TITLE: **INTELLIGENT CONTROL VIA WIRELESS SENSOR NETWORKS FOR ADVANCED COAL COMBUSTION SYSTEMS**

TYPE OF REPORT: Final Scientific/Technical Report

REPORTING PERIOD START DATE: 02/05/2006

REPORTING PERIOD END DATE: 08/05/2007

PI: Aman Behal, Sunil Kumar, and Goodarz Ahmadi

REPORT ISSUE DATE: 11/2007

GRANT NO.: DE-FG26-06NT42687

INSTITUTION: Clarkson University
Potsdam, NY 13699-5700
Tel: 315-268-6446
Email: ahmadi@clarkson.edu

ABSTRACT

Numerical Modeling of Solid Gas Flow, System Identification for purposes of modeling and control, and Wireless Sensor and Actor Network design were pursued as part of this project. Time series input-output data was obtained from NETL's Morgantown CFB facility courtesy of Dr. Lawrence Shadle. It was run through a nonlinear kernel estimator and nonparametric models were obtained for the system. Linear and first-order nonlinear kernels were then utilized to obtain a state-space description of the system. Neural networks were trained that performed better at capturing the plant dynamics. It is possible to use these networks to find a plant model and the inversion of this model can be used to control the system. These models allow one to compare with physics based models whose parameters can then be determined by comparing them against the available data based model. On a parallel track, Dr. Kumar designed an energy-efficient and reliable transport protocol for wireless sensor and actor networks, where the sensors could be different types of wireless sensors used in CFB based coal combustion systems and actors are more powerful wireless nodes to set up a communication network while avoiding the data congestion. Dr. Ahmadi's group studied gas solid flow in a duct. It was seen that particle concentration clearly shows a preferential distribution. The particles strongly interact with the turbulence eddies and are concentrated in narrow bands that are evolving with time. It is believed that observed preferential concentration is due to the fact that these particles are flung out of eddies by centrifugal force.

TABLE OF CONTENTS

Executive Summary

The specific objectives of this project were to (a) develop a rough lower order dynamical system model for the co-generation coal power plant and/or parts thereof such as the gasifier, combustor and/or hot-gas cleaning devices, (b) develop an intelligent control algorithm for control of the processes involved, (c) develop the basis for wireless network approach for the intelligent control of the power plants and/or the associated equipment, and (d) perform computer simulation of the equipment and develop interfaces between the dynamical system and the CFD simulation.

Success was achieved in terms of both research and training of graduate students. The project began with a demonstration of nonlinear system identification for a simple test model where excitation signal dependence of identified system model was illustrated. Next, better dynamic performance in closed-loop with multiple control inputs as opposed to a single input was demonstrated. Additionally, reliable Transport protocols for wireless sensor and actor network were developed. In the second quarter, experimental cold-flow CFB data obtained from NETL's Morgantown, West Virginia facility was utilized to show nonparametric nonlinear system identification. Next, a state space representation was obtained via parameterization of the linear kernel. On a parallel track, reliable and energy efficient transport protocols for wireless sensor networks were developed in order to monitor various CFB process parameters. In the next phase, parameterization of the linear and the first-order nonlinear kernels was performed and discussion was provided to explain the results obtained. Finally, a non-linear state space representation of the overall dynamical system was obtained. To obtain results better than the average model obtained from using the Volterra analysis, neural networks were trained for capturing more details of the plant dynamics. An inversion of the obtained plant model can be utilized for controller design. In the last stage of the project, Direct Numerical Simulation of gas solid flows in a duct was performed. It was seen that particle concentration clearly shows a preferential distribution. The particles strongly interact with the turbulence eddies and are concentrated in narrow bands that are evolving with time. It is believed that observed preferential concentration is due to the fact that these particles are flung out of eddies by centrifugal force.

1: MODELING DETAILS

An overview of the tasks in the project and their relation to one another can be seen in Figure 1.
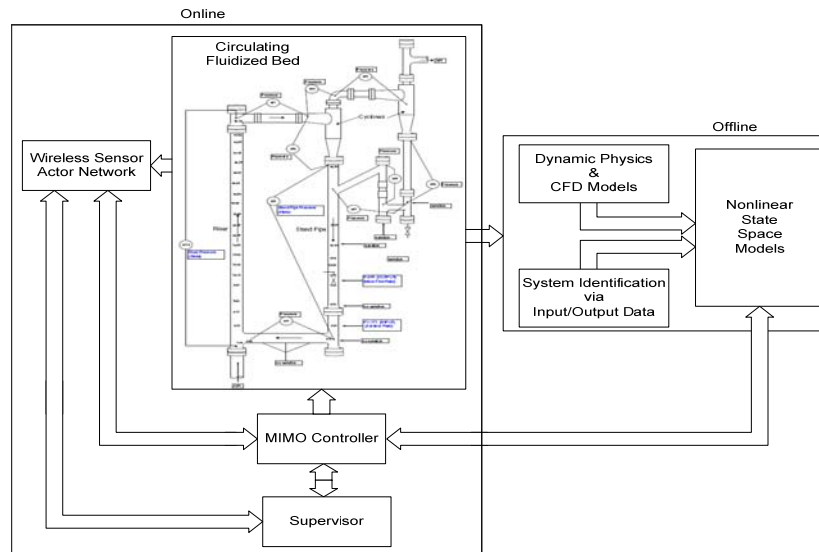

Figure 1: Overview of the tasks in the project

1.1: VOLTERRA SERIES SYSTEM IDENTIFICATION

1.1.1: A Brief Introduction

Nonparametric modeling approach is considered to identify the cold flow circulating fluidized bed system. Nonparametric modeling of this nonlinear dynamic system from input and output is done based on "Volterra Functional Expansion (or Volterra series)". We use Volterra Kernels since they are input-output dependent, and are therefore more practically desirable. In equation 1, the discrete-time finite-memory Volterra model limited to a third order is shown, where $x(n)$ and $y(n)$ are the input and output, respectively, $k_j(i_1,.....,i_j)$ is the jth – order Volterra kernel and I is the length of the system memory.

"Volterra Series: Volterra series is used to denote the (generally infinite) functional expansion of an analytic functional representing the input-output relation of any continuous and stable nonlinear dynamic system with finite memory"

$$y(n) = \sum_{i_1=0}^{I} k_1(i_1)x(n-i_1) + \sum_{i_1=0}^{I}\sum_{i_2=0}^{I} k_2(i_1,i_2)x(n-i_1)x(n-i_2)$$
$$+ \sum_{i_1=0}^{I}\sum_{i_2=0}^{I}\sum_{i_3=0}^{I} k_3(i_1,i_2,i_3)x(n-i_1)x(n-i_2)x(n-i_3) + \cdots ,$$

….1

Volterra Kernels are the unknown quantities in "Volterra Models". "Laguerre Expansion technique" (LET) is the most efficient method for the estimation of Volterra Kernels which are critically important in the actual application. The discrete laguerre functions (DLFs) used in LET approach are given by

$$b_j(m) = \alpha^{(m-j)/2}(1-\alpha)^{1/2}\sum_{k=0}^{j}(-1)^k \binom{m}{k}\binom{j}{k}\alpha^{j-k}(1-\alpha)^k$$

….2

where $b_j(m)$ denotes the jth-order orthonormal DLF, the integer m ranges from *0* to *M-1*, and the real positive number α *(0<α<1)* determines the rate of exponential (asymptotic) decline of these functions.

1.1.2: Results from NETL Experimental Data

To support advanced coal conversion technologies, NETL has constructed a cold flow circulating fluidized bed unit in Morgantown, West Virginia a schematic for which can be seen in Figure 4. Here, we show how the data from the NETL (See Figure 2 (using higher order modeling)) unit is used for the system identification using Laguerre Kernel Estimation Technique. Aeration rate [SCFH] at tag FY-171 and Mass Flow rate [lb/hour] at tag F-874 in stand pipe are considered as input and output, respectively. The calculated kernel estimates are shown in Figure 3. The first order kernel shows the linear properties of the CFBCF unit and the second order kernel shows the nonlinear properties of the CFBCF unit.

