]^T \Gamma^{-1}[\theta(k)] \varepsilon[l, \theta(k)] + \theta^T(k) D \theta(k) \right) \quad (10)$$

where the second term in (10) is the regularization (weight

decay) term [31] which has been introduced to reduce modeling errors, improve the robustness and performance of the two proposed training algorithms.  $D = \alpha_d I = [\alpha_h \ \alpha_o] I$  is a penalty norm and also removes ill-conditioning, where  $I$  is an identity matrix,  $\alpha_h$  and  $\alpha_o$  are the weight decay values for the input-to-hidden and hidden-to-output layers respectively. Note that both  $\hat{\Gamma}^{(j)}[\theta(k)]$  and  $D$  are adjusted simultaneously during network training with  $\theta(k)$  and are used to update  $\hat{\theta}(k)$  iteratively. The algorithm for estimating the covariance noise matrix and updating  $\hat{\theta}(k)$  is summarized in Table 2. Note that this algorithm is implemented at each sampling instant until  $\hat{\Gamma}^{(j)}[\theta(k)]$  has reduced significantly as in step 7).

**Table 2.** Iterative algorithm for estimating the covariance noise matrix

- 
- 1) Given initial network weights  $\theta(k) = \theta^{(0)}(k)$   
and  $j = j_{\max}$ .
  - 2) For  $k = 1$  to Number of Samples ( $N$ ), Do,
  - 3) Initialize  $\Gamma^{(0)}[\theta(k)] = I$ , Do,
  - 4) Set  $j = 1$
  - 5) Train the network for  $\tau$  iterations with a training algorithm  
using  $\Gamma^{(j-1)}[\theta(k)]$  to obtain  
 $\hat{\theta}(k) \leftarrow \theta_\tau(k) \triangleq \theta^{(j)}(k)$ .
  - 6) Estimate the covariance matrix for the noise using  
 $\hat{\Gamma}^{(j)}[\theta(k)] = \frac{1}{2N} \sum_{l=1}^N \varepsilon^T[\theta^{(j)}(k)] \varepsilon[\theta^{(j)}(k)]$
  - 7) If  $\hat{\Gamma}^{(j)}[\theta(k)] < esp$ , where  $esp$  is a convergence  
criteria.  
Set  $j \leftarrow j + 1$  and Go To Step 4).  
Else, set  $\hat{\theta}(k) = \theta^{(j)}(k)$  and End Set  $j$ .
  - 8) End For  $k$ .
- 

### 3.3. Formulation of the Neural Network-Based ARLS Algorithm

Unlike the BP which is a steepest descent algorithm, the ARLS and MLMA algorithms proposed here are based on the Gauss-Newton method with the typical updating rule [7],

[8], [10], [11]:

$$\hat{\theta}(k) = \theta_\tau(k) + \Delta\theta_\tau(k) \quad (11)$$

where

$$\Delta\theta_\tau(k) = -R[\theta_\tau(k)]^{-1} G[\theta_\tau(k)] \quad (12)$$

$\theta_\tau(k)$  denotes the value of  $\theta(k)$  at the current iterate  $\tau$ ,  $\Delta\theta_\tau(k)$  is the search direction,  $G[\theta_\tau(k)]$  and  $R[\theta_\tau(k)]$  are the Jacobian (or gradient matrix) and the Gauss-Newton Hessian matrices evaluated at  $\theta(k) = \theta_\tau(k)$ .

As mentioned earlier, due to the model  $\theta(k)$  dependency on the regression vector  $\varphi(k, \theta(k))$ , the NNARMAX model predictor depends on a *posteriori* error estimate using the feedback as shown in Fig. 2. Suppose that the derivative of the network outputs with respect to  $\theta(k)$  evaluated at  $\theta(k) = \theta_\tau(k)$  is given as [8]:

$$\psi[k, \theta_\tau(k)] = \frac{d\hat{Y}(k | \theta(k))}{d\theta(k)} \quad (13)$$

The derivative of (13) is carried out in a BP fashion for the input-to-hidden layer and for the hidden-to-output layer respectively for the two-layer DFNN of Fig. 2. Thus, the derivative of the NNARMAX model predictor can be expressed as

$$\psi[k, \theta_\tau(k)] = \left. \begin{aligned} & \frac{\partial \hat{Y}(k | \theta(k))}{\partial \theta(k)} \\ & - \frac{\partial \hat{Y}(k | \theta(k))}{\partial \varepsilon(k-1, \theta(k))} \frac{\partial \hat{Y}(k-1 | \theta(k))}{\partial \theta(k)} \\ & - \dots - \frac{\partial \hat{Y}(k | \theta(k))}{\partial \varepsilon(k-n_c, \theta(k))} \frac{\partial \hat{Y}(k-n_c | \theta(k))}{\partial \theta(k)} \end{aligned} \right\} \quad (14)$$

Thus, Equation (14) can be expressed equivalently as

$$\psi[l, \theta(k)] = \left. \begin{aligned} & \frac{d\hat{Y}(k | \theta(k))}{d\theta(k)} - C_1(k) \psi[k-1, \theta(k)] - \\ & \dots - C_n(k) \psi[k-n, \theta(k)] \end{aligned} \right\} \quad (15)$$

By letting  $C(k, z^{-1}) = I + C_1(k)z^{-1} + \dots + C_n(k)z^{-n_c}$ , then (15) can be reduced to the following form

$$\psi[k, \theta(k)] = \frac{1}{C(k, z^{-1})} \frac{d\hat{Y}(k | \theta(k))}{d\theta(k)} \quad (16)$$

As it can be seen from (16), the gradient is calculated by filtering the partial derivatives with the time-varying filter  $1/C(k, z^{-1})$  which depends on the prediction error based on the predicted output. Equation (16) is the only component

that actually impedes the implementation of the NN training algorithms depending on its computation.

Due to the feedback signals, the NNARMAX model predictor may be unstable if the system to be identified is not stable since the roots of (16) may, in general, not lie within the unit circle. The approach proposed here to iteratively ensure that the predictor becomes stable is summarized in the algorithm of Table 3. Thus, this algorithm ensures that roots of  $C(k, z^{-1})$  lies within the unit circle before the weights are updated by the training algorithm proposed in the next sub-section.

**Table 3.** An algorithm for placing the roots of the time-varying filter of the NNARMAX model predictor within the unit circle for stability

1) Given network weights  $\theta(k) = \theta^{(0)}(k)$ , time-varying filter

$$C(k, z^{-1}) = C^{(0)}(k, z^{-1}) \text{ and regression vector } \varphi(k, \theta(k))$$

2) Compute the roots of  $C(k, z^{-1})$  as  $C_{Roots}(k, z^{-1})$  and

$$\text{length of } C_{Roots}(k, z^{-1}) \text{ as } l_{CRoots}.$$

3) Compute the absolute value of

$$C_{Roots}(k, z^{-1}) = \text{abs}(C_{Roots}(k, z^{-1}))$$

4) For  $i = 1$  to  $l_{CRoots}$ ,

$$\text{if } \text{abs}(C_{Roots}^{(i)}(k, z^{-1})) > 1$$

$$C_{Roots}^{(i)}(k, z^{-1}) = \frac{1}{C_{Roots}^{(i)}(k, z^{-1})}$$

End if, End for

5) Compute the  $C(k, z^{-1})$  using the real root from Step 4).

### 3.3.1. The Adaptive Recursive Least Squares (ARLS) Algorithm

The proposed ARLS algorithm is derived from (11) with the assumptions that: 1) new input-output data pair is added to  $Z^N$  progressively in a first-in first-out fashion into a sliding window, 2)  $\hat{\theta}(k)$  is updated after a complete sweep through  $Z^N$ , and 3) all  $Z^N$  is repeated  $\tau$  times. Thus, Equation (10) can be expressed as [8]:

$$J(\theta(k)) = \frac{1}{2N} \sum_{l=1}^N \left( \pi^{N-l} \varepsilon^T[l, \theta_\tau(k)] \Gamma^{-1}[\theta(k)] \varepsilon[l, \theta_\tau(k)] + \theta^T(k) D \theta(k) \right) \quad (17)$$

$\pi \in (0, 1)$  is the exponential forgetting and resetting parameter for discarding old information as new data is acquired online and progressively added to the set  $Z^N$ .

Assuming that  $\theta(k-1)$  minimized (17) at time  $k-1$ ; then using (17), the updating rule for the proposed ARLS algorithm can be expressed from (11) as:

$$\hat{\theta}(k) = \theta_\tau(k|k-1) - R[\theta_\tau(k|k-1)]^{-1} G[\theta_\tau(k|k-1)] \quad (18)$$

where  $G[\theta_\tau(k)]$  and  $R[\theta_\tau(k)]$  given respectively as:

$$\begin{aligned} G[\theta_\tau(k)] &= -\frac{1}{N} \left( \psi[l, \theta_\tau(k-1)] \cdot \Gamma^{-1}[\theta(k)] \cdot \varepsilon[l, \theta_\tau(k-1)] + D \theta_\tau(k-1) \right) \\ R[\theta_\tau(k)] &= R[l, \theta_\tau(k-1)] + \frac{1}{N} \left( \psi^T[l, \theta_\tau(k)] \cdot \Gamma^{-1}[\theta(k)] \cdot \psi[l, \theta_\tau(k)] - R[l, \theta_\tau(k-1)] + D \right) \end{aligned} \quad (19)$$

where  $\psi[l, \theta_\tau(k)]$  is computed according to (16).

