2015

Sensor Placement Algorithms for Process Efficiency Maximization

Prokash Paul

Sensor Placement Algorithms for Process Efficiency Maximization

Prokash Paul

Dissertation submitted
to the College of Engineering and Mineral Recourses
at West Virginia University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in
Chemical Engineering

Dr. Debangsu Bhattacharyya
Dr. Richard Turton
Dr. Charter D. Stinespring
Dr. Stephen E. Zitney
Dr. Parviz Famouri

Department of Chemical Engineering

Morgantown, West Virginia
2015

Keywords: Sensor Network Design, Genetic Algorithm, Kalman Filter, Parallel Computation, Acid Gas Removal Unit.

ABSTRACT

Sensor Placement Algorithm for Maximizing Process Efficiency

Prokash Paul

Even though the senor placement problem has been studied for process plants, it has been done for minimizing the number of sensors, minimizing the cost of the sensor network, maximizing the reliability, or minimizing the estimation errors. In the existing literature, no work has been reported on the development of a sensor network design (SND) algorithm for maximizing efficiency of the process. The SND problem for maximizing efficiency requires consideration of the closed-loop system, which is unlike the open-loop systems that have been considered in previous works. In addition, work on the SND problem for a large fossil energy plant such as an integrated gasification combined cycle (IGCC) power plant with CO_2 capture is rare.

The objective of this research is to develop a SND algorithm for maximizing the plant performance using criteria such as efficiency in the case of an estimator-based control system. The developed algorithm will be particularly useful for sensor placement in IGCC plants at the grassroots level where the number, type, and location of sensors are yet to be identified. In addition, the same algorithm can be further enhanced for use in retrofits, where the objectives could be to upgrade (addition of more sensors) and relocate existing sensors to different locations. The algorithms are developed by considering the presence of an optimal Kalman Filter (KF) that is used to estimate the unmeasured and noisy measurements given the process model and a set of measured variables. The designed algorithms are able to determine the location and type of the sensors under constraints on budget and estimation accuracy. In this work, three SND algorithms are developed: (a) steady-state SND algorithm, (b) dynamic model-based SND algorithm, and (c) nonlinear model-based SND algorithm. These algorithms are implemented in an acid gas removal (AGR) unit as part of an IGCC power plant with CO_2 capture. The AGR process involves extensive heat and mass integration and therefore, is very suitable for the study of the proposed algorithm in the presence of complex interactions between process variables.

Dedicated to

Our research group and my Ph.D. committee members

Acknowledgements

I would like to thank National Energy Technology Laboratory's Regional University Alliance (NETL-RUA), a collaborative initiative of the NETL, for funding to this project. This technical effort was performed under the RES contract DE-FE0004000.

I would never have been able to finish my dissertation without the guidance of my committee members, help from friends, and support from my family.

I would like to express my deepest gratitude to my advisor, Dr. Debangsu Bhattacharyya, for his excellent guidance, caring, patience, and providing me with an excellent atmosphere for doing research. I would also like to thank Dr. Richard Turton, Dr. Charter D. Stinespring, Dr. Stephen E. Zitney (NETL) and Dr. Parviz Famouri for guiding my research for the past several years and helping me to develop my background in process system engineering, sensor network design and process modeling. My research would not have been possible without their helps.

I would like to thank Linda Rogers, Bonita Helmick and Kelydra Welcker for their great help in fulfilling administrative requirements.

I would like to thank Dustin Jones, Job Kasule and Kiran Chaudhari, who were good friends, were always willing to help and give their best suggestions. It would have been a lonely lab without them. I would like to thank other friends in our research group for their help in group study and coursework.

I would also like to thank my parents, younger siblings and my wife. They were always supporting me and encouraging me with their best wishes.