We remark here that it is possible to calculate higher order kernels if required and that this systematic approach allows for extension to MIMO system identification. As an example of a MIMO process, we can consider the various aeration rates in both standpipe and riser as inputs and the mass flow rate in stand pipe and chemical composition of the expelled gases as outputs.
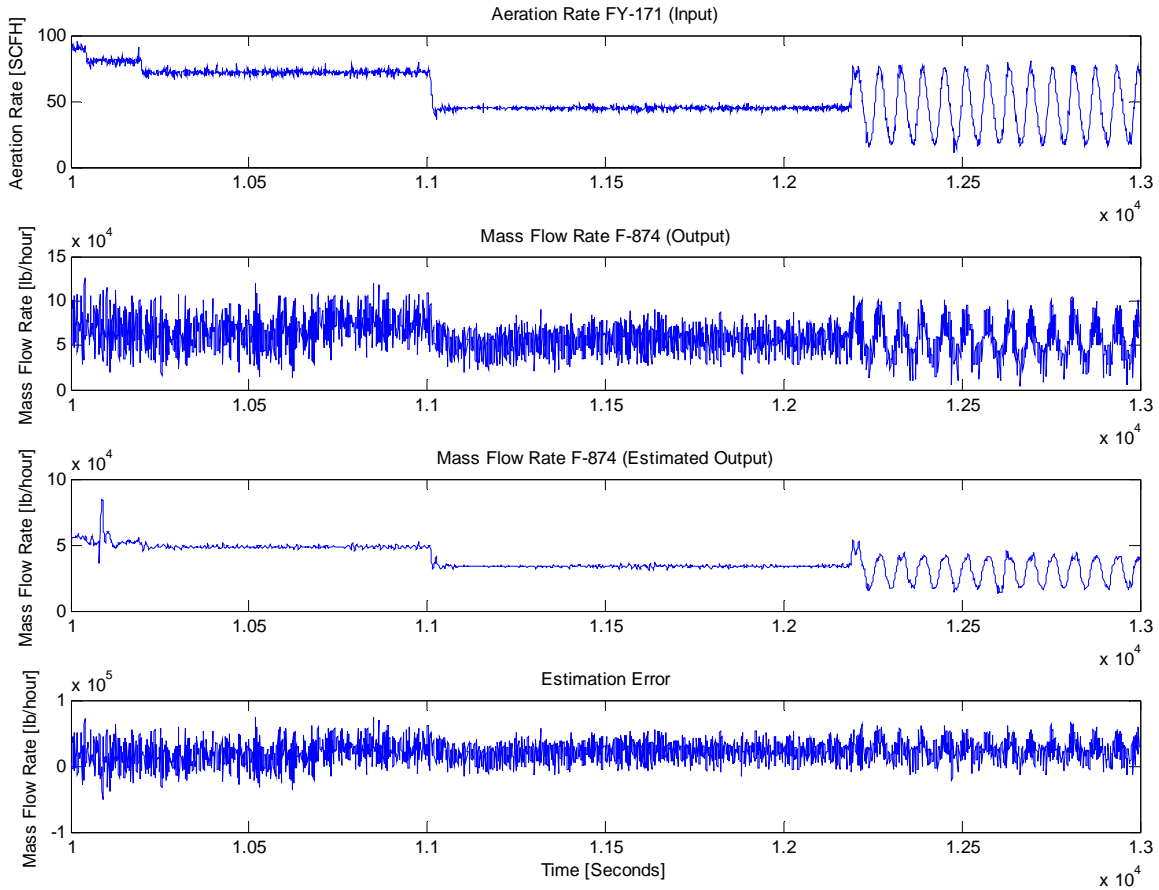
Figure 2: Data from NETL's CFB (Circulating Fluidized Bed) cold flow unit (CFBCF) and the estimation using Volterra Kernels
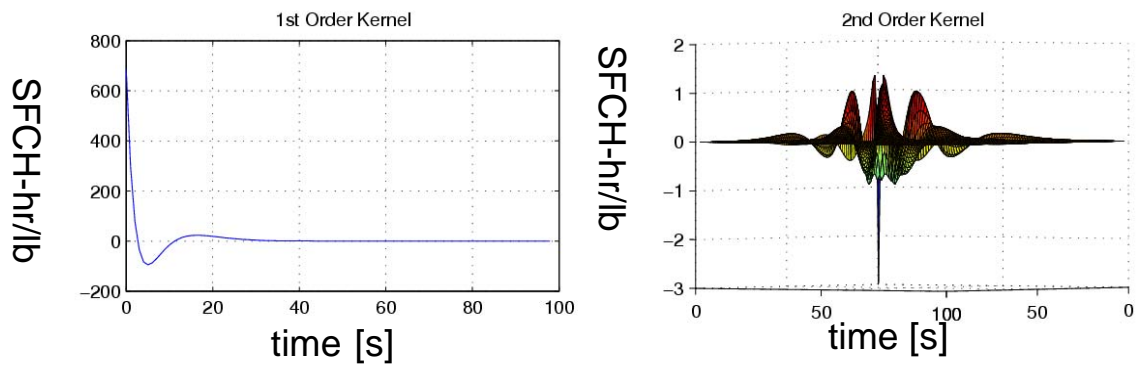
Figure 3: First and Second order kernels of NETL's Morgantown CFBCF unit

1.1.3: Parameterization of obtained kernels

While the nonparametric or kernel representation is generally good from a predictive point of view (See Discussion in 1.1.4 for why the estimate of mass circulation rate has significant differences from the measured values in Fig 2) and can be utilized to simulate

various operating scenarios for the system on a computer, it is not very effective for control design. What is useful for control design is if one could obtain a transfer function (matrix for MIMO) or a nonlinear state space model. Availability of such a parameterized model would facilitate the use of both standard linear frequency or time domain techniques and state-of-the-art nonlinear adaptive and robust control techniques.

Here, we demonstrate how we can parameterize the computed experimental first order kernel (from Figure 3) using available computation techniques and MATLAB toolboxes. By computing the numerical first order impulse response via fast Fourier transform and with the knowledge of system order, we utilize a least squares based data fitting approach to compute the 4th order transfer function G(s)

$$G(s) = \frac{704.1s^3 + 868.3s^2 + 56.13s + 11.56}{s^4 + 1.998s^3 + 1.149s^2 + 0.2693s + 0.02289} \qquad \dots 3$$

The location of poles and zeros of the linear part of the above equation 3 is

Poles	Zeros
-1.2291	-1.1773
-0.2546 + 0.0830i	-0.0279 + 0.1147i
-0.2546 - 0.0830i	-0.0279 - 0.1147i
-0.2596	

Next, this transfer function can be converted to the following (A,B,C,D) state-space representation (controllable canonical form)

$$A = \begin{bmatrix} -1.9975 & -1.1492 & -0.2693 & -0.0229 \\ 1.0 & 0 & 0 & 0 \\ 0 & 1.0 & 0 & 0 \\ 0 & 0 & 1.0 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T$$

$$C = \begin{bmatrix} 704.0650 & 868.3317 & 56.1282 & 11.5636 \end{bmatrix}, D = 0$$

This represents our simplest parametric linear dynamic representation of the NETL system data. A new approach was taken to parameterize higher order kernels and is

explained later. The new approach is taken for higher order kernels because no similar parameterization techniques are available in MATLAB for higher order kernels. The point of truncation presents an upper bound on the degree of uncertainty in our system models. Eventually, we can obtain nonlinear input-output mathematical models of the form

$$L(D)y + \sum_{i=0}^{\infty} \sum_{j=0; i+j \geq 2}^{\infty} c_{i,j} y^i (Dy)^j = M(D)x$$

…4

where *L(D)* and *M(D)* are polynomials of the differential operator *D = d(.).dt*, and *x(t)* and *y(t)* are the system input and output, respectively. The polynomial *L(D)* is assumed to be of degree higher than first (i.e., it involves at least the second derivative of *y*), and the polynomial *M(D)* is of lower degree than *L(D)* in order to avoid the emergence of singular (delta) function in the equivalent Volterra kernels. The nonlinear terms are of degree two and higher, and may be viewed as terms of a Taylor expansion of an analytic function of *y* and its derivative.

The kernel functions *{k_n}* characterize the dynamics of the nonlinear system, the mathematical model similar to Equation 4 can be derived analytically from the kernels using the method of "Generalized Harmonic Balance" [Marmarelis, 1982, 1989, 1991]. The first and second order kernels shown in Figure 3 are considered and the method of "Generalized Harmonic Balance" is used to come up with a mathematical model shown below

$$\underbrace{D^4 y + 1.998 D^3 y + 1.149 D^2 y + 0.2693 Dy + 0.02289 y}_{Linearity}$$
$$\underbrace{+6.91e^{-9} y^2 + 1.98e^{-5} y(Dy) - 5.547e^{-5}(Dy)^2}_{Nonlinearity}$$
$$= \underbrace{704.1 D^3 x + 868.3 D^2 x + 56.13 Dx + 11.56x}_{Linearity}$$

…5

It can be noticed that the nonlinear coefficients are comparatively negligible with respect to the linear coefficients. It can be concluded that the effect of nonlinearity is very less in the system, analyzed with the given set of input and output data. Due to the negligible

effect of 2nd order kernels, the Volterra kernel estimation was truncated to 2nd order.