In order to avoid the inversion of  $R[\theta_\tau(k)]$ , Equation (19) is first computed as a covariance matrix estimate,  $P(k)$ , as

$$P(k) = \frac{1}{N} R[l, \theta_\tau(k)]^{-1}, \quad l = 1, 2, \dots, N \quad (20)$$

Then, by using the following matrix inversion lemma:

$$[A + BCD]^{-1} = A^{-1} - A^{-1}B[DA^{-1}B + C^{-1}]^{-1}DA^{-1}$$

By setting  $A = \pi R[l, \theta_\tau(k)]^{-1}$ ,  $B = D^{-1}$  and  $C = 1$ , Equation (20) can also be expressed equivalently as

$$P(k) = \frac{1}{\pi} P(k-1) - \Lambda(k) \psi^T[l, \theta_\tau(k-1)] P(k-1) + \beta I - \delta' P^2(k-1) \quad (21)$$

where  $\Lambda(k)$  is the adaptation factor given by

$$\Lambda(k) = \frac{\alpha P(k-1) \psi[l, \theta_\tau(k-1)]}{\left[ \Gamma^{-1}[\theta(k)] + \psi^T[l, \theta_\tau(k-1)] P(k-1) \psi[l, \theta_\tau(k-1)] \right]}$$

and  $I$  is an identity matrix of appropriate dimension,  $\alpha$ ,  $\beta$ ,  $\delta'$  and  $\pi$  are four design parameters are selected such that the following conditions are satisfied [7], [8]:

$$\left. \begin{aligned} 0 < \gamma < \alpha < 1, \quad \beta > 0, \quad \delta' > 0, \\ (\gamma - \alpha)^2 + 4\beta\delta' < (1 - \alpha)^2 \end{aligned} \right\} \quad (22)$$

where  $\alpha \in [0.1, 0.5]$  in  $\Lambda(k)$  adjusts the gain of the (21),  $\delta' \in [0, 0.01]$  is a small constant that is inversely

related to the maximum eigenvalue of  $P(k)$ ,  $\pi \in [0.9, 0.99]$  is the exponential forgetting factor which is selected such that  $\gamma \triangleq \frac{1-\pi}{\pi}$  and  $\beta \in [0, 0.01]$  is a small constant which is related to the minimum  $\bar{e}_{\min}$  and maximum  $\bar{e}_{\max}$  eigenvalues of (21) given respectively as [7], [8]:

$$\left. \begin{aligned} \bar{e}_{\min} &= ((\alpha - \gamma)/2\delta') \left( -1 + \sqrt{1 + (4\beta\delta'/(\alpha - \gamma)^2)} \right) \\ \bar{e}_{\max} &= (\gamma/2\delta') \left( 1 + \sqrt{1 + (4\beta\delta'/\gamma^2)} \right) \end{aligned} \right\} \quad (23)$$

The values of  $\alpha$ ,  $\beta$ ,  $\delta'$  and  $\pi$  in (22) is selected such that  $\bar{e}_{\max}/\bar{e}_{\min} \approx 10^4$  while the initial value of  $P(k)$ , that is  $P(0)$ , is selected such that  $\bar{e}_{\min} I < P(0) < \bar{e}_{\max} I$  [8].

Thus, the ARLS algorithm updates based on the exponential forgetting and resetting method is given from (18) as

$$\left. \begin{aligned} \hat{\theta}(k) &= \theta_{\tau}(k-1) \\ &+ \Lambda(k) \Gamma^{-1} [\theta(k)] [Y(k) - \hat{Y}(k | \theta_{\tau}(k-1))] \end{aligned} \right\} \quad (24)$$

where the second term in (20) is  $\Delta\theta_{\tau}(k)$ . Note that after  $\hat{\theta}(k)$  has been obtained, the algorithm of Table 2 is implemented the conditions in Step 7) of the Table 2 algorithm is satisfied.

### 3.4. Proposed Validation Methods for the Trained NNARMAX Model

Network validations are performed to assess to what extend the trained network has approximated and capture the operation of the underlying dynamics of a system and as measure of how well the model being investigated will perform when deployed for the actual process [10]–[12].

The first test involves the comparison of the predicted outputs with the true training data and the evaluation of their corresponding errors using (3).

The second validation test is the Akaike's final prediction error (AFPE) estimate [9]–[12] based on the weight decay parameter  $D$  in (10). A smaller value of the AFPE estimate indicates that the identified model approximately captures all the dynamics of the underlying system and can be presented with new data from the real process. Evaluating the  $\varepsilon(k, \hat{\theta}(k))$  portion of (3) using the trained network with  $\theta(k) = \hat{\theta}(k)$  and taking the expectation  $\mathbf{E}\{J(Z^N, \hat{\theta}(k))\}$  with respect to  $\varphi(k)$  and  $\tilde{d}(k)$  leads to the following AFPE estimate [9]–[12]:

$$\hat{F}(Z^N \hat{\theta}(k)) \approx \frac{N + p_a}{N - p_b} J(Z^N \hat{\theta}(k)) + \gamma \quad (25)$$

where

$$p_a = \text{tr} \left\{ V(\hat{\theta}(k)) [V(\hat{\theta}(k)) + D]^{-1} V(\hat{\theta}(k)) [V(\hat{\theta}(k)) + D]^{-1} \right\}$$

and  $\text{tr}\{\cdot\}$  is the trace of its arguments and it is computed as the sum of the diagonal elements of its arguments,  $p_b = \text{tr}\{V(\hat{\theta}^*) [V(\hat{\theta}^*) + (1/N)D]^{-1}\}$  and  $\gamma$  is a positive quantity that improves the accuracy of the estimate and can be computed according to the following expression:

$$\gamma = \frac{\hat{\theta}(k)^T D}{N^2} \left( R[\hat{\theta}(k)] + \frac{D}{N} \right)^{-1} R[\hat{\theta}(k)] \left( R[\hat{\theta}(k)] + \frac{D}{N} \right)^{-1} D \hat{\theta}(k)$$

The third method is the K-step ahead predictions [10] where the outputs of the trained network are compared to the unscaled output training data. The K-step ahead predictor follows directly from (8) and for  $\varphi(k) = \hat{\varphi}(k+K)$  and

$\theta(k) = \hat{\theta}(k)$ , takes the following form:

$$\hat{Y}((k+K) | k, \hat{\theta}) = \hat{J}(Z^N, \hat{\varphi}(k+K), \hat{\theta}(k)) \quad (26)$$

where

$$\begin{aligned} \hat{\varphi}(k+K) &= [U((k+K-1) | \hat{\theta}), \dots, U((k+K-m) | \hat{\theta}), \\ &\hat{Y}((k+K-1) | \hat{\theta}), \dots, \hat{Y}((k+K+1-\min(k,n)) | \hat{\theta}), \\ &Y((k+K-1) | \hat{\theta}), \dots, Y((k+K-\max(n-k,0)) | \hat{\theta})]^T \end{aligned}$$

The mean value of the K-step ahead prediction error (MVPE) between the predicted output and the actual training data set is computed as follows:

$$\begin{aligned} MVPE &= \\ &= \text{mean} \left( \sum_{k=m+K}^N \frac{Y(k) - \hat{Y}((k+K) | k, \hat{\theta})}{Y(k)} \right) \times 100\% \quad (27) \end{aligned}$$

where  $Y(k)$  corresponds to the unscaled output training data and  $\hat{Y}((k+K) | k, \hat{\theta})$  the K-step ahead predictor output.

## 4. Formulation of the Neural Network-Based Online Prediction Problem for the Influent Characteristics

### 4.1. Selection of the Manipulated Inputs and Controlled Outputs of the Influent Characteristics Problem

The online determination of the influent characteristics prior to its arrival at the wastewater treatment environment can give information about the control requirements. This idea for predicting the influent characteristics, as illustrated in Fig.4 based on ANN, is by evaluating the influent quality index (IQ), chemical oxygen demand (COD<sub>inf</sub>), biochemical oxygen demand (BOD<sub>inf</sub>), total Kjeldahl nitrogen (TKN) and the food-to-microorganisms ratio



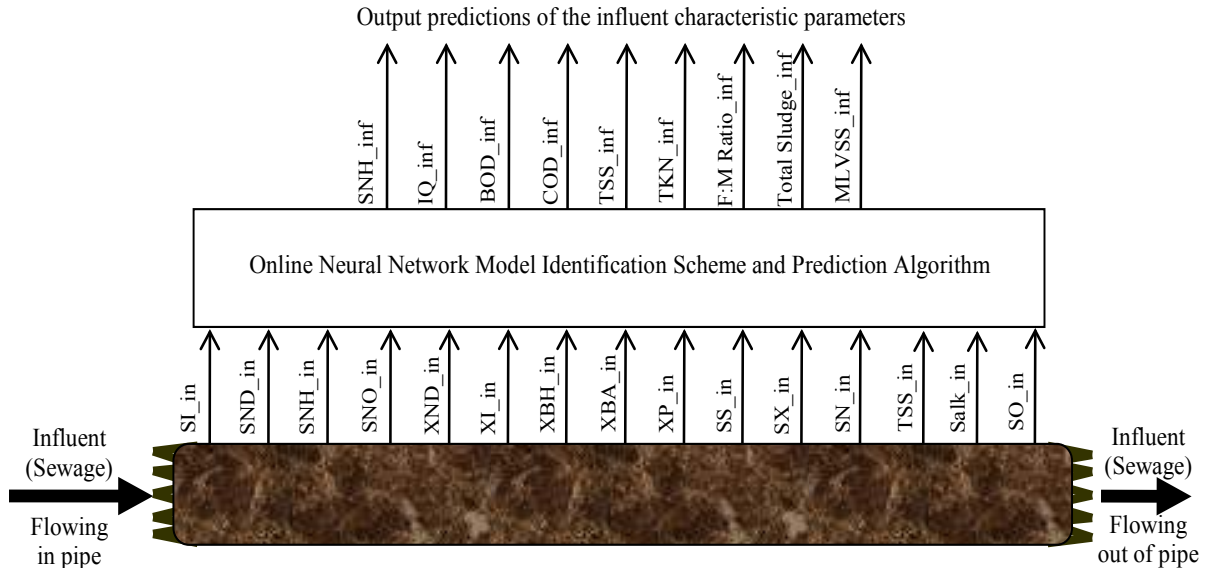
(F\_M\_R) as predicted outputs using TSS\_inf, XS\_inf, XI\_inf, XBH\_inf, XBA\_inf, XP\_inf, SNH\_inf, SND\_inf, XND\_inf, SS\_inf, SI\_inf, SO\_inf, SNO\_inf, SN\_inf and Salk\_inf as inputs. Based on the COST 624 standards, the expected values for predicted outputs should be for TKN\_inf = 10 g.m<sup>-3</sup>, BOD\_inf = 2 g.m<sup>-3</sup>, COD\_inf = 48.2 g.m<sup>-3</sup>, Influent quality (IQ) = 42000 kg.d<sup>-1</sup>, ammonia and ammonium nitrogen (SNH\_pinf) = 4 g.m<sup>-3</sup>, total sludge = 18692.5 kg, MLVSS = 1130 g.m<sup>-3</sup>, TSS = 211.3 g.m<sup>-3</sup> and food-to-microorganisms ratio (F:M ratio) = 0.2 mg.BOD/mg. MLVSS. These predicted outputs forms the decision parameters and the starting point for the design of the adaptive self-organizing fuzzy logic controller and its implementation in our next study.