Table of Contents

List of Figures

List of Tables

Nomenclature

A	process transition matrix
a	power consumption coefficient due to solvent regeneration
α	time delay (min)
α_{sc}	primary scaling parameter
B	input matrix
b	scalar budget ($)
β_{sc}	secondary scaling parameter
C	measurement matrix
c_i	cost of individual sensor ($)
$chol(.)$	lower Cholesky decomposition
$E(.)$	expected value
$\varepsilon(t)$	deviation of the controlled variable from set point
F	flowrate (mole/hr)
f	polynomial
Gen	number of generations in GA
g	input memory
H	parameter vector
h	output memory
J_H	is Jacobian of $h(.)$
K	steady state Kalman gain matrix
K_c	proportional gain of P-only controller
K_p	steady state process gain

κ_{sc}	tertiary scaling parameter
L	length of state vector
l	number of sensor selected by GA
λ	scaling parameter
m	no of inputs
μ	mean
N	no. of individuals in population of each generation in GA
N_c	pre-specified number of counter
N_s	total number of candidate sensor
n	number of states
P	pressure
P	state error variance-covariance matrix
P_c	consumed power (MWh)
Q	process noise variance-covariance matrix
R	measurement noise variance-covariance matrix
s	polynomial
T	temperature
t	time (hr)
τ	process time constant (min)
τ_I	reset time
W	weighting factor
u	inputs
u_c	vector of control inputs

u_d	vector of disturbance
v	measurement noise vector
w	process noise vector
x	vector of states
$\bar{x}$	mean
x_{CO_2}	composition of CO_2 in liquid phase
$\hat{x}$	vector of estimated states
y	vector of measurements
$\hat{y}$	vector of estimated variables
z	noisy measurements

Subscripts

act	actual variable
β	decision variable vector of binary numbers (0 and 1)
$cont$	control/controlled variable
CO_2	carbon dioxide
c	covariance
est	estimated/ estimator-based
in	inlet or, input
$incum$	incumbent
m	mean
ma	process monitoring variables and active constraints
mon	monitoring variables

opt	optimal
r	reduced order model
set	desired set point
$solvent$	Selexol solvent
ss	steady state
$term$	termination
$(.)_i$	ith tray

Chapter 1

Introduction

Power plants are facing increasingly higher targets for efficiency. At the same time, environmental emission standards are becoming tighter. Under these constraints, the sensor network can play an important role in meeting these goals. An optimal sensor network can help to achieve the desired performance in the process and power plants. However, due to the possibility of trillions of candidate combination of sensors, it is very difficult to find the optimal locations, numbers and types of sensors in a large scale plant.

A number of process variables that are measured have low precision, reliability, or signal-to-noise ratio. However, plant operators or control systems take action based on these poor measurements thus resulting in a suboptimal operation. The variables that are measured can be of two types. The first type is used for monitoring purposes. For example, if the measured variable is an environmental variable, then a measurement error can lead to violation of environmental emission limits. If the measured variable is a key variable for monitoring equipment health, an error can lead to undesired conditions such as equipment damage. In addition, many other process variables are monitored to avoid safety hazards, or unwanted products or other undesired conditions. Therefore, desired estimation accuracy must be achieved by the measurement network for these variables. The second type of measured variables is used as controlled variables. Some variables under this category can affect the plant efficiency. In the method proposed by Skogestead (2004), if the plant control structure is systematically designed by optimizing the economic performance, then all the controlled variables (also called primary controlled variables) in such control structure affect plant efficiency.

Jones et al. (2014) have extended the work of Skogestad (2004) by incorporating the control performance of the primary controlled variables in the selection criteria. In this approach, optimizations are performed for maximizing an economic objective with respect to steady-state degrees of freedom (DOF) by considering various disturbances. The active constraints are

selected as primary controlled variables. In addition, depending on the remaining DOF, additional controlled variables are selected so that they are self-optimizing. The self-optimizing controlled variables are those that when left constant, result in an acceptable economic loss in the face of disturbances (Skogestad, 2000). If the primary controlled variables in a plant are self-optimizing and the plant has been optimally designed, then a deviation from the optimal values of the controlled variables would result in a loss in efficiency. The extent of this loss in efficiency depends on the magnitude and direction of the deviation. A low estimation accuracy of these variables will lead to a loss in efficiency. On the other hand, setting arbitrarily high estimation accuracy will result in undesired increase in the cost of the senor network. Therefore, unlike existing sensor network design (SND) methods where the desired estimation accuracy of all variables of interest is set by the user, no specifications are needed for estimation accuracy of the self-optimizing controlled variables when using the estimator-based control system.