1.1.4: Discussion

Estimation error of the output i.e., the Mass Flow Rate F-874 from the Volterra kernel is plotted in Figure 2 and can be noticed that the estimation error is very high. It can be concluded that the mathematical model obtained is accurate in an average sense in that it only describes the low frequency perturbations in the systems. The high frequency oscillations are seen to be uncorrelated to the input signal considered, we believe that the oscillations may be linked to other inputs into the system that have not been considered in this analysis. It may be possible that noise in measurements is being amplified since some of the controls are working in a closed-loop fashion. It is hard to determine whether a poor reference signal or a noisy measurement could be causing such an effect. It is also possible that the system is operating in a limit cycle oscillation due to possible stick-slip friction effects.

1.2: DETAILS OF NEURAL NETWORK FOR MODELING AND CONTROL

1.2.1: Introduction

In this phase of our work, we resorted to exploiting a neural network based technique – these neural networks allow not only for modeling but may also be used for control design. Figure 4 shows a schematic of the CFB that we modeled from data obtained from NETL Morgantown laboratory courtesy of Dr. Lawrence Shadle.

The system has chaotic nature and is highly nonlinear. Also a number of immeasurable unknowns make the system very complex. Artificial neural networks have the ability to characterize the system. While a model would allow one to map the outputs of the CFB over a defined operating range, it could not be used to predict outputs outside this operating range. Some of critical parameters graphs are given Figure 5 below. Parameter in Figure 5 were chosen based on engineering judgment exercised by the researchers, it was not possible to expound on a systematic choice because of limited resources (time and money) for this project. It is envisioned that future studies can allow for more discussion.

1.2.2: Training the Network

The neural network is allowed to train on a single input vector a finite time, each time it receives a new one. Each time the network completes a training iteration and it looks for a pattern in the data by adjusting its weight values. Based on the past input vectors, the network predicts an output value, one time-step in the future. The real data obtained from the CFB are used in training the network.
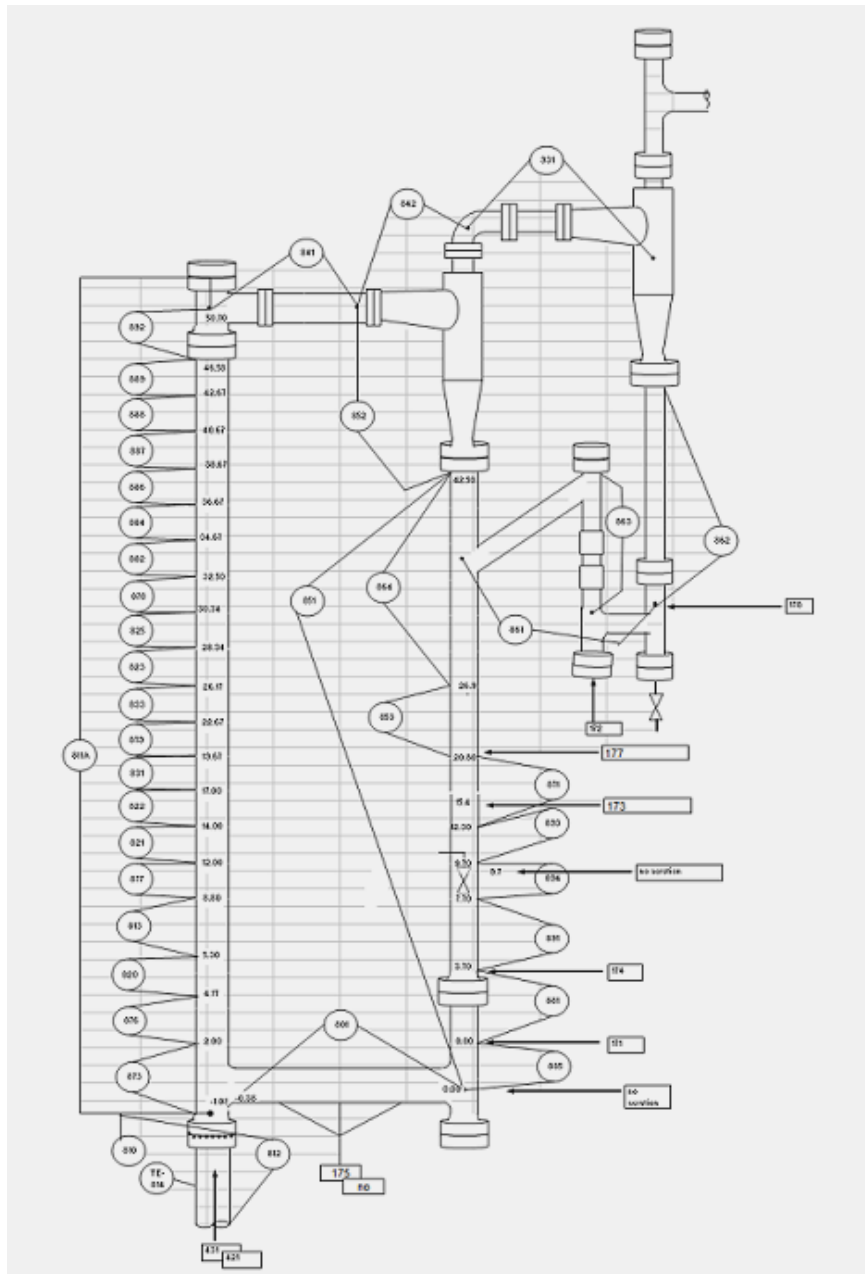

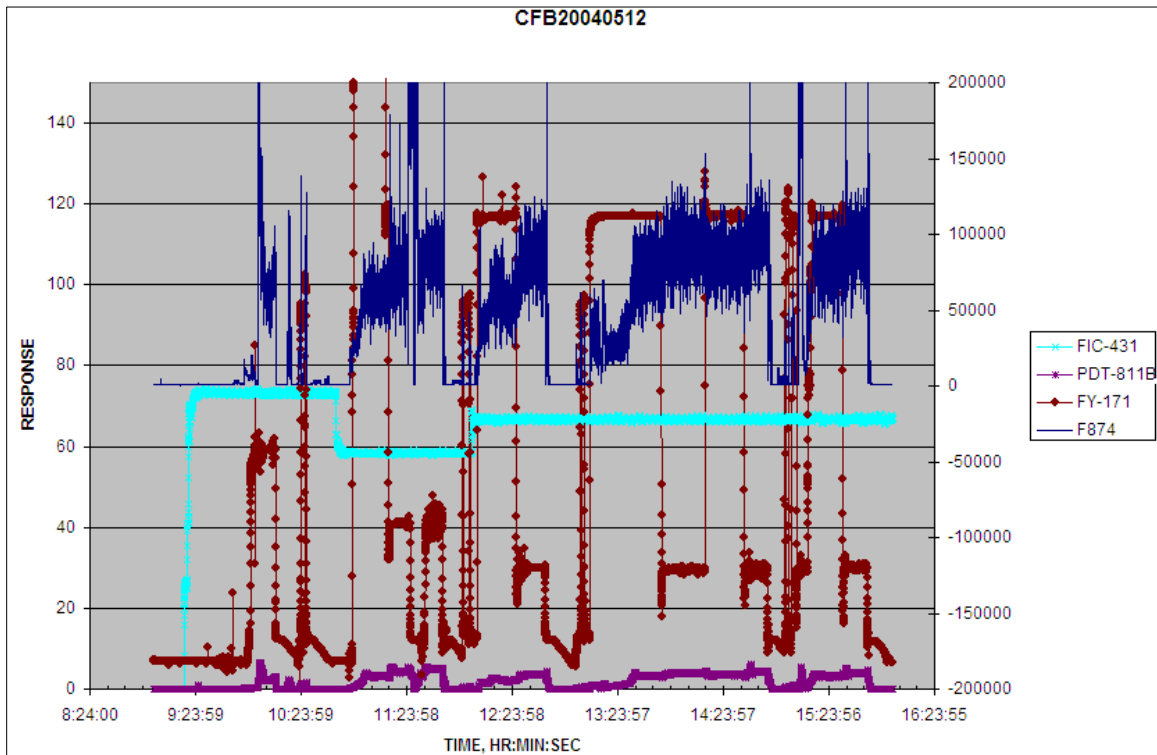Figure 4: Schematic view of the Circulating fluidized bed (CFB)

Figure 5: The graphs of critical parameters (FIC-431, PDT-811B, FY-171, F874)

1.2.3: Results

The stand-pipe (Stand-pipe was chosen to be a benchmark study, further studies are not possible for riser or non-mechanical valve because of resource restriction) is one of the major units of the CFB, therefore it is of interest to predict the operation of this unit. The neural network is designed with different kind of independent variables (different variables are chosen because poor results in Figure 2 were obtained with earlier choice), measured from aeration rates and pressure across the stand-pipe. The mass flow rate (F 874) is considered to be the dependent variable. The independent variables are used as the inputs of the neural networks, and the dependent variable is the output.

The soft computing method used here is the back propagation neural network with the generalized delta rule employed as the learning mechanism. The tuning of the multilayer network is achieved by changing the activation function, error function and learning factor.