## 4.2. Formulation of the AS-WWTP Influent Characteristics Model Identification and Prediction Problem

### 4.2.1. Statement of the AS-WWTP Influent Characteristics Neural Network Model Identification Problem

The activated sludge wastewater treatment plant model defined by the benchmark simulation model no. 1 (BSM1) is described by eight coupled nonlinear differential equations

given in Appendix A. The BSM1 model consist of thirteen states defined in Table 1 but they are redefined here for the incoming influent (with  $I$  and subscript  $in$  and  $inf$  for inputs and outputs respectively) as follows:  $I_{SI\_in}$ ,  $I_{SS\_in}$ ,  $I_{XI\_in}$ ,  $I_{XS\_in}$ ,  $I_{XBH\_in}$ ,  $I_{XBA\_in}$ ,  $I_{XP\_in}$ ,  $I_{SO\_in}$ ,  $I_{SNO\_in}$ ,  $I_{SNH\_in}$ ,  $I_{SND\_in}$ ,  $I_{XND\_in}$ ,  $I_{SALK\_in}$ ,  $I_{SN\_in}$  and  $I_{TSS\_in}$  for the inputs; while  $I_{SI\_inf}$ ,  $I_{SS\_inf}$ ,  $I_{XI\_inf}$ ,  $I_{XS\_inf}$ ,  $I_{XBH\_inf}$ ,  $I_{XBA\_inf}$ ,  $I_{XP\_inf}$ ,  $I_{SO\_inf}$ ,  $I_{SNO\_inf}$ ,  $I_{SNH\_inf}$ ,  $I_{SND\_inf}$ ,  $I_{XND\_inf}$ ,  $I_{SALK\_inf}$ ,  $I_{TSS\_inf}$ ,  $I_{SN\_inf}$ ,  $I_{total\_sludge\_inf}$ ,  $I_{F:M\_ratio\_inf}$ ,  $I_{MLVSS\_inf}$ ,  $I_{IQ\_inf}(k)$ ,  $I_{BOD\_inf}$ ,  $I_{COD\_inf}$  and  $I_{TKN\_inf}$  are the outputs. Out of thirteen states, only four states are measurable namely:  $I_{SS\_in}$  (readily biodegradable substrate),  $I_{XBH\_in}$  (active heterotrophic biomass),  $I_{SO\_in}$  (oxygen) and  $I_{SNO\_in}$  (nitrate and nitrite nitrogen).



**Figure 4.** The proposed scheme for the neural network-based NNAMARX model identification and prediction of the incoming influent characteristics for the AS-WWTP process

Thus, from the discussions so far, the measured inputs that influence the behaviour of the influent characteristics of the AS-WWTP process shown in Fig. 5 are:

$$U(k) = \left[ \begin{array}{c} I_{SI\_in}(k) \quad I_{SND\_in}(k) \quad I_{SNH\_in}(k) \quad I_{SNO\_in}(k) \quad I_{XND\_e}(k) \\ I_{XI\_e}(k) \quad I_{XBH\_e}(k) \quad I_{XBA\_in}(k) \quad I_{XP\_in}(k) \quad I_{SS\_in}(k) \\ I_{XS\_in}(k) \quad I_{SN\_in}(k), \quad I_{SO\_in}(k), \quad I_{TSS\_in}(k), \quad I_{SALK\_in}(k) \end{array} \right]^T \quad (28)$$



$$\varphi_{n_b}(k) = \left[ \begin{array}{ccc} I_{SI\_in}(k-n_b) & I_{SND\_in}(k-n_b) & I_{SNH\_in}(k-n_b) \\ I_{XBH\_in}(k-n_b) & I_{XBA\_in}(k-n_b) & I_{XP\_in}(k-n_b) \\ I_{TSS\_in}(k-n_b) & I_{TSS2\_in}(k-n_b) & I_{TSS3\_in}(k-n_b) \\ I_{TSS7\_in}(k-n_b) & I_{TSS8\_in}(k-n_b) & I_{TSS9\_in}(k-n_b) \\ I_{SNO\_in}(k-n_b) & I_{SO\_in}(k-n_b) & I_{TSS\_in}(k-n_b) \end{array} \right]^T \quad (31)$$

$$\varphi_{n_c}(k) = \left[ \begin{array}{cc} \varepsilon_{I_{SNH\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{IQ\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{BOD\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{COD\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{TSS\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{TKN\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{SI\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{SND\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{XND\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{XI\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{XBH\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{XBA\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{XP\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{SS\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{total\_sludge\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{XS\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{SN\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{MLVSS\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{SALK\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{SNO\_inf}}(k-n_c, \theta(k)) \\ \varepsilon_{I_{SO\_inf}}(k-n_c, \theta(k)) & \varepsilon_{I_{F:M\_ratio\_inf}}(k-n_c, \theta(k)) \end{array} \right]^T \quad (32)$$

$$\varphi_{NNARMAX}(k, \theta(k)) = \left[ \varphi_{n_a}(k) \quad \varphi_{n_b}(k) \quad \varphi_{n_c}(k, \theta(k)) \right] \quad (33)$$

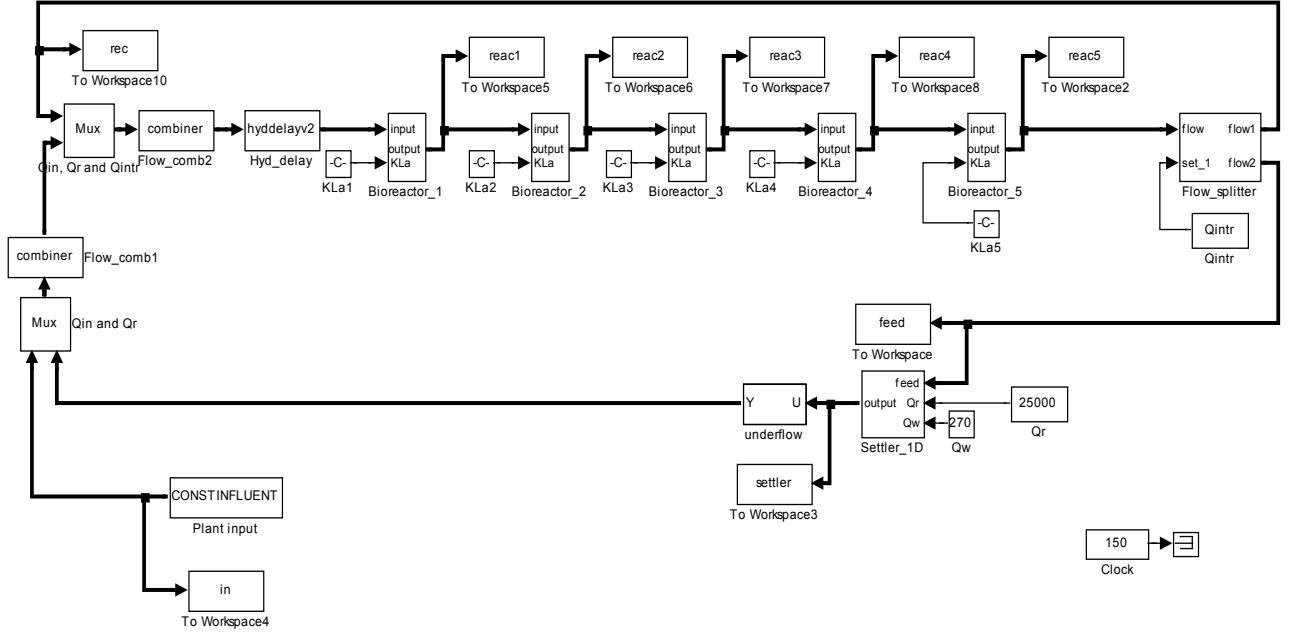
The outputs of the neural network for the AS-WWTP process are the predicted values of the fourteen states together with the nine output parameters that are used to determine the influent characteristics. Thus, resulting in 21 parameters to be predicted at each sampling instant given by:

$$\hat{Y}(k) = \left[ \begin{array}{cccc} \hat{y}_{SNH\_inf}(k) & \hat{y}_{IQ\_inf}(k) & \hat{y}_{BOD\_inf}(k) & \hat{y}_{COD\_inf}(k) \\ \hat{y}_{TSS\_inf}(k) & \hat{y}_{TKN\_inf}(k) & \hat{y}_{SI\_inf}(k) & \hat{y}_{SND\_inf}(k) \\ \hat{y}_{XND\_inf}(k) & \hat{y}_{XI\_inf}(k) & \hat{y}_{XBH\_inf}(k) & \hat{y}_{XBA\_inf}(k) \\ \hat{y}_{XP\_inf}(k) & \hat{y}_{SS\_inf}(k) & \hat{y}_{XS\_inf}(k) & \hat{y}_{SN\_inf}(k) \\ \hat{y}_{SNO\_inf}(k) & \hat{y}_{SO\_inf}(k) & \hat{y}_{SALK\_inf}(k) & \hat{y}_{MLVSS\_inf}(k) \\ \hat{y}_{total\_sludge\_inf} & \hat{y}_{F:M\_ratio\_inf}(k) & & \end{array} \right]^T \quad (34)$$

Since disturbances play important roles in the evaluation of controller performances, three influent disturbance data are defined for the three different weather conditions, namely: dry-weather data, rain weather data, and storm weather data. The data for these three influent disturbances are provided by the European COST Actions for evaluating controller performances [4]–[6], [8]. In this study, the dry weather influent data is used in order to measure how well the trained neural network mimic the dynamics of the AS-WWTP process to meet the control requirement specified above. The dry weather data contains two weeks of influent data at 15 minutes sampling interval. Although, disturbances  $\tilde{d}(k)$  affecting the AS-WWTP are

incorporated into dry-weather data provided by the COST Action Group, additional sinusoidal disturbances with non-smooth nonlinearities are introduced to further investigate the closed-loop performances based on an updated neural network model at each sampling time instants.

#### 4.2.2. Experiment with the BSM1 for AS-WWTP Process Neural Network Training Data Acquisition



**Figure 6.** Open-loop steady-state benchmark simulation model No.1 (BSM1) with constant influent.

For the efficient control of the activated sludge wastewater treatment plant (AS-WWTP) using neural network, a neural network (NN) model of the AS-WWTP process is needed which requires that the NN be trained with dynamic data obtained from the AS-WWTP process. In other to obtain dynamic data for the NN training, the validated and generally accepted COST Actions 624 benchmark simulation model no. 1 (BSM1) is implemented and simulated using MATLAB and Simulink as shown in Fig. 6. The BSM1 process model for the AS-WWTP process is given in Appendix A.

A two-step simulation procedure defined in the COST Actions simulation benchmark [4]–[6], [8] is used in this study. The first step is the steady state simulation using the constant influent flow (CONSTINFLUENT) for 150 days as shown and implemented in Fig. 6. Note that each simulation sample period indicated by the “Clock” of the AS-WWTP Simulink model in Fig. 6 corresponds to one day. In the second step, starting from the steady state solution obtained with the CONSTINFLUENT data and using the dry-weather influent weather data (DRYINFLUENT) as inputs, the AS-WWTP process is then simulated for 14 days using the same Simulink model of Fig. 6 but by replacing the CONSTINFLUENT influent data with the DRYINFLUENT influent data. This second simulation generates 1345 dynamic data in which is used for NN training while the 130 first day dry-weather data samples provided by the COST Actions 624 and 682 is used for the trained NN validation.