For the primary controlled variables that are active constraints, the change in the process efficiency with respect to a change in the variable is monotonic, at least locally. Therefore, specifications have to be provided by the user for either the positive or the negative estimation accuracy of these variables, but not necessarily for both positive and negative. This aspect is better explained by the following example. Consider a CO_2 capture unit with an operational objective of 90% CO_2 capture. It has been well-documented that CO_2 capture can strongly affect plant efficiency (Figueroa, 2008). Due to inaccuracies in the measurement system, two undesired operational scenarios may occur while maintaining the target capture rate. In the first scenario, the measurement system might show that CO_2 capture is less than the target (e.g. 89.8%) even though the actual capture is exactly 90%. In this scenario, the plant operators/control system will change the operating conditions to increase the amount of CO_2 capture to maintain it at the set point thereby causing a loss in process efficiency. For this scenario, the negative estimation accuracy can be determined by considering the tradeoff between efficiency and cost. In the second scenario, if the measurement system shows a greater (i.e. 90.2%) CO_2 capture level even though the actual capture is 90%, the plant operators/control system will change the operating conditions to decrease the CO_2 capture. As a result more CO_2 will be released to the environment, which can result in a penalty from the regulating agencies. For this scenario, the allowable positive estimation accuracy has to be set by the user. For many measurement

instruments, the estimation accuracy guaranteed by the manufacturer is the same in both positive and negative direction. For such instruments, the SND algorithm should automatically determine the limiting deviation and design the sensor network accordingly.

With these motivations, a new SND algorithm has been developed assuming an estimator-based control system where an optimal Kalman filter (KF) (Kalman, 1961) is used to estimate the states in the presence of measurement and process noise. Due to the feedback loop in the control system, the resulting system of equations becomes very difficult to converge for any arbitrary set of integer variables (i.e., set of sensors). Thus, the objective of this research is to develop SND algorithm for optimal selection of sensor location, number, and type that can maximize the plant efficiency in addition to obtaining a desired precision of the key measured/unmeasured states in a large, highly-integrated industrial process.

Chapter 2 presents existing literatures in the area of SND algorithm development. Literature has been reviewed mostly from the perspective of designing sensor network and inherent computational expense in solving large scale problems. Chapter 3 discusses state estimation in the presence of large model mismatch and high measurement noise.

A steady-state SND (SSND) algorithm, presented in the fourth chapter, is developed using a sequential optimization algorithm. The algorithm follows the infeasible path method where a ‘tearing’ approach is used to solve the feedback loops. The methodology is developed in a way that large-scale systems can be solved efficiently. In this work, the integer programming problem is solved by the genetic algorithm (GA) method while other linear and nonlinear constraints are satisfied by a sequential equation solver using a ‘tear’ stream approach. Chapter 4 also discusses in more detail how this formulation helps in satisfying the linear and nonlinear equality constraints for every combination of integer variables.

Chapter 5 presents a dynamic model-based sensor network design (DMSND) algorithm for efficiency maximization of a transient system. DMSND algorithms can be computationally very expensive due to the study of the transient behavior for each candidate set of sensors. This computational expense significantly increases as the number of state variables and the number of

candidate sensor variables increase. In particular, the solution of the matrix Riccati equation that appears in KF, takes significant computational time. However, for the DMSND algorithm to be usable for large-scale industrial applications, an efficient algorithm is desired that can be solved within reasonable run times using reasonable computing resources. With this incentive, a computationally efficient DMSND algorithm using the estimator-based control system approach has been developed in this work for maximizing the efficiency. In DMSND algorithm, KF is used for estimating process states (Paul et al., 2013) and particular focus is given to its convergence properties. In addition, several strategies have been developed to reduce significantly the computational expenses. This algorithm is designed to be implemented using a GA approach.

Chapter 6 extends the DMSND algorithm for nonlinear process model thus referred to as nonlinear model-based SND algorithm (NDMSND). A multi-objective optimization problem has been solved for optimal sensor network design. Chapter 6 presents the identification of a nonlinear process model using input-output data and this is followed by the lexicographic optimization of process efficiency and budget for the sensors. GA is used to implement the designed NDMSND algorithm.