23 experiments were given with three different data sets. Four of them are given

with the graphs (Experiments 1,8,11,17). Others are in Appendix A. Experiments are given with number of inputs, name of the inputs, number of outputs, name of the output, and mean square error (mse). Also explanation of the inputs is given as follows:

Aeration Rates: 170, 171, 172, 173, 174, 175, 176, 177, 178, 421, 431

Pressures:801, 811A, 811B, 813, 820, 821, 822, 823, 825, 831, 833, 841, 842, 851, 852, 853, 861, 862, 863, 864, 865, 871, 873, 876, 877, 878, 879, 881, 882, 883, 884, 885, 886, 887, 888, 888, 889, 891, 892, 893, 894, 902, 931, 932, 980, 985, 986, 988
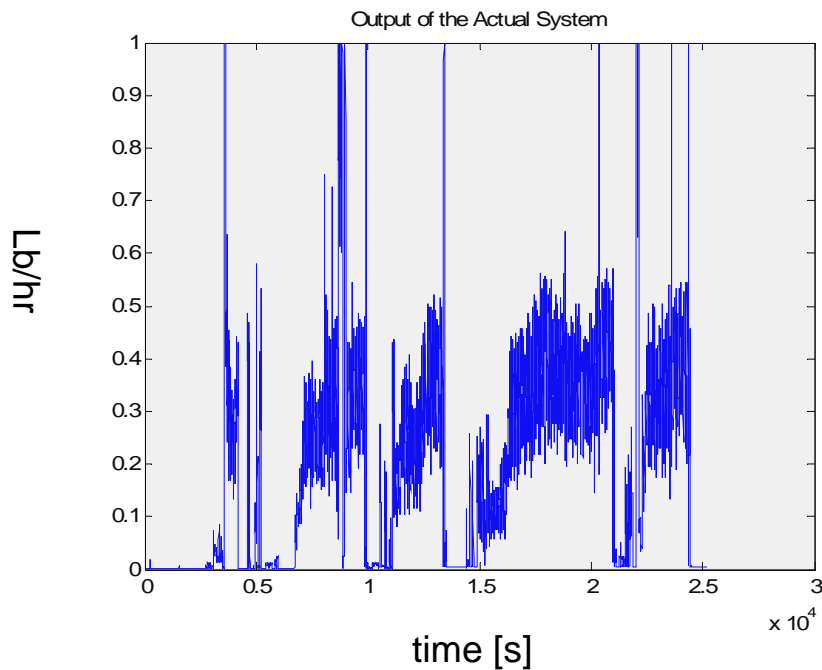

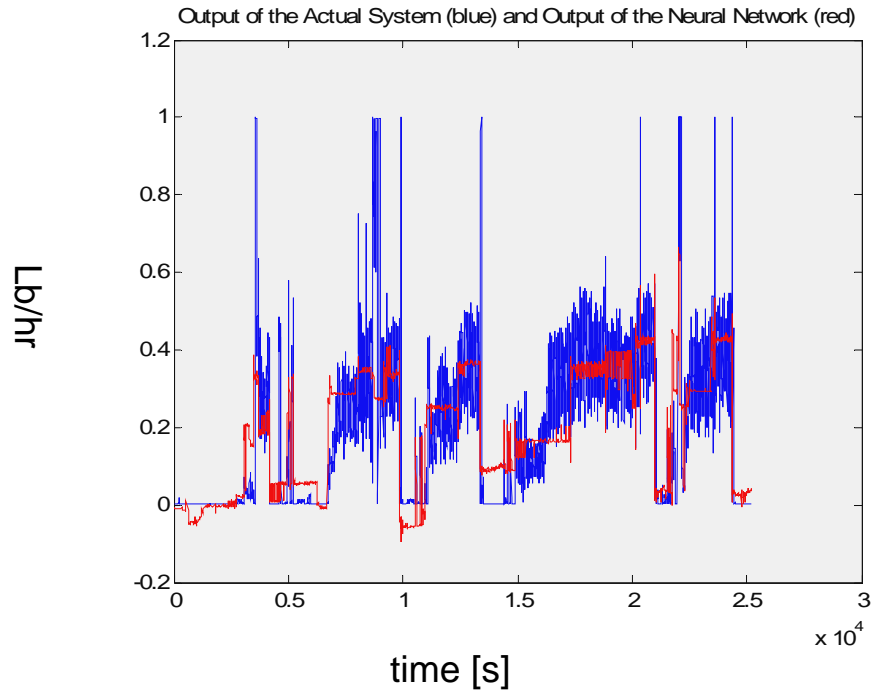Figure 6: Plant output: mass flow rate (F 874)

Figure 7: Actual mass flow rate (blue) and trained neural network output (red)
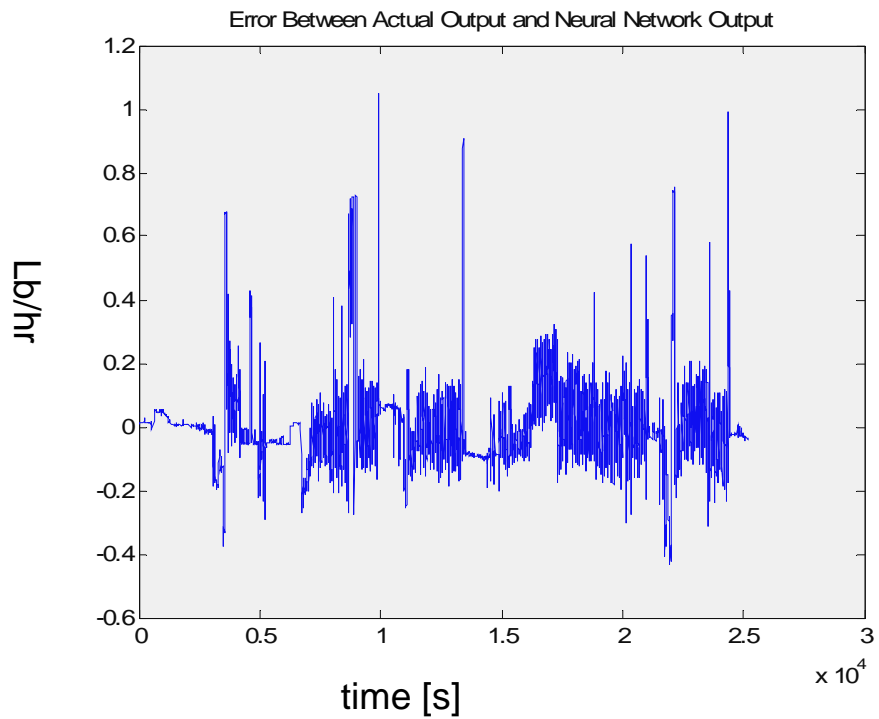
Figure 8: Error between actual mass flow rate and neural network output. [Experiment 1]

Experiment 1

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)

Input #:10

Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178

Output#:1

Output: F 874 (mass flow rate)

Mse : 0.015

Experiment 8

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)

Input #:27

Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178, 811A, 801, 885, 881, 891, 894, 893, 871, 853, 864, 861, 863, 862, 852, 931, 842, 841.

Output#:1

Output: F 874 (mass flow rate)

Mse : 0.0034
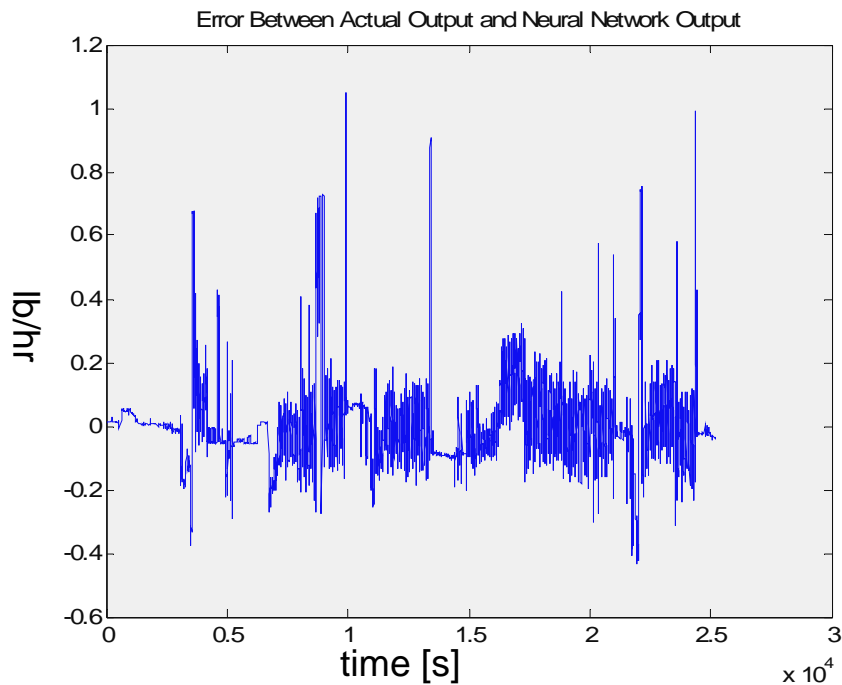
Figure 9: Actual mass flow rate (blue) and trained neural network output (red).