#### 4.2.3. The Incremental or Online Back-Propagation (INCBP) Algorithm

In order to investigate the performance of the ARLS, the so-called incremental (or online) back-propagation (INCBP) algorithm is used for this purpose. The incremental or online back-propagation (INCBP) algorithm was originally proposed by [13] which has been modified in [7], [8] is used in this paper. The incremental back-propagation (INCBP) algorithm is easily derived by setting the covariance matrix  $P(k) = \mu I$  on the left hand side of (20) in Section 3.3.1 under the formulation of the ARLS algorithm; that is:

$$\mu_{\tau} I = \frac{1}{k} \sum_{i=1}^k R[\iota, \theta_{\tau}(k)]^{-1} \quad (35)$$

where  $\mu$  is the step size and  $I$  is an identity matrix of appropriate dimension. Next, the basic back-propagation given from [10] as:

$$\hat{\theta}(k) = \theta_{\tau}(k) - \mu_{\tau} \frac{dJ[\theta(k)]}{d\theta(k)} \bigg|_{\theta(k)=\theta_{\tau}(k)} \quad (36)$$

is used to update the algorithm in (35). Finally, all that is required is to specify a suitable step size  $\mu$  and carry out the recursive computation of the gradient given by (36).

#### 4.2.4. Scaling the Training Data and Rescaling the Trained Network that Models the AS-WWTP Process

Due to the fact the input and outputs of a process may, in general, have different physical units and magnitudes; the scaling of all signals to the same variance is necessary to prevent signals of largest magnitudes from dominating the identified model. Moreover, scaling improves the numerical robustness of the training algorithm, leads to faster convergence and gives better models. The training data are scaled to unit variance using their mean values and standard deviations according to the following equations [7], [8]:

$$\left. \begin{aligned} U^{(s)}(k) &= \frac{U(k) - \bar{U}(k)}{\sigma_{U(k)}} \\ Y^{(s)}(k) &= \frac{Y(k) - \bar{Y}(k)}{\sigma_{Y(k)}} \end{aligned} \right\} \quad (37)$$

where  $\bar{U}(k)$ ,  $\bar{Y}(k)$  and  $\sigma_{U(k)}$ ,  $\sigma_{Y(k)}$  are the mean and standard deviation of the input and output training data pair; and  $U^{(s)}(k)$  and  $Y^{(s)}(k)$  are the scaled inputs and outputs respectively. Also, after the network training, the joint weights are rescaled according to the expression

$$\hat{Y}(k, \hat{\theta}(k)) = \hat{Y}(k, \hat{\theta}(k))\sigma_{Y(k)} + \bar{Y}(k) \quad (38)$$

so that the trained network can work with other unscaled validation data and test data not used for training. However, for notational convenience,  $U(k) = U^{(s)}(k)$  and  $Y(k) = Y^{(s)}(k)$  shall be used.

#### 4.2.5. Training the Neural Network that Models the Biological Reactors of the AS-WWTP Process

The NN input vector to the neural network (NN) is the NNARMAX model regression vector  $\varphi_{NNARMAX}(k, \theta(k))$  defined by (33). The input  $\varphi_{n_c}(k, \theta(k))$ , that is the initial error estimates  $\varepsilon(k, \theta(k))$  given by (32), is not known in advance and it is initialized to small positive random matrix of dimension  $n_c$  by  $n_c$ . The outputs of the NN are the predicted values of  $\hat{Y}(k)$  given by (34).

For assessing the convergence performance, the network was trained for  $\tau = 50$  epochs (number of iterations) with the following selected parameters:  $p = 15$ ,  $q = 22$ ,  $n_a = 2$ ,  $n_b = 2$ ,  $n_c = 2$ ,  $n_\phi = 118$  (NNARMAX),  $n_h = 5$ ,  $n_o = 22$ ,  $\alpha_h = 1e-6$  and  $\alpha_o = 1e-5$ . The details of these parameters are discussed in Section 3; where  $p$  and  $q$  are the number of inputs and outputs of the system,  $n_a$ ,  $n_b$  and  $n_c$  are the orders of the regressors in terms of the past values,  $n_\phi$  is the total number of regressors (that is, the total number of inputs to the network),

$n_h$  and  $n_o$  are the number of hidden and output layers neurons, and  $\alpha_h$  and  $\alpha_o$  are the hidden and output layers weight decay terms. The four design parameters for adaptive recursive least squares (ARLS) algorithm defined in (22) are selected to be:  $\alpha=0.5$ ,  $\beta=5e-3$ ,  $\delta'=1e-5$  and  $\pi=0.99$  resulting to  $\gamma=0.0101$ . The initial values for  $\bar{e}_{min}$  and  $\bar{e}_{max}$  in (23) are equal to 0.0102 and 1.0106e+3 respectively and were evaluated using (23). Thus, the ratio of  $\bar{e}_{min}/\bar{e}_{max}$  from (23) is 9.9018e+4 which imply that the parameters are well selected. Also,  $\mu = 1e-3$  is selected to initialize the INCBP algorithm given in (36).

The 1345 dry-weather training data is first scaled using equation (37) and the network is trained for  $\tau = 50$  epochs using the proposed adaptive recursive least squares (ARLS) and the incremental back-propagation (INCBP) algorithms proposed in Sections 3.3 and 4.2.3. After network training, the trained network is again rescaled respectively according to (38), so that the resulting network can work or be used with unscaled AS-WWTP data. Although, the convergence curves of the INCBP and the ARLS algorithms for 50 epochs each are not shown but the minimum performance indexes for both algorithms are given in the third rows of Tables 4(a), (b) and (c). As one can observe from these Tables, the ARLS has smaller performance index when compared to the INCBP which is an indication of good convergence property of the ARLS at the expense of higher computation time when compared the small computation time used by the INCBP for 50 epochs as evident in the first rows of Tables 4(a), (b) and (c).

The total square error (TSE) discussed in subsection 3.1, for the network trained with the INCBP and the ARLS algorithms are given in the second rows of Table 4(a), (b) and (c). Again, the ARLS algorithm also has smaller mean square errors and minimum performance indices when compared to the INCBP algorithm. The small values of the total square error (TSE) and the minimum performance indices indicate that ARLS performs better than the INCBP for the same number of iterations (epochs). These small errors suggest that the ARLS model approximates better the AS-WWTP process giving smaller errors than the INCBP model.

#### 4.3. Validation of the Trained NNARMAX Model for the Prediction of the Influent Characteristics for the AS-WWTP Process

According to the discussion on network validation in Section 3.4, a trained network can be used to model a process once it is validated and accepted, that is, the network demonstrates its ability to predict correctly both the data that were used for its training and other data that were not used during training. The network trained by the INCBP and the proposed ARLS algorithms has been validated with three different methods by the use of scaled and unscaled training data as well as with the 130 dry-weather data reserved for the validation of the trained network for the AS-WWTP process.

**Table 4(a).** Influent characteristics and the influent constrained parameter predictions

	SNH_inf		IQ_inf		BOD_inf	
	INCBP	ARLS	INCBP	ARLS	INCBP	ARLS
Computation time for model identification (sec)	4.3212e+00	1.4399e+01	4.2588e+00	1.5584 e+01	4.2276e+00	1.4477 e+01
Total square error (TSE)	4.8675 e-01	9.2769e-03	4.1666 e+03	1.7782e-02	8.5000e+00	6.4933 e-02
Minimum performance index	9.1389 e-02	4.4021e-03	8.0045 e-01	1.9263 e-02	4.6210 e-01	3.9978 e-03
Mean error of one step ahead prediction of training data	1.8955e-03	1.2487e-03	3.0655 e-01	1.288 e-03	2.3676 e-01	3.9064 e-04
Mean error of one step prediction of test data	3.7378 e-02	9.6692 e-06	2.8115 e-01	4.3028 e-05	4.0457 e-02	3.1597 e-05
Mean value of 5-step ahead prediction error	1.4303e+01	6.155 e-01	1.4685 e+01	1.8433 e-01	7.3485 e-01	3.6822 e-02
Akaike's final prediction error (AFPE) estimate	3.8102e+00	2.9882 e-02	2.0893 e05	1.3746 e+00	3.8142 e+02	9.5570 e+00

**Table 4(b).** Influent characteristics and the influent constrained parameter predictions

	COD_inf		TSS_inf		TKN_inf	
	INCBP	ARLS	INCBP	ARLS	INCBP	ARLS
Computation time for model identification (sec)	4.1964e+00	1.4477e+01	4.2588 e+00	1.4898 e+01	4.2276e+00	1.4399 e+01
Total square error (TSE)	4.8177e+00	2.0131e-02	8.6363 e+00	1.6281 e-02	1.4522 e-01	2.2347 e-03
Minimum performance index	9.0511 e-02	6.5630e-03	5.8515 e-02	3.8735 e-03	1.0556 e-01	7.0020 e-03
Mean error of one step ahead prediction of training data	3.9020 e-03	4.8326e-04	6.9311 e-03	1.9931 e-04	9.0802 e-03	4.1416 e-04
Mean error of one step prediction of test data	9.1919 e-03	1.7951e-03	1.6923 e-02	1.0176 e-03	6.1617 e-03	6.2926 e-04
Mean value of 5-step ahead prediction error	8.4582 e-01	7.6351e-03	3.4175 e+00	5.3885 e-02	4.2677 e+00	7.2354 e-01
Akaike's final prediction error (AFPE) estimate	3.8100e+03	4.4859e-02	2.4248 e+03	1.8178 e+01	2.5590 e+01	2.6081 e-01

**Table 4(c).** Influent characteristics and the influent constrained parameter predictions

	Total_Sludge_inf		F_to_M Ratio_inf		MLVSS_inf	
	INCBP	ARLS	INCBP	ARLS	INCBP	ARLS
Computation time for model identification (sec)	4.1340e+00	1.4617 e+01	4.0560e+00	1.4539 e+01	4.5864e+00	1.5678 e+01
Total square error (TSE)	1.0880e+6	5.0840e+01	1.5034e-03	1.8605e-05	1.5519e+02	2.2785e-01
Minimum performance index	1.0381 e-02	4.8658 e-05	6.9110 e-02	1.0419 e-02	6.5752 e-02	1.3233 e-02
Mean error of one step ahead prediction of training data	4.8346 e-03	3.7691 e-05	1.3029 e-02	9.6307 e-04	1.1192 e-02	2.6519 e-04
Mean error of one step prediction of test data	3.0786 e-03	1.3797 e-04	2.9108 e-02	9.6846 e-04	1.2093 e-02	1.4434 e-03
Mean value of 5-step ahead prediction error	4.9577 e+0	9.0834 e-03	9.6695 e+0	1.1355 e-02	2.1424 e-01	8.9861 e-03
Akaike's final prediction error (AFPE) estimate	6.9192 e+04	2.1402 e+00	2.0216 e+03	2.4250 e-01	3.4902 e-2	5.9632 e-04

#### 4.3.1. Validation by the One-Step Ahead Predictions Simulation

In the one-step ahead prediction method, the errors obtained from one-step ahead output predictions of the trained network are assessed. In Fig. 7(a)–(i) the graphs for the one-step ahead predictions of the scaled training data (blue -) against the trained network output predictions (red --\*) using the neural network models trained by INCBP and ARLS algorithms respectively are shown for 50 epochs.