To achieve the objective mentioned before, the system that needs to be considered for developing the SND algorithm is shown in Figure 1.1. In the estimator-based control system, the measurement network affects the estimation accuracy assuming an optimal KF is implemented for estimation. As a result, the control action is affected and finally, due to the control action, the process efficiency is affected.

Many of the SND algorithms in the existing literature have been applied to small simplified test problems. In this work, the developed methodology is applied to a large, highly integrated acid gas removal (AGR) unit as part of an integrated gasification combined cycle (IGCC) power plant with CO_2 capture. This AGR unit comprises of a number of typical unit operations involving considerable mass and energy integration and, therefore, is a very good industrial case study for the proposed algorithm.

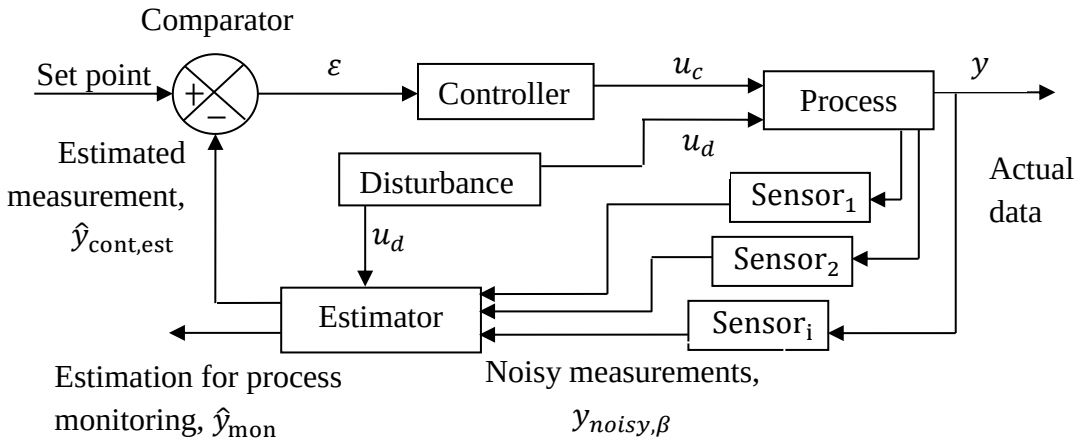

Figure 1.1: Schematic of the estimator-based control system for development of the SND algorithm.

Chapter 2

Literature Review

Over the last two decades, optimal sensor placement has been an area of active research. Researchers have primarily focused on sensor placement either for process monitoring or fault detection and identification purposes. Fault diagnosis is beyond the scope of this research. Interested readers are referred to some of the seminal works in SND by Raghuraj et al.(1999), Bhushan et al. (2000, 2002a, 2002b, 2008), Muslin et al. (2004) and Narasimhan et al.(2007). One of the popular goals for SND is to obtain the cost-optimal sensor network. Bagajewicz (1997, 2000), Bagajewicz and Cabrera (2002), and Chmielewski et al. (2002) have obtained a minimal cost SND subject to constraints on precision, error detectability, reliability, and resilience. Kelly and Zyngier (2008) have minimized the cost of a sensor network with constraints on software and hardware redundancy. Kadu et al., (2008) has presented an SND algorithm for maximizing estimation accuracy. A review of SND algorithms through 2000 can be found in the book of Bagajewicz et al. (2000). SND from an economic perspective has been presented by Bagajewicz and Markowski (2203) and Bagajewicz (2005a). Later, Bagajewicz (2005b) has extended the concept of economic value of precision by introducing the effect of induced bias obtained by evaluating the economic value of accuracy. Bagajewicz et al. (2005, 2006, 2008) have also investigated economic value of data reconciliation and instrumentation upgrades. Peng and Chmielewski (2006, 2005) have placed sensors from the controls perspective.