[Experiment 8]

Experiment 11

Data was taken on 5/12/2005 (20050512.xls)

Input #:20

Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178, 811A, 801, 851, 852, 841, 842, 931, 862, 863, 861.

Output#:1

Output: F 874 (mass flow rate)

Mse : 0.0028
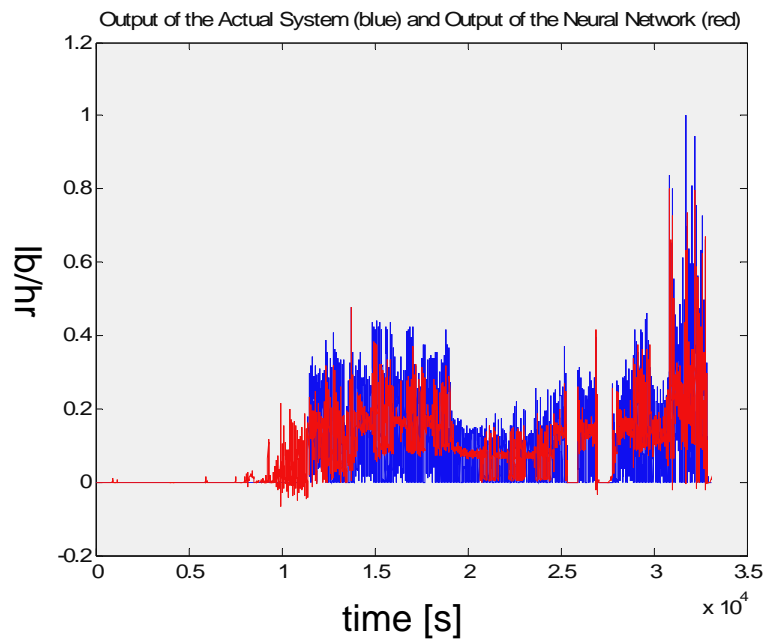
Figure 10: Actual mass flow rate (blue) and trained neural network output (red).

[Experiment 11]

Experiment 17 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)

Input #:7

Inputs :171, 431, 811B, 801, 881, 891, 174

Output#:1

Output: F 874 (mass flow rate)

Mse : 0.00254

Figure 11: Output of the actual system. [Experiment 17]

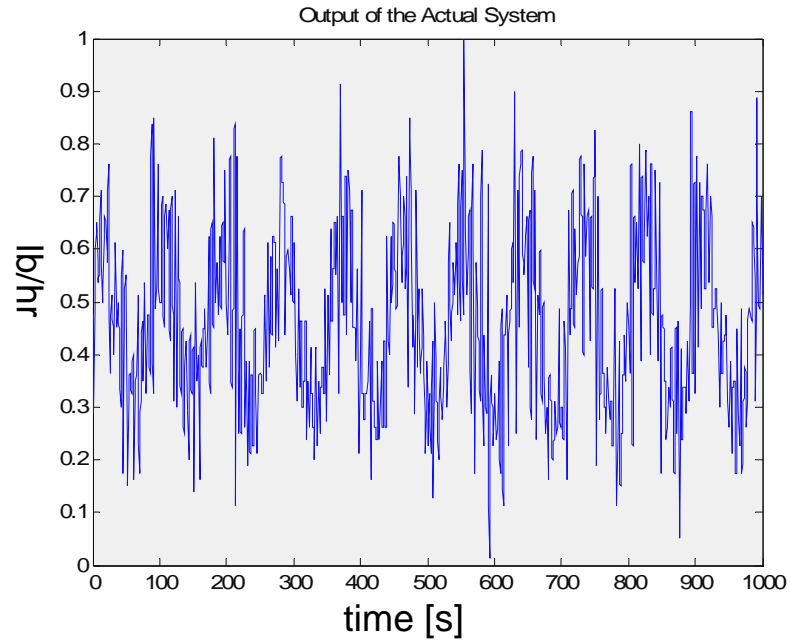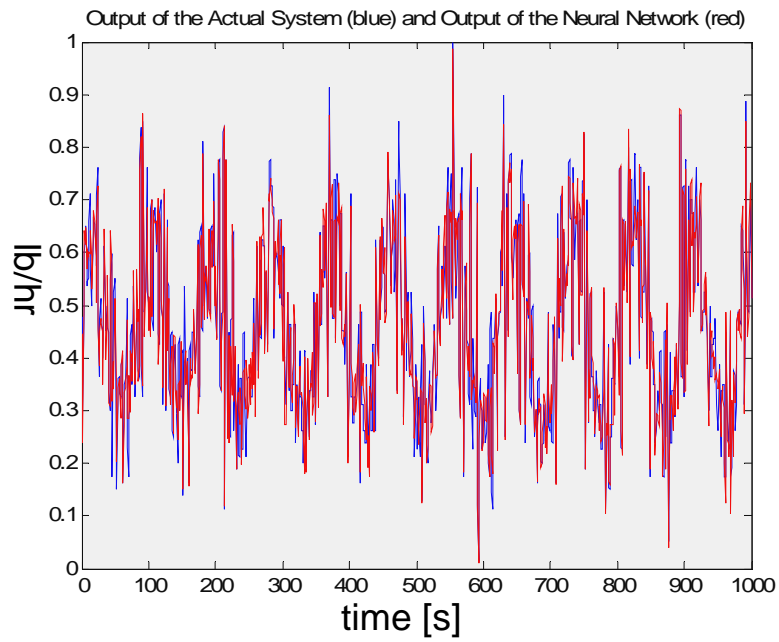Figure 12: Actual mass flow rate (blue) and trained neural network output (red).

[Experiment 17]
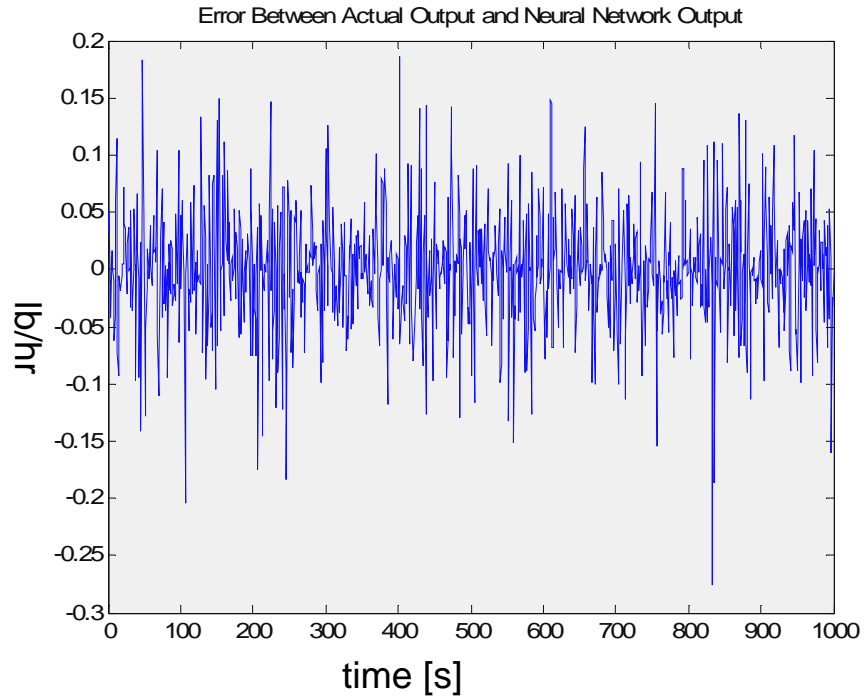
Figure 13: Error between actual mass flow rate and neural network output.

[Experiment 4]

1.2.4: Conclusions

We demonstrated that neural networks can be use to design a predictor for CFB system. The back propagation algorithm is used for training the neural network. The results are indicating that neural network learn faster in the 17th experiment. The reason for this is most probably the type of the input. In that experiment, the data was obtain with smooth sinusoidal input. From the results, we can easily say that the use of pressures as inputs of the neural network decrease the error. If the neural network could be trained even more, it would be used as the identification of the plant; it can be used as a controller by finding the inverse plant model. It is not possible to extend the modeling work to this control strategy because of limited resources.

2: DETAILS OF DIRECT NUMERICAL SIMULATION OF GAS SOLID FLOW IN A DUCT

2.1: INTRODUCTION

In the last section, we modeled the stand-pipe to predict the operation of the unit. The back propagation neural network with the generalized delta rule employed as the learning mechanism was utilized. The tuning of the multilayer network is achieved by changing the activation function, error function and learning factor. It was seen that the system had chaotic nature and was highly nonlinear. Also a number of immeasurable unknowns make the system very complex.