The mean value of the one-step ahead prediction errors are given in the fourth rows of Table 4(a), (b) and (c) respectively. It can be seen in the figures that the network predictions of the training data closely match the original training data. Although, the scaled training data prediction errors by both algorithms are small, the ARLS algorithm appears to have a much smaller error when compared to the INCBP algorithm as shown in the fourth rows of Table 4(a), (b) and (c). These small one-step ahead prediction errors are indications that the networks trained using the ARLS

captures and approximate the nonlinear dynamics of the five reactors of the AS-WWTP process to a high degree of accuracy. This is further justified by the small mean values of the TSE obtained for the networks trained using the proposed ARLS algorithms for the process as shown in the second rows of Table 4(a), (b) and (c).

Furthermore, the suitability of the INCBP and the proposed ARLS algorithms for neural network model identification for use in the real AS-WWTP industrial environment is investigated by validating the trained network with the 130 unscaled dynamic data obtained for dry-weather as provided by the COST Action Group. Graphs of the trained network predictions (red --\*) of the validation (test) data with the actual validation data (blue -) using the INCBP and the proposed ARLS algorithms are shown in Fig. 8(a)–(i) for the five reactors of the AS-WWTP process based on the selected process parameters. The almost identical prediction of these data proves the effectiveness of the proposed approaches. The prediction accuracies of the unscaled test data by the networks trained using the INCBP and the proposed ARLS algorithm evaluated by the computed mean prediction errors shown in the fifth rows of Table 4(a), (b) and (c). Again, one can observe that although the validation data prediction errors obtained by both algorithms are small, the validation data predictions errors obtained with the model trained by the proposed ARLS algorithm appears much smaller when compared to those obtained by the model trained using the INCBP algorithm. These predictions of the unscaled validation data given in Fig. 8(a)–(i) as well as the mean value of the one step ahead validation (test) prediction errors in the fifth rows of Tables 4(a), (b) and (c) verifies the neural network ability to model accurately the dynamics of the five reactors of the AS-WWTP process based on the dry-weather influent data using the proposed ARLS training algorithm.

#### 4.3.2. $K$ -Step Ahead Prediction Simulations for the AS-WWTP Process

The results of the  $K$ -step ahead output predictions (red --\*) using the  $K$ -step ahead prediction validation method discussed in Section 3.4 for 5-step ahead output predictions ( $K = 5$ ) compared with the unscaled training data (blue -) are shown in Fig. 9(a) to Fig. 9(i) for the networks trained using the INCBP and the proposed ARLS. Again, the value  $K = 5$  is chosen since it is a typical value used in most model predictive control (MPC) applications. The comparison of the 5-step ahead output predictions performance by the network trained using the INCBP and the proposed ARLS algorithms indicate the superiority of the proposed ARLS over the so-called INCBP algorithm.

The computation of the mean value of the  $K$ -step ahead prediction error (MVPE) using (27) is given in the sixth rows of Tables 4(a), (b) and (c) by the network trained using INCBP and the proposed ARLS algorithms respectively. The small mean values of the 5-step ahead prediction error (MVPE) are indications that the trained network

approximates the dynamics of the five reactors of the AS-WWTP process to a high degree of accuracy with the networks of both algorithms but with the network based on the ARLS algorithm giving much smaller distant prediction errors.

#### 4.3.3. Akaike's Final Prediction Error (AFPE) Estimates for the AS-WWTP Process

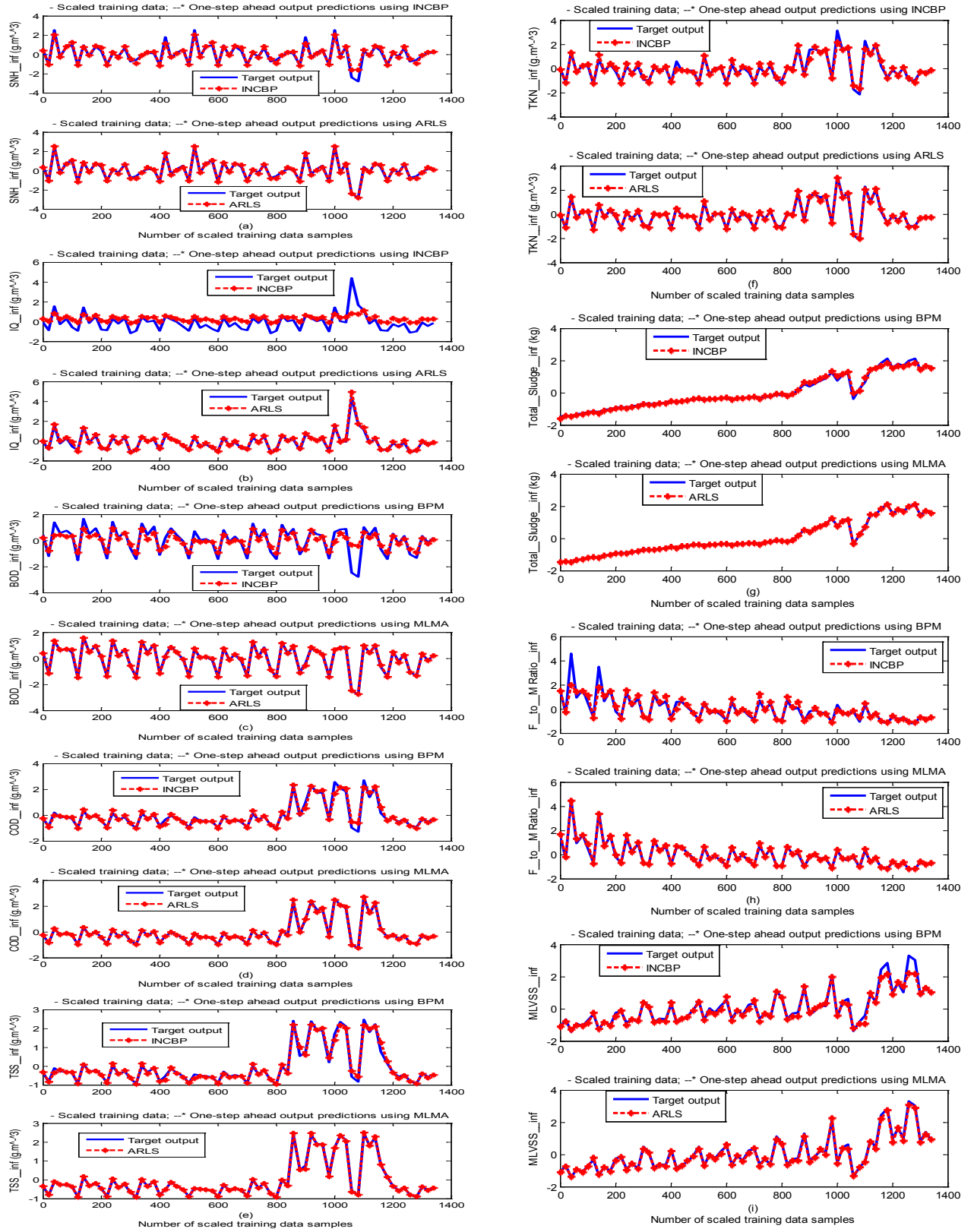
The implementation of the AFPE algorithm discussed in Section 3.4 and defined by (25) for the regularized criterion for the network trained using the INCBP and the proposed ARLS algorithms with multiple weight decay gives their respective AFPE estimates which are defined in the seventh rows of Tables 4(a), (b) and (c) respectively. These relatively small values of the AFPE estimate indicate that the trained networks capture the underlying dynamics of the aerobic reactor of the AS-WWTP and that the network is not over-trained [9]. This in turn implies that optimal network parameters have been selected including the weight decay parameters. Again, the results of the AFPE estimates computed for the networks trained using the proposed ARLS algorithm are much smaller when compared to those obtained using INCBP algorithm.

#### 4.3.4. Influent Characteristics Prediction Based on the CST Action Constrained Parameters for the Influent

In order to predict the influent characteristics of the incoming influent (sewage), the Simulink model of the AS-WWTP process shown in Fig. 6 has been simulated in closed-loop with  $\pm 30\%$  disturbances about the nominal values of all the input parameters distributed over 200 random samples in 200 simulations. However, due to space economy, on 10 simulations (Sim) results at an interval of 20 samples are shown in Table 5. Note that in Table 5, the last column shows the nominal values as published by COST Actions 624 and 628 [4]–[6].