Other than different objectives considered for the SND problem, different computational methods have also been developed in the open literature for designing optimal SND. A tree search approach has been used by Bagajewicz (1997), Bagajewicz and Sanchez (2000), and Bagajewicz and Cabrera (2002) to solve a mixed integer problem. Later an equation-based tree search method for the design of a nonlinear sensor network is presented by Nguyen and Bagajewicz (2008, 2013). A genetic algorithm (GA) has been used by Zumoffen and Basualdo (2010). A graph theoretic approach has been used by Meyer et al. (1994) and Luong et al. (1994)

to design a sensor network for process monitoring. An approach combining the GA and graph theoretic approaches has been developed by Sen et al. (1998) to synthesize a non-redundant SND algorithm for linear processes. Madron and Veverka (1992) have adopted a Gauss-Jordan elimination method to optimize overall measurement cost and overall precision of a system. Recently, a stochastic optimization-based method is proposed by Ghosh et al. (2014) to identify an optimal subset of measured variables for effective statistical process monitoring.

Computational expense has always been an issue for solving large-scale SND problems. Due to this difficulty, Chmielewski et al. (2002) has offered an alternative SND formulation to obtain a minimum cost sensor network. The authors have improved computational efficiency by converting the nonlinear programming problem into a convex program through the use of linear matrix inequalities. They have applied the SND approach to both steady-state and dynamic processes subject to single/multiple constraints on precision, gross-error detectability, resilience, and reliability. Nguyen and Bagajewicz (2008) have proposed a rigorous equation-based tree search method for designing nonlinear sensor networks but its performance is not satisfactory when dealing with large-scale problems ($\geq$ 35 measured variables and $\geq$ 25 balance equations). Later on, they have proposed an approximate method (Nguyen and Bagajewicz, 2013) to solve a large-scale problem with 35 variables and 28 balance equations where the equation-based tree search method is used for initialization but still optimality of the solution is not guaranteed. Singh and Hahn (2005) have obtained optimal sensor locations for stable nonlinear state estimation by maximizing the degree of observability based upon observability covariance matrix. In addition, they have presented an optimal SND approach (Singh and Hahn, 2006) for a nonlinear dynamic system by considering the trade-off among process information, measurement cost, and redundancy subject to the constraint on number of sensors. Due to the computational expense, they have performed the calculation of empirical Gramians for all sensor locations outside the optimization and then finally GA is used to solve a mixed integer nonlinear programming (MINLP) problem. Recently, Nguyen and Bagajewicz (2011) have presented an optimal SND approach based on value of information. Serpas (2012) creates a generic approach for finding the optimal sensor network design for nonlinear systems. In order to choose the best sensor network, a metric is defined. In this work, the determinant of the empirical observability Gramian is chosen. However, for systems that are unobservable or marginally observable, this

metric creates numerical problems. As a way of producing meaningful results, state space reduction is performed on the matrix before the determinant. Finally, this paper incorporates methods from the optimization literature for efficiently solving the mixed integer nonlinear programming problem (MINLP) that results from the maximization of the determinant. This combination of proposed approaches utilizes the information provided by the observability Gramian in order to determine the best sensor network design.

Effort has also been made to design sensor network for improving power plant performance. Lee and Diwekar (2012) have developed an optimal sensor placement algorithm for advanced power plants where a stochastic integer programming problem is solved to maximize the Fisher information subject to budget constraints. Recently, Sahraei et al. (2014) have presented a comprehensive literature review on the sensor placement methodologies and control strategies to improve power plant efficiency. Recently, Nguyen and Bagajewicz (2011) have proposed a SND algorithm for maximizing the difference between the economic value of information and cost.

Literature review on state estimation using KF and adaptive KF is provided in Chapter 3. Review of the existing literature in the area of steady-state process model-based SND is discussed in Chapter 4. Chapter 5 includes the literature review on DMSND algorithm and computational expense in solving large scale problems. Existing work in the area of nonlinear state estimation, nonlinear model identification and NDMSND algorithm development are discussed in Chapter 6.

In the following chapters, the contributions to the field of sensor network design will be discussed. Major contributions during the course of this work are in the area of algorithm development for known/unknown state estimation and SND algorithm for determining optimal location, number and type of measurements.

Chapter 3

State Estimation

Accurate estimation of measured/unmeasured process variables is crucial to satisfy the constraints on environmental emission. In this chapter, traditional and adaptive Kalman Filter (KF) is introduced and basic discrete KF is discussed. Use of an adaptive KF will be presented that adapts process noise covariance matrix (