In what follows, we utilize Direct Numerical Simulation to gain insight into the physics behind the flow inside the unit.

2.2: SIMULATION OF GAS-SOLID FLOWS

The gas solid flows in a duct were analyzed using a direct numerical simulation (DNS) method. The Navier-Stokes equation was solved using a pseudospectral computational approach. The particle equation of motion included the drag, the Saffman lift, and the Brownian forces. Several simulations for a wide range of particle relaxation time were performed, and the effect of presence of particles on the mean and the RMS particle velocities were studied. It was observed that particles with small relaxation time have the same streamwise velocity as the flow. Particles with moderate relaxation times, however, will lead the flow slightly very near the wall due to the turbophoretic effect. The larger particles will lead the flow significantly in the wall region due to the "free flight" mechanism. As particle relaxation time increases, the particle streamwise fluctuation velocity exceeded the flow one, while the particle transverse and spanwise fluctuation velocities became smaller than the flow transverse and spanwise fluctuation velocities. It was also observed that as particle relaxation time increases, the turbophoretic force decreases but the particle transverse velocity increases. Ultra fine particles were uniformly distributed in the channel, and with no accumulation near the wall. Medium size particles, however, had a tendency to accumulate near the wall because of the turbophoretic effects. The interaction of particles with the flow causes the particles to be preferentially distributed in the channel centerplane.

2.3: GOVERNING EQUATIONS

The instantaneous fluid velocity field in the channel is evaluated by the direct numerical simulation of the Navier-Stokes equation. It is assumed that the flow is incompressible, and a constant mean pressure gradient in x-direction is imposed. The corresponding governing equations of motion are:

Continuity equation:
$$\nabla \cdot u = 0 \qquad \qquad \ldots 6$$

Momentum equation:
$$\frac{Du}{Dt} = -\frac{1}{\rho^f}\nabla p + \nu \nabla^2 u \qquad \qquad \ldots 7$$

Here, $u = (u, v, w)$ is the fluid velocity vector, p is the pressure, ρ^f is the fluid velocity, ν is the gas kinematic viscosity. No-slip boundary conditions are assumed on the channel walls and periodic boundary conditions are imposed in the x- and z-directions. That is,

$$u = 0, \qquad y = \pm H \qquad \qquad \ldots 8$$

$$u(x + \lambda_x, y, z + \lambda_z, t) = u(x, y, z, t)$$

In these simulations, a channel that has a width of H, and a $\lambda_x \times \lambda_z$ periodic segment in x- and z- directions is analyzed.

The channel flow code used in this study used a pseudo-spectral method. Accordingly, the fluid velocity is expanded in a three-dimensional Fourier-Chebyshev series. The fluid velocity field in the x- and z- direction is expanded by Fourier series, while in the y-direction the Chebyshev series is used. The code uses an Adams-Bashforth-Crank-Nickolson (ABCN) scheme to compute the nonlinear and viscous terms in the Navier-Stokes equation and performs three fractional time steps to advance the fluid velocity.

The equation of motion for a spherical particle is given as:

$$\frac{du^p}{dt} = F_d + F_l + F_g \qquad \qquad \ldots 9$$

$$\frac{dx}{dt} = u^p \qquad \qquad ..10$$

In Equation (9), the terms on the right-hand side are, respectively, the drag force, the lift force and the gravitational force per unit mass of particle.

2.4: RESULTS

Sample particle concentration in the channel center plane is shown in Figure 14. It is seen that particle concentration clearly shows a preferential distribution. In this case, the particles strongly interacts with the turbulence eddies and are concentrated in narrow bands that are evolving with time. It is believed that observed preferential concentration is due to due to the facts that these particles are flung out of eddies due to the centrifugal force. Because of time and resource restrictions, it was not possible to develop interfaces between the dynamical system and the CFD simulation.
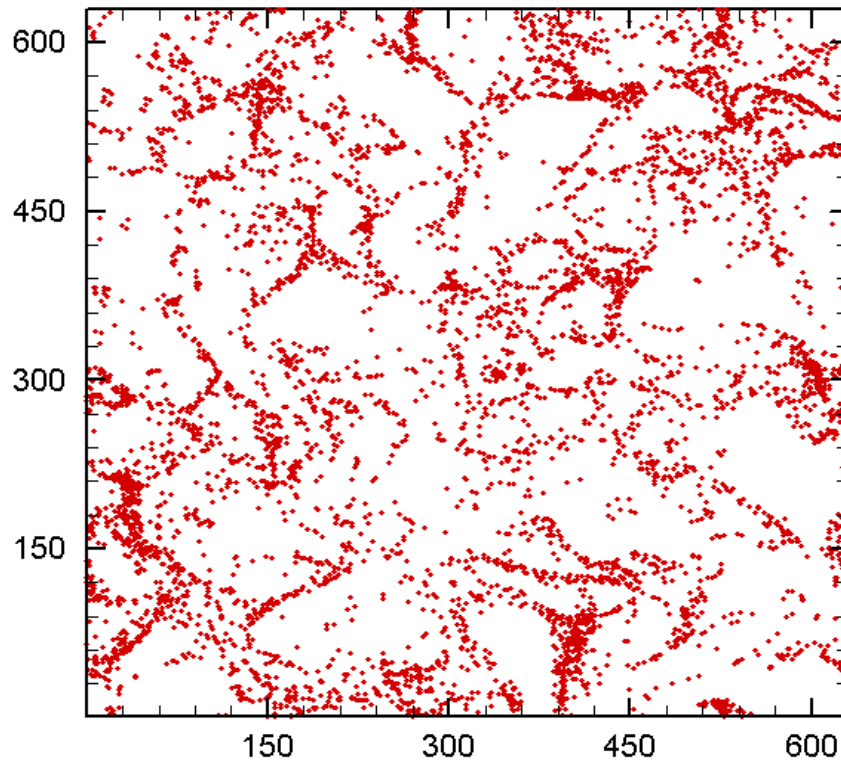

Figure 14. Preferential particle concentration in the duct center plane.

3: A RELIABLE EVENT-TO-SINK TRANSPORT SCHEME IN WIRELESS SENSOR AND ACTOR NETWORKS

3.1: INTRODUCTION

One graduate student was supported to carry out simulation studies of energy efficient

and reliable protocols for wireless sensor and actor networks applicable to CFB. The details of the scheme are given below.

The proposed scheme is implemented on a scenario shown in Figure 15 below. The actor nodes monitor various sensor nodes that are measuring various CFB process parameters under their control and transmit the collected data to the sink (i.e., computer in Control room). The sensor nodes convey the event packets to the actor nodes through multihop in a unicast manner but actor nodes on the other hand send messages to the sensor nodes with high power in a single hop. The sink maintains a database of information of the network.

SENSOR
ACTOR
WIRELESS COMMUNICATION

A1
A2
A4
A3
A6
SINK
A5
A7
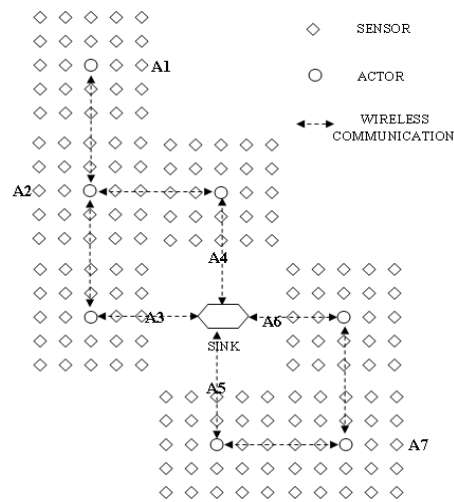
Fig 15: Network Scenario

The *event-to-sink reliability* and traffic control is very important in wireless sensor networks. In this research, we investigated the performance of the event-to-sink reliability transport (ESRT) scheme in the presence of various traffic loads due to event detection and reporting requirements. The ESRT scheme has been designed to achieve the optimum traffic reporting frequency (known as 'reliability'). During our investigations, we learnt that ESRT essentially assumes a single event reporting frequency requirement. When multiple events are detected by the sensors deployed in the plant leading to a situation where the desired reliability is sufficiently larger (we will discuss this later) than the maximum capacity of the network, the ESRT algorithm

fluctuates between two inefficient working states – congested, low reliability (C, LR) and no congestion, low reliability (NC, LR). This results in much lower than desired and unstable reliability performance.

We proposed a modified algorithm to overcome this problem. Simulation results show that our algorithm E^2SRT (Enhanced ESRT) outperforms ESRT in terms of both reliability and energy consumption in the presence of 'over demanding traffic'.