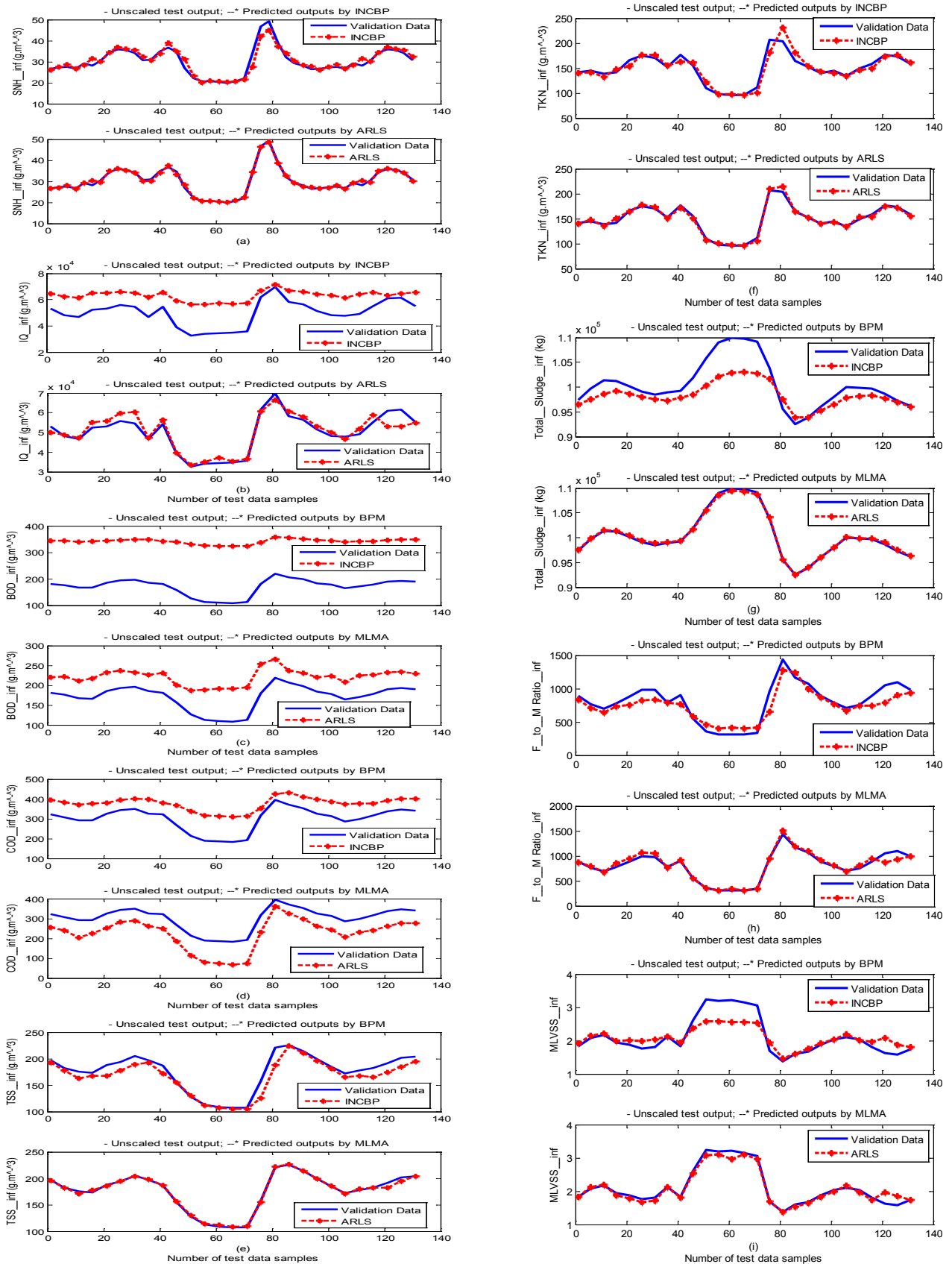
It can be seen from Table 5 that both the INCBP and the ARLS algorithms gives appreciable predictions of all the influent constrained parameters. A close study of Table 5 further reveals that the better prediction results are obtained using the NN models trained with ARLS. For clarity, the results of Table 5 is plot as shown in Fig. 10 where the prediction performances by the models based on INCBP and ARLS are compared to the COST standards. As it can be seen in Fig. 10, (a), (b), (c), (d) and (f) are completely out of phase when compared to COST standard for our simulation studies. However, the predictions in (e), (g), (h) and (i) fluctuates around and are within the ranges of the COST standards except for the MLVSS<sub>inf</sub> in (i) due to INCBP.

It is obvious that the incoming influent in the first three simulations (i.e. 0, 1 and 2) is of high toxic content and may be discarded while the incoming influent in the last three simulations (i.e. 7, 8 and 9) may be pretreated before passing it to the WWTP. The incoming influent from the third simulation up the third before the last (i.e. 2, 3, 4, 5, 6 and 7) can be channeled directly to the WWTP.

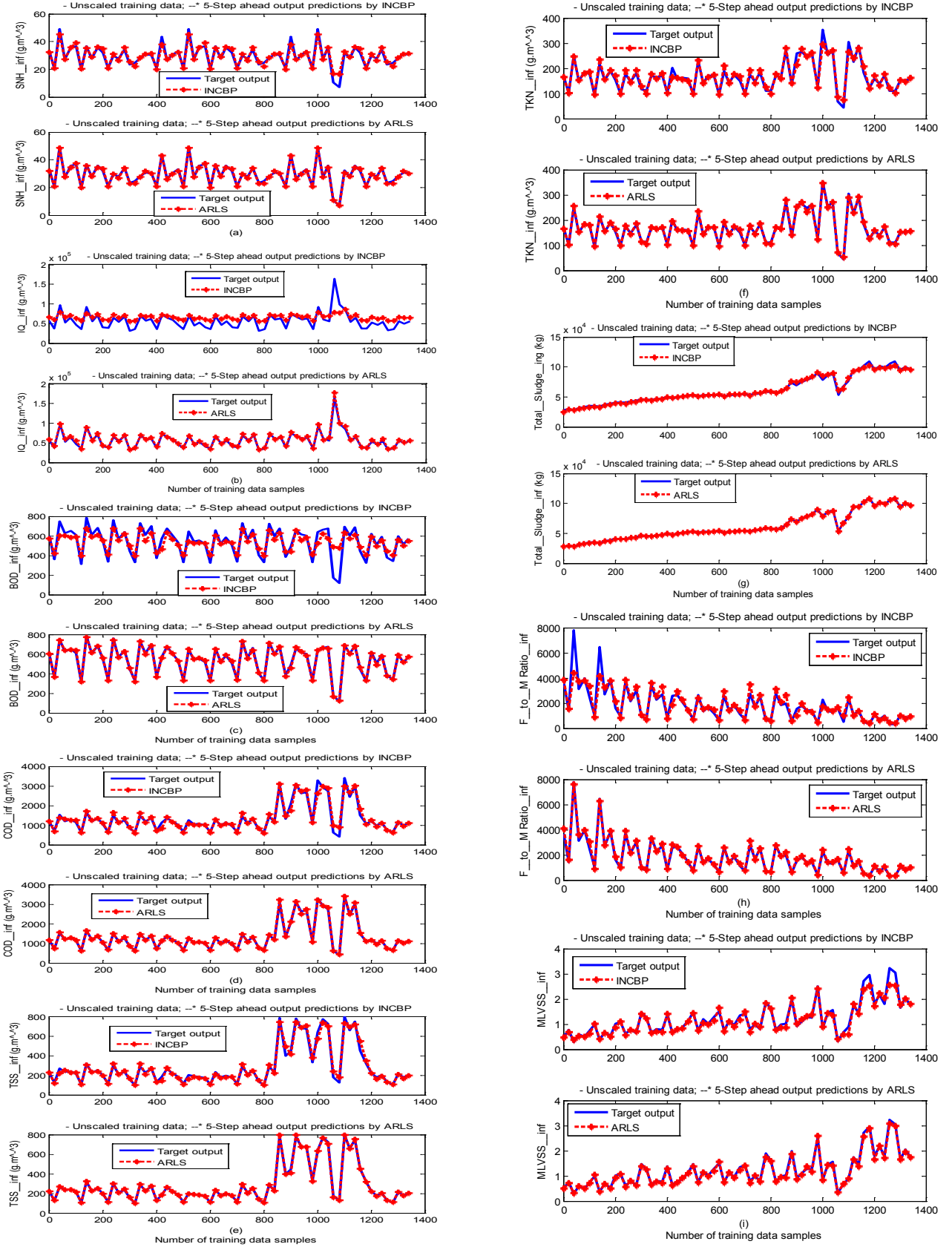


**Figure 7(a).** Comparison of the one-step ahead prediction of scaled training data by INCBP and ARLS: (a) SNH<sub>inf</sub>, (b) IQ<sub>inf</sub>, (c) BOD<sub>inf</sub>, (d) COD<sub>inf</sub>, (e) TSS<sub>inf</sub>, (f) TKN<sub>inf</sub>, (g) Total\_Sludge<sub>inf</sub>, (h) F<sub>to\_M</sub> Ratio<sub>inf</sub>, (i) MLVSS<sub>inf</sub>

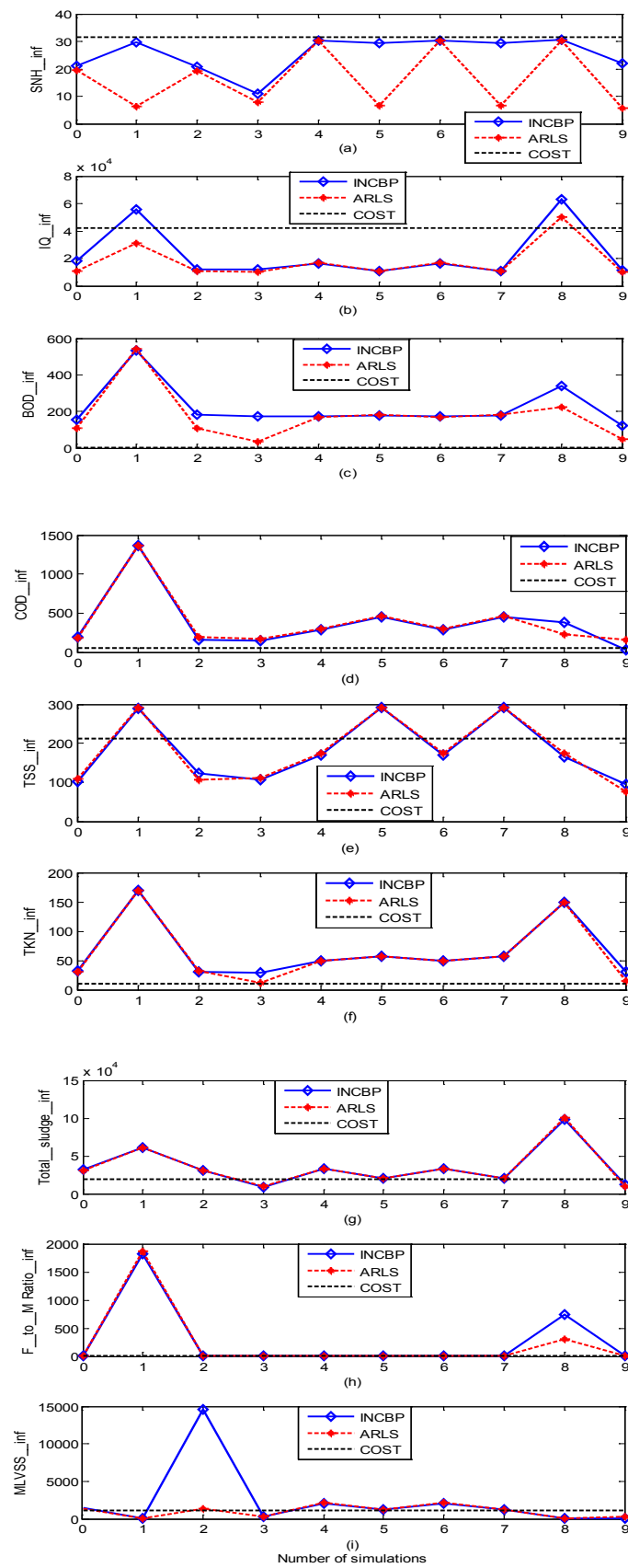




**Figure 8(a).** Comparison of the one-step ahead prediction of unscaled validation data by INCBP and ARLS: (a) SNH<sub>inf</sub>, (b) IQ<sub>inf</sub>, (c) BOD<sub>inf</sub>, (d) COD<sub>inf</sub>, (e) TSS<sub>inf</sub>, (f) TKN<sub>inf</sub>, (g) Total\_Sludge<sub>inf</sub>, (h) F<sub>to\_M</sub> Ratio<sub>inf</sub>, (i) MLVSS<sub>inf</sub>