The proposed scheme has the following features:

1. It inherits all the merits of ESRT while eliminating the undesirable fluctuation in case of 'over demanding'.

2. Robust Convergence: it has no assumption at the server side on reliability demand. The user can pose any reliability requirement on the network. Instead of fluctuations, the network will recursively estimate the maximum reachable network capacity and converge to the realistic Optimal Region (OR).

3. Awareness of Dynamic Environment: many factors will affect the available resources for wireless sensor network, such as environmental change, sensor node movement, sensor power etc. This contributes to a highly dynamic network system. E^2SRT can dynamically make adjustment to such unpredictable changes and quickly converges to render best performance with given network parameters.

3.2. ESRT performance in case of over-demanding desired event reliability

Our simulation results for the ESRT scheme in WSN (shown in Fig. 16) reveal that when the desired reliability is beyond the capability of current network settings (known as over-demanding), the network will never be able to converge to the optimum operating range (OOR) state where normalized reliability equals to 1. Simulation results also show that the original ESRT scheme cannot detect this situation by itself. In fact, the ESRT algorithm tries to generate a reporting frequency according to this desired reliability value. However, these values lead the network either to severe congestion and packet losses or the network operates at a very low frequency rate, thus wasting most of its bandwidth. As a result, the network oscillates between congested low reliability state (C, LR) with fairly high reporting rate to the 'not congested low reliability' state (NC, LR) with very low reporting rate.
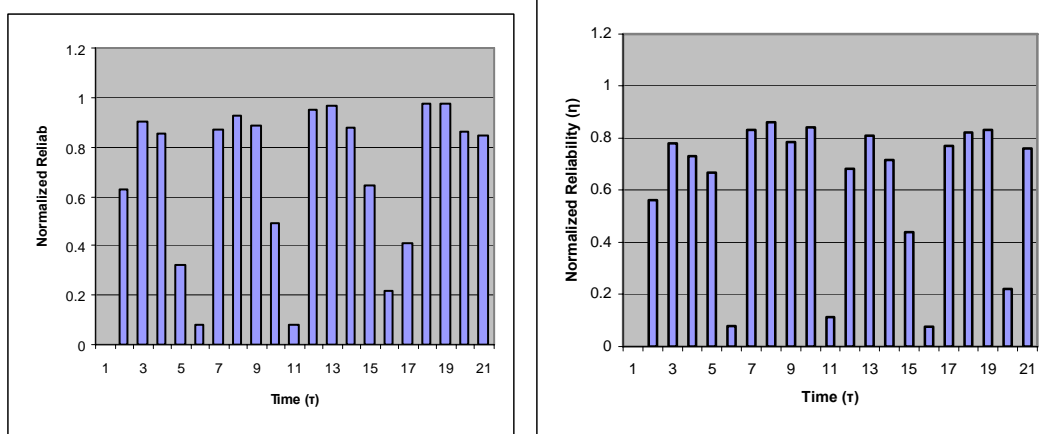
Figure 16: Normalized reliability fluctuates for ESRT in case of over-demanding desired reliability requirements. Reliability request is 4000 (left) and 4500 (right) for two graphs.

3.3. The Proposed E^2SRT Scheme

Before discussing our proposed Enhanced Event-to-Sink Reliability Transport (E^2SRT) scheme, we formally define the over-demanding desired reliability problem in ESRT.

The *over-demanding desired event reliability problem* denotes a situation where the desired reliability R is sufficiently larger than R_{max}, so that $R_{max}/R < 1-\varepsilon$. Here R_{max} represents the maximum Reliability that can be supported by the network. In such a situation, the network is in OR (Over-demanding Reliability) state as shown in Fig. 17. We shall represent this desired reliability situation as R_{od}.
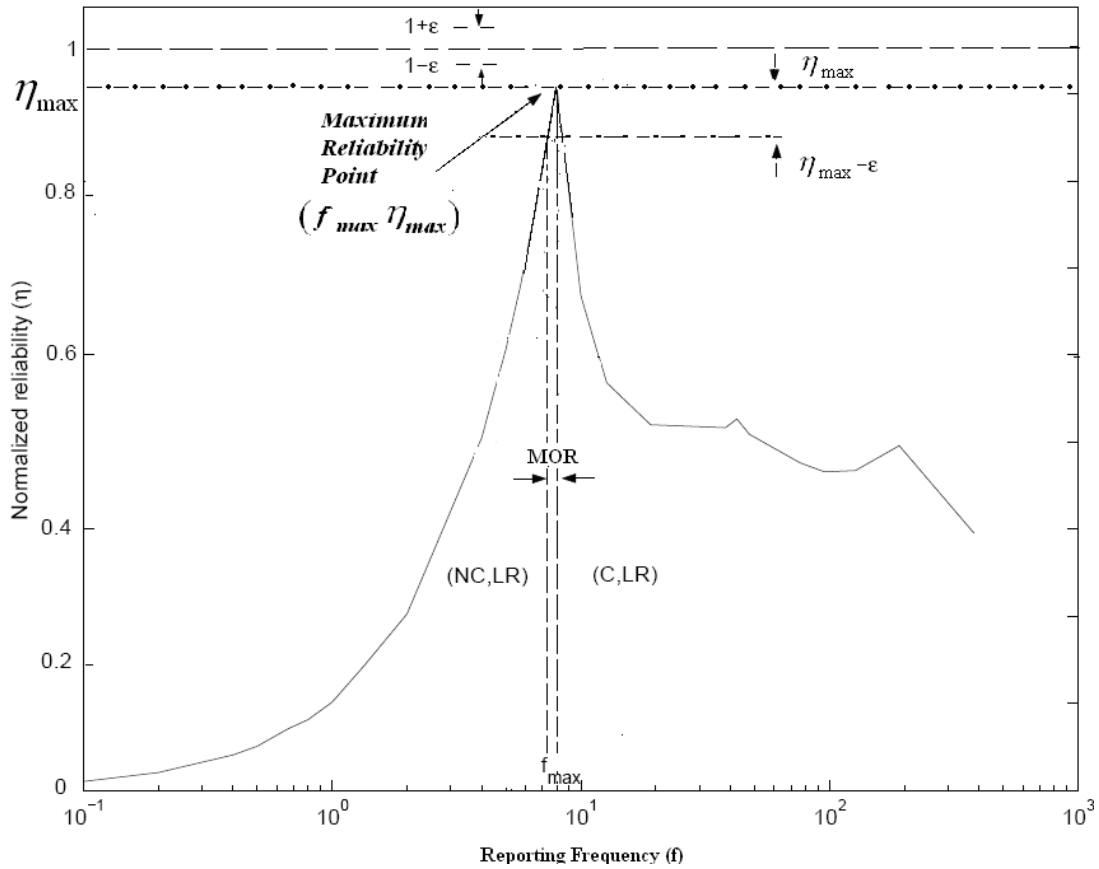
Figure 17: When ESRT cannot support desired event reliability, only two network states (NC, LR) and (C, LR) exist. For example, the desired reliability is 4000 packet and the network can only handle around 3500 packets per 10 second interval in our simulation settings. The requirement is beyond the network capability and no OOR (optimum operating range) state exists. ESRT does not take into account this situation, the network would fluctuate between (no congestion (NC), low reliability (LR)) and (congestion (C), low reliability (LR)) states.

To address this issue, we have divided the problem into the following two sub problems:

a. How to detect the over-demanding desired event reliability situation, and

b. If R_{od} problem exists, how to quickly converge to the maximum reliability the network can reach without requiring the full knowledge of the network conditions.

For question a, we have noticed in our simulations that the network will have a

transition from (NC, LR) to (C, LR) only when R_{od} exists. Otherwise it will follow the standard ESRT states model and transition graph.

Now the major design consideration is how to push the network to approach the maximum reliability point (f_{max}, η_{max}) (MRP) for a given network setting. For practical reasons, we allow a tolerance zone of width ε around MRP as illustrated in Fig. 3. If at the end of a decision interval i, the normalized reliability η_i is within $[\eta_{max} - \varepsilon, \eta_{max}]$ and no congestion is detected in the network, the network is in Maximum Operating Region (MOR). Here we follow the definition of tolerance zone of ESRT. It is a protocol parameter decided by the user. A smaller ε will generally give greater proximity to MRP while it may take longer convergence time.

If MRP (i.e., the maximum observed reliability) is known, sink can reduce the desired reliability such that the network can converge to OOR as in ESRT. However, it is difficult to calculate the exact value of MRP (f_{max}, η_{max}) due the following reasons: initial sensor nodes deployment; network topology changes due to sensor node movement, death or sleeping state; relocation/movement of events; radio interference; and deliberate over demanding to maximize the network throughput.

Consequently, more sophisticated algorithms are needed that can read feedback from sensor network and predict MRP in a recursive manner. Please note that the ESRT is a sink based transport scheme that serves to enforce reliability while conserving energy. The E^2SRT scheme introduces several new components to eliminate the *Rod* problem. However, the modification essentially resides at the algorithm level. Thus, E^2SRT inherits all the major features of ESRT such as communication model and network modes definitions. As an enhanced version, E^2SRT is more resilient to abrupt network changes and resource constraints due to its operations in OR states.

3.4. E^2SRT Performance Evaluation

Our results show that E^2SRT converges to operating point in MOR when the network is in OR state. For comparison, the simulation environment and network settings are the same as we used for the ESRT. As shown in Fig. 2, the normalized reliability (η) achieved by ESRT scheme varies from as high as 0.95 to as low as 0.1. On the other

hand, as shown in Fig. 18, normalized reliability is stabilized after several round of operation using the E^2SRT scheme when R_{od} situation exists. The desired reliability request is 4000 and 4500 respectively for the two figures in Fig. 18, while the maximum reachable reliability (R_{max}) is around 3500. The mean normalized reliability reached by the E^2SRT in the first 20 decision intervals is about 0.894 and 0.793, respectively, as compared to 0.683 and 0.609 for ESRT scheme. The proposed E^2SRT scheme thus improves the network reliability performance by 29.9% and 30.2%, respectively.
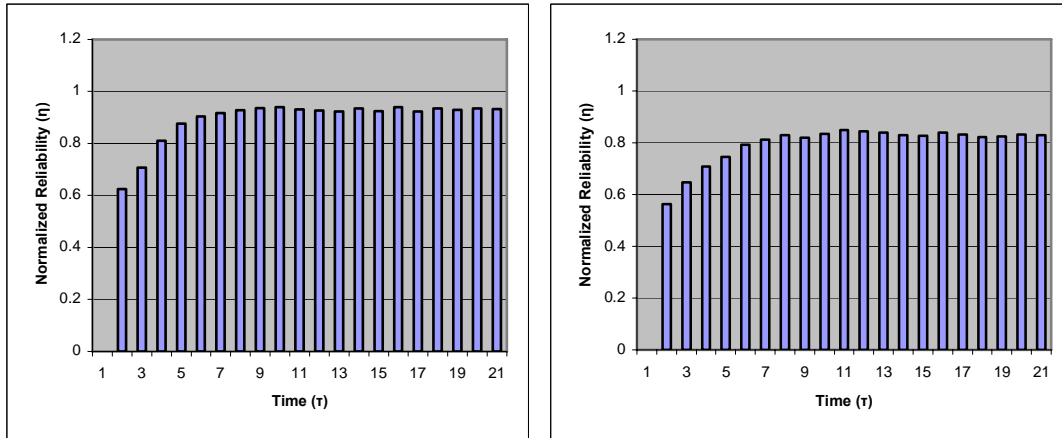

Figure 18: Normalized reliability stabilized after several rounds of operation using the E^2SRT scheme, for R_{od} situation is present. Reliability request is 4000 (left) and 4500 (right) graphs. Network settings are the same as they are for ESRT simulation. The maximum reachable reliability (R_{max}) is around 3500.

4. OVERALL CONCLUSION

Overall, the project was able to produce the following results in line with the expectations:

1. Linear and Nonlinear System Identification was accomplished based on the data for the NETL system at Morgantown.
2. A wireless sensor actor network was designed that provides congestion avoidance as well as reliability.
3. Direct Numerical Simulation provided clues to the flow of particle concentration inside the unit.

PAPERS PUBLISHED/CONFERENCE PUBLICATIONS/AWARDS RECEIVED/STUDENTS SUPPORTED

Students

$ V.M. Rao, (MS) student, Electrical and Computer Engineering, Clarkson University (Major Advisor, Dr. Behal)

$ K. K. Ram Kashyap, (MS) student, Electrical and Computer Engineering, Clarkson University (Major Advisor, Dr. Kumar)

$ H. Nasr (Ph.D. candidate), Mechanical and Aeronautical Engineering, Clarkson University (Major Advisor, Dr. Ahmadi)

$ Zenhua Feng (MS) student, Electrical and Computer Engineering, Clarkson University (Major Advisor, Dr. Kumar)

Publications

Conferences

- H. Nasr, G. Ahmadi, "Turbulence Modification by Particles in A Vertical Channel Flow," 7th International Aerosol Conference, St Paul, MN, September 10-15, 2006.

- H. Nasr, G. Ahmadi, and J.B. McLaughlin, "Effects of Inter-Particle Collisions and Two-Way Coupling on Particle Deposition Velocity in a Turbulent Channel Flow," 6th International Conference on Multiphase Flow, ICMF 2007, Leipzig, Germany, July 9-13, 2007.

REFERENCES

1. Marmarelis, V.Z. (1982). Non-parametric validation of parametric models. Mathematical Modeling 3:305-309.

2. Marmarelis, V.Z. (1989a). Identification and modeling of a class of nonlinear systems. Mathematical Computer Modeling 12:991-995.

3. Marmarelis, V.Z. (1989b). Linearized models of a class of nonlinear dynamics.Applied Mathematical Modeling 13:21-26.

4. Marmarelis, V.Z. (1989c). Signal transformation and coding in in neural systems. IEEE Transactions on Biomedical Engineering 36:15-24.

5. Marmarelis, V.Z. (1989d).The role of nonlinear models in neurophysiological system analysis. In: Modelling and Control in Biomedical Systems, C. Cobelli and L. Marianni (Eds.), pp 39-50, Pergamon Press, Oxford.

6. Marmarelis, V.Z. (Ed.). (1989e). Advanced Methods of Physiological System Modeling, Volume 2. Plenum, New York.

7. Marmarelis, V.Z. (1989f).Voltera-Wiener analysis of a class of nonlinear feedback systems and application to sensory biosystems. In: Advanced Methods of Physiological System Modeling, Volume 2, Plenum, New York, pp. 1-52.

8. Marmarelis, V.Z. (1991). Wiener analysis of nonlinear feedback in sensory systems. Annals of Biomedical Engineering 19:345-382.

Appendix A

Experiment 2

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)
Input #:19
Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178, 851, 852, 853, 864, 871, 885, 891, 893, 894
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.008

Experiment 3

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)
Input #:27
Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178, 851, 852, 853, 864, 871, 885, 891, 893, 894, 841, 842, 852, 931, 932, 902, 861, 863
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0052

Experiment 4

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)
Input #:51
Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178, 851, 852, 853, 864, 871, 885, 891, 893, 894, 841, 842, 852, 931, 932, 902, 861, 863, 811A, 811B, 801, 941, 892, 889, 888, 887, 886, 884, 882, 878, 825, 823, 833, 879, 831, 822, 821, 877, 813, 820, 876, 873.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.009

Experiment 5

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)
Input #:3
Inputs : 431, 811B, 171
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0118

Experiment 6

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)
Input #:11
Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178, 811A.
Output#:1
Output: F 874 (mass flow rate)

Mse : 0.008

Experiment 7

Data was taken on 5/12/2004 (CFB040512 time series (3).xls)
Input #:20
Inputs : 421, 431, 171, 172, 173, 174, 175, 176, 177, 178, 811A, 801, 851, 852, 841, 842, 931, 862, 863, 861.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0024

Experiment 9

Data was taken on 5/12/2005 (20050512.xls)
Input #:2
Inputs : 431, 171
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0083

Experiment 10

Data was taken on 5/12/2005 (20050512.xls)
Input #:6
Inputs : 431, 171, 811A, 885, 881, 801
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0067

Experiment 12

Data was taken on 9/7/2006 (20060907.xls)
Input #:2
Inputs :171, 431.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0052

Experiment 13

Data was taken on 9/7/2006 (20060907.xls)
Input #:3
Inputs :171, 431, 811B
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0045

Experiment 14

Data was taken on 9/7/2006 (20060907.xls)
Input #:4
Inputs :171, 431, 811B, 801.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0063

Experiment 15 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:6
Inputs :171, 431, 811B, 801, 881, 891
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0041

Experiment 16 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:7
Inputs :171, 431, 811B, 801, 881, 891, 841
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0022

Experiment 18 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:7
Inputs :171, 431, 170, 174, 176, 177, 175.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0046

Experiment 19 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:8
Inputs :171, 431, 170, 174, 176, 177, 175, 801.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0038

Experiment 20 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:9
Inputs :171, 431, 170, 174, 176, 177, 175, 801, 885.
Output#:1
Output: F 874 (mass flow rate)

Mse : 0.0045

Experiment 21 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:9
Inputs :171, 431, 170, 174, 176, 177, 175, 801, 881.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0074

Experiment 22 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:9
Inputs :171, 431, 170, 174, 176, 177, 175, 801, 891.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0032

Experiment 23 (1000 data)

Data was taken on 9/7/2006 (20060907.xls)
Input #:10
Inputs :171, 431, 170, 174, 176, 177, 175, 801, 891, 894.
Output#:1
Output: F 874 (mass flow rate)
Mse : 0.0026