**Figure 9(a).** Comparison of the five-step ahead prediction of unscaled training data by INCBP and ARLS: (a) SNH<sub>inf</sub>, (b) IQ<sub>inf</sub>, (c) BOD<sub>inf</sub>, (d) COD<sub>inf</sub>, (e) TSS<sub>inf</sub>, (f) TKN<sub>inf</sub>, (g) Total\_Sludge<sub>inf</sub>, (h) F<sub>to\_M</sub> Ratio<sub>inf</sub>, (i) MLVSS<sub>inf</sub>



**Figure 10.** Comparison of the constrained parameter predictions by models trained using INCBP and ARLS with the true COST Actions standards: (a)  $SNH\_inf$ , (b)  $IQ\_inf$ , (c)  $BOD\_inf$ , (d)  $COD\_inf$ , (e)  $TSS\_inf$ , (f)  $TKN\_inf$ , (g)  $Total\_Sludge\_inf$ , (h)  $F\_to\_M\ Ratio\_inf$ , (i)  $MLVSS\_inf$

**Table 5.** Mean values of the constrained parameters for the prediction of the influent characteristics in 10 simulation studies

		Sim 20	Sim 40	Sim 60	Sim 80	Sim 100	Sim 120	Sim 140	Sim 160	Sim 180	Sim 200	COST 624 & 628
SNH_inf	INCBP	2.1091e+01	2.9604e+01	2.0840e+01	1.1028e+01	3.0446e+01	2.9287e+01	3.0446e+01	2.9287e+01	3.0467e+01	2.2053e+01	31.56
	ARLS	1.9418e+01	6.3213e+00	1.9324e+01	7.8323e+00	3.0327e+01	6.6975e+00	3.0327e+01	6.6975e+00	3.0334e+01	5.6763e+00	
IQ_inf	INCBP	1.8204e+04	5.5395e+04	1.2118e+04	1.1863e+04	1.6264e+04	1.0469e+04	1.6264e+04	1.0469e+04	6.3032e+04	1.1138e+04	42,000
	ARLS	1.0886e+04	3.0918e+04	1.0904e+04	1.0276e+04	1.6629e+04	1.0570e+04	1.6629e+04	1.0570e+04	4.9912e+04	1.0379e+04	
BOD_inf	INCBP	1.5582e+02	5.3432e+02	1.8195e+02	1.7478e+02	1.7087e+02	1.7799e+02	1.7087e+02	1.7799e+02	3.4145e+02	1.2350e+02	2.0
	ARLS	1.0919e+02	5.4039e+02	1.0880e+02	3.4952e+01	1.6985e+02	1.8011e+02	1.6985e+02	1.8011e+02	2.2166e+02	4.7786e+01	
COD_inf	INCBP	1.8944e+02	1.3654e+03	1.5699e+02	1.4387e+02	2.8782e+02	4.5509e+02	2.8782e+02	4.5509e+02	3.7663e+02	2.7856e+01	48.2
	ARLS	1.8510e+02	1.3709e+03	1.8803e+02	1.6877e+02	3.0044e+02	4.5706e+02	3.0044e+02	4.5706e+02	2.2431e+02	1.5938e+02	
TSS_inf	INCBP	1.0242e+02	2.8937e+02	1.2275e+02	1.0553e+02	1.7005e+02	2.9127e+02	1.7005e+02	2.9127e+02	1.6560e+02	9.5121e+01	211.3
	ARLS	1.0817e+02	2.9260e+02	1.0706e+02	1.1094e+02	1.7454e+02	2.9278e+02	1.7454e+02	2.9278e+02	1.7489e+02	7.5171e+01	
TKN_inf	INCBP	3.1282e+01	1.6925e+02	3.0263e+01	2.8511e+01	4.9887e+01	5.6464e+01	4.9887e+01	5.6464e+01	1.4935e+02	3.0820e+01	10.0
	ARLS	3.0663e+01	1.6958e+02	3.1712e+01	1.1733e+01	4.9848e+01	5.6549e+01	4.9848e+01	5.6549e+01	1.4976e+02	1.4666e+01	
Total _sludge_inf	INCBP	3.1508e+04	6.1323e+04	3.1119e+04	8.6374e+03	3.2806e+04	2.0477e+04	3.3215e+04	2.0541e+04	9.8390e+04	1.1750e+04	18,692.5
	ARLS	3.0784e+04	6.1395e+04	3.0786e+04	1.0225e+04	3.3540e+04	2.0464e+04	3.3533e+04	2.0464e+04	1.0061e+05	1.0039e+04	
F_to_M Ratio_inf	INCBP	4.3475e-02	1.8253e+03	1.9136e-02	8.1025e-04	8.0780e-02	2.0607e-01	8.0780e-02	2.0607e-01	7.4674e+02	5.6454e-02	0.2
	ARLS	3.5497e-02	1.8644e+03	3.6656e-02	4.6316e-02	3.7079e-02	2.0688e-01	3.7079e-02	2.0688e-02	3.0288e+02	4.0663e-02	
MLVSS_inf	INCBP	1.4946e+03	1.1726e+00	1.4554e+04	2.7921e+02	2.1057e+03	1.1780e+03	2.1057e+03	1.1780e+03	2.0839e+00	9.2462e+01	1,130
	ARLS	1.3682e+03	1.1811e+00	1.3686e+03	2.3976e+02	2.1143e+03	1.1804e+03	2.1143e+03	1.1804e+03	2.1159e+00	2.4749e+02	

## 5. Conclusions

This paper presents a novel problem formulation for the prediction of influent characteristics together with the formulation of advanced online nonlinear adaptive recursive least squares (ARLS) model identification algorithm based on artificial neural networks for the nonlinear model identification and performance parameter prediction for AS-ASWWTP process management. In order to investigate the performance of the proposed ARLS algorithm, the incremental backpropagation (INCBP), which is also an online algorithm, is implemented and compared with proposed ARLS. The results from the application of these algorithms to the modelling and prediction of the influent characteristics as well as the validation results show that the neural network-based ARLS outperforms the INCBP algorithm with much smaller predictions error and good tracking and prediction abilities with an appreciable degree of accuracy. It is concluded that the proposed ARLS model identification algorithm can be used for the AS-ASWWTP process in an industrial environment.

From the view point of the authors, the nonlinear modeling and adaptive control of the AS-WWTP process is better formulated as a five-stage multivariable modeling and control problem with tight constraints because of its nonlinear and structural complexities. Thus, as a first step, the online prediction of the incoming influent (sewage) characteristics has been considered. The next stage is on the nonlinear modeling of the biological reactors followed by the nonlinear modeling of the secondary settler and clarifier. The

fourth stage is on the development and implementation of an intelligent self-organizing fuzzy logic decision controller (SOFLDC) for the complete AS-WWTP process. The fifth stage is on the development of a nonlinear adaptive model-based predictive control (NAMBP) algorithm for the adaptive control of the complete AS-WWTP by manipulating the pumps based on some decision parameters.

The next aspect of the work is on the dynamic modelling and nonlinear model identification of the multivariable NNARMAX model of the secondary settler and the clarifier as well as effluent tank to complete the modelling of the AS-WWTP process.

## Appendix

### Appendix A: AS-WWTP Process Model

As mentioned in above, the BSM1 model involves eight different chemical reactions ( $\rho_j$ ) incorporating thirteen different components [4]–[6], [8]. These components are classified into soluble components ( $S$ ) and particulate components ( $X$ ). The nomenclatures and parameter definitions used for describing the AS-WWTP in this work are given in Table 1. The Moreover, four fundamental processes are considered: the growth and decay of biomass (heterotrophic and autotrophic), ammonification of organic nitrogen and the hydrolysis of particulate organics. The typical schematic of the AS-WWTP is shown in Fig. 1.

The eight basic processes that are used to describe the biological behaviour of the AS-WWTP process are:

$j = 1$  : Aerobic growth of heterotrophs

$$\rho_1 = \mu_H \left( \frac{S_S}{K_S + S_S} \right) \left( \frac{S_O}{K_S + S_O} \right) X_{B,H} \quad (\text{A.1})$$

$j = 2$  : Anoxic growth of heterotrophs

$$\rho_2 = \mu_H \left( \frac{S_S}{K_S + S_S} \right) \left( \frac{K_{O,H}}{K_{O,H} + S_O} \right) \left\{ \times \left( \frac{S_{NO}}{K_{NO} + S_{NO}} \right) \eta_g X_{B,H} \right\} \quad (\text{A.2})$$

$j = 3$  : Aerobic growth of autotrophs

$$\rho_3 = \mu_H \left( \frac{S_{NH}}{K_{NH} + S_{NH}} \right) \left( \frac{S_O}{K_{O,A} + S_O} \right) X_{B,A} \quad (\text{A.3})$$

$j = 4$  : Decay of heterotrophs

$$\rho_4 = b_H X_{B,H} \quad (\text{A.4})$$

$j = 5$  : Decay of autotrophs

$$\rho_5 = b_A X_{B,A} \quad (\text{A.5})$$

$j = 6$  : Ammonification of soluble organic nitrogen

$$\rho_6 = k_a S_{ND} X_{B,H} \quad (\text{A.6})$$

$j = 7$  : Hydrolysis of entrapped organics

$$\rho_7 = k_h \frac{X_S / X_{B,H}}{K_X + (X_S / X_{B,H})} \times \left[ \left( \frac{S_O}{K_{O,H} + S_O} \right) + \eta_h \left( \frac{K_{O,H}}{K_{O,H} + S_O} \right) \left( \frac{S_{NO}}{K_{NO} + S_{NO}} \right) \right] X_{B,H} \quad (\text{A.7})$$

$j = 8$  : Hydrolysis of entrapped organic nitrogen

$$\rho_8 = k_h \frac{X_S / X_{B,H}}{K_X + (X_S / X_{B,H})} \times \left[ \left( \frac{S_O}{K_{O,H} + S_O} \right) + \eta_h \left( \frac{K_{O,H}}{K_{O,H} + S_O} \right) \left( \frac{S_{NO}}{K_{NO} + S_{NO}} \right) \right] \times (X_{B,H} (X_{ND} / X_S)) \quad (\text{A.8})$$

The observed thirteen conversion rates ( $r_i$ ) result from combinations of basic processes (A.1) to (A.8) as follows:

$$S_I(i=1) : r_1 = 0 \quad (\text{A.9})$$

$$S_S(i=2) : r_2 = -\frac{1}{Y_H} \rho_1 - \frac{1}{Y_H} \rho_2 + \rho_7 \quad (\text{A.10})$$

$$X_I(i=3) : r_3 = 0 \quad (\text{A.11})$$

$$X_S(i=4) : r_4 = (1 - f_P) \rho_4 + (1 - f_P) \rho_5 - \rho_7 \quad (\text{A.12})$$

$$X_{B,H}(i=5) : r_5 = \rho_1 + \rho_2 - \rho_4 \quad (\text{A.13})$$

$$X_{B,A}(i=6) : r_6 = \rho_3 - \rho_5 \quad (\text{A.14})$$

$$X_P(i=7) : r_7 = f_P \rho_4 + f_P \rho_5 \quad (\text{A.15})$$

$$S_O(i=8) : r_8 = -\frac{1-Y_H}{Y_H} \rho_1 - \frac{4.57-Y_A}{Y_A} \rho_3 \quad (\text{A.16})$$

$$S_{NO}(i=9) : r_9 = -\frac{1-Y_H}{2.86Y_H} \rho_2 + \frac{1}{Y_A} \rho_3 \quad (\text{A.17})$$

$$S_{NH}(i=10) : r_{10} = -i_{XB} \rho_1 - i_{XB} \rho_2 - \left( i_{XB} + \frac{1}{Y_A} \rho_3 \right) + \rho_6 \quad (\text{A.18})$$

$$S_{ND}(i=11) : r_{11} = -\rho_6 + \rho_8 \quad (\text{A.19})$$

$$X_{ND}(i=12) : r_{12} = (i_{XB} - f_P i_{XP}) \rho_4 + (i_{XB} - f_P i_{XP}) \rho_5 - \rho_8 \quad (\text{A.20})$$

$$S_{ALK}(i=13) : r_{13} = -\frac{i_{XB}}{14} \rho_1 + \left( \frac{1-Y_H}{14 \times 2.86Y_H} - \frac{i_{XB}}{14} \right) \rho_2 - \left( \frac{i_{XB}}{14} + \frac{1}{7Y_A} \right) \rho_3 + \frac{1}{14} \rho_6 \quad (\text{A.21})$$

The biological parameter values used in the BSM1 correspond approximately to a temperature of 15°C. The stiochiometric parameters are listed in Table A.1 and the kinetic parameters are listed in Table A.2.

**Table A.1.** Stiochiometric parameters with their units and values

Parameters	Unit	Value
$Y_A$	g cell COD formed.(g N oxidized) <sup>-1</sup>	0.24
$Y_H$	g cell COD formed.(g COD oxidized) <sup>-1</sup>	0.67
$f_P$	Dimensionless	0.08
$i_{XB}$	g N.(g COD) <sup>-1</sup> in biomass	0.08
$i_{XP}$	g N.(g COD) <sup>-1</sup> in particulate products	0.06

**Table A.2.** Kinetic parameters with their units and values

Parameters	Unit	Value
$\mu_H$	(day) <sup>-1</sup>	4.0
$K_S$	g COD.m <sup>-3</sup>	10.0
$K_{O,H}$	g (-COD).m <sup>-3</sup>	0.2
$K_{NO}$	g $NO_3^-$ N.m <sup>-3</sup>	0.5
$b_H$	(day) <sup>-1</sup>	0.3
$\eta_g$	Dimensionless	0.8
$\eta_h$	Dimensionless	0.8
$k_h$	g slowly biodegradable COD.(g cell COD.day) <sup>-1</sup>	3.0

$K_X$	g slowly biodegradable COD.(g cell COD) <sup>-1</sup>	0.1
$\mu_A$	(day) <sup>-1</sup>	0.5
$K_{NH}$	g $NH_3$ -N.m <sup>-3</sup>	1.0
$b_A$	(day) <sup>-1</sup>	0.05
$K_{O,A}$	G (COD).m <sup>-3</sup>	0.4
$k_a$	m <sup>-3</sup> .(g COD.day) <sup>-1</sup>	0.05

## Appendix B: Criteria for Evaluating and Assessing the Performances of the AS-WWTP Control

### Appendix B.1: Influent Quality (IQ)

As a check on the IQ calculation, an influent quality index (IQ) can be calculated by applying the above equations to the influent file but the BOD coefficient must be changed from 0.25 to 0.65. It is defined as:

$$I.Q. = \frac{1}{T \cdot 1000} \int_{t=7 \text{ days}}^{t=14 \text{ days}} \left( \begin{aligned} &B_{SS} \cdot SS_0(t) + B_{COD} \cdot COD_0(t) \\ &+ B_{NKj} \cdot S_{NKj,0}(t) + B_{NO} \cdot S_{NO,0}(t) \\ &+ B_{BOD} \cdot BOD_{5,0}(t) \end{aligned} \right) \times 3 * Q_{in\_tr}(t) dt \quad (B.1)$$

where the composition variables are calculated as follows:

$$S_{NKj,0} = S_{NH,0} + S_{ND,0} + X_{ND,0} + i_{XB}(X_{B,H,0} + X_{B,A,0}) + i_{XP}(X_{P,0} + X_{i,0}) \quad (B.2)$$

$$SS_0 = 0.75 \cdot (X_{S,0} + X_{I,0} + X_{B,H,0} + X_{B,A,0} + X_{P,0}) \quad (B.3)$$

$$BOD_0 = 0.65 \cdot (S_{S,0} + X_{S,0} + (1 - f_p) \cdot (X_{B,H,0} + X_{B,A,0})) \quad (B.4)$$

$$COD_0 = S_{S,0} + S_{I,0} + X_{S,0} + X_{I,0} + X_{B,H,0} + X_{B,A,0} + X_{P,0} \quad (B.5)$$

and the  $B_i$  above are weighting factors for the different types of pollution to convert them into pollution units ( $g \cdot m^{-3}$ ) and were chosen to reflect these calculated fractions as follows:  $B_{TSS} = 2$ ,  $B_{COD} = 1$ ,  $B_{NKj} = 20$ ,  $B_{NO} = 20$  and  $B_{BOD5} = 2$ .

### Appendix B.2: The Sludge production to be disposed (kg / d)

This is the sludge production,  $P_{sludge}$ , is calculated from the total solid flow from wastage and solid accumulated in the system over the period of time considered ( $t_f = 7$  for each weather file). The amount of solids in the system at time  $t$  is given by:

$$M(TSS_{system})(t) = M(TSS_{reactor})(t) + M(TSS_{settler})(t) \quad (B.6)$$

where  $M(TSS_{reactor})(t)$  is the amount of solids in the reactor given by:

$$M(TSS_{reactor})(t) = 0.75 \cdot \sum_{i=1}^{i=5} \left( \begin{aligned} &X_{S,i} + X_{I,i} + X_{B,H,i} \\ &+ X_{B,A,i} + X_{P,i} \end{aligned} \right) \cdot V_i \quad (B.7)$$

$M(TSS_{settler})(t)$  is the amount of solids in the settler given by:

$$M(TSS_{settler})(t) = 0.75 \sum_{j=1}^{j=7} \left( X_{S,j} + X_{I,j} + X_{B,H,j} + X_{B,A,j} + X_{P,j} \right) z_i A \quad (B.8)$$

$\Delta M(TSS)_{system}$  the change in system sludge mass from the end of day 7 to the end of day 14 given by:

$$\Delta M(TSS)_{system} = M(TSS_{system})_{14days} - M(TSS_{system})_{7days}$$

and  $M(TSS_{waste})(t)$  the amount of waste sludge is given by:

$$M(TSS_{waste})(t) = 0.75 \int_{t=7days}^{t=14days} \left( X_{S,w} + X_{I,w} + X_{B,H,w} + X_{B,A,w} + X_{P,w} \right) \cdot Q_w(t) dt \quad (B.9)$$

So that the total sludge to be disposed becomes:

$$P_{sludge} = \frac{1}{T} \left( \Delta M(TSS_{system})(t) + M(TSS_{waste})(t) \right) \quad (B.10)$$

### Appendix B.3: The Total Sludge production ( kg / d ):

The total sludge production takes into account the sludge to be disposed and the sludge lost to the weir and is calculated as follows:

$$P_{total\_sludge} = P_{sludge} + M(TSS_e) \quad (B.11)$$

where

$$M(TSS_e) = \frac{0.75}{T} \int_{t=7days}^{t=14days} \left( X_{S,e} + X_{I,e} + X_{B,H,e} + X_{B,A,e} + X_{P,e} \right) \cdot Q_e(t) dt$$

## REFERENCES

- 
- [1] M. Henze, P. Harremoës, J. Jansen and E. Arvin, "Wastewater Treatment," *Biological and Chemical Processes*, 2nd ed., Berlin: Springer Verlag, 1996.
  - [2] E. Arden and W. T. Lockett, "Experiments on the oxidation of sewage without the aid of filters", *J. Soc. Chem. Ind.*, vol. 33, pp. 523 – 539.
  - [3] F.R. Spellman, *Handbook of Water and Wastewater Treatment Plant Operations*. Boca Raton, Florida: CRC Press LLC, 2003.
  - [4] J.B. Coop. (2000, Sept.). The COST Simulation Benchmark: Description and Simulation Manual (a product of COST Actions 624 & 628). [Online]. Available: <http://www.ensic.inpl-nancy.fr/COSTWWTP/>.
  - [5] Working Groups of COST Actions 632 and 624. (2000, Sept.). IWA Task Group on Benchmarking of Control Strategies for WWTPs. [Online]. Available: <http://www.ensic.inpl-nancy.fr/benchmarkWWTP/Bsm1/Benchmark1.htm>.
  - [6] Working Groups of COST Actions 632 and 624. (Apr., 2008). IWA Task Group on Benchmarking of Control Strategies for WWTPs: <http://www.ensic.inpl-nancy.fr/benchmarkWWTP/Bsm1/Benchmark1.htm>.
  - [7] V. A. Akpan and G. D. Hassapis, "Training dynamic feedforward neural networks for online nonlinear model identification and control applications". *International Reviews of Automatic Control: Theory & Applications*, vol. 4, no. 3, pp. 335 – 350, 2011.
  - [8] V. A. Akpan (Jul., 2011): *Development of new model adaptive predictive control algorithms and their implementation on real-time embedded systems*, Ph.D. Dissertation, 517 pages. [Online] Available: <http://invenio.lib.auth.gr/record/127274/files/GRI-2011-7292.pdf>.
  - [9] A. Iratni, R. Katebi, R. Vilanova and M. Mostefai, "On estimation of unknown state variables in wastewater systems", *In the Proceedings of the 14th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'2009)*, Palma de Mallorca, Spain, 22 – 26 Sept., 2009, pp. 1 – 8.
  - [10] L. Lung, *System Identification: Theory for the User*. 2nd ed. Prentice-Hall, 1999.
  - [11] V. A. Akpan and G. D. Hassapis, "Nonlinear model identification and adaptive model predictive control using neural networks", *ISA Transactions*, vol. 50, no. 2, pp. 177 – 194, 2011.
  - [12] J. Sjöberg and L. Ljung, Overtraining, regularization, and searching for minimum in neural networks, *International Journal of Control*, vol. 62: 1391-1408, 1995.
  - [13] J. Hertz, A. Kroug and R. G. Palmer, "An Introduction to the Theory of Neural Computation", Lecture Notes, vol. 1, Redwood City, California: Addison-Wesley, 1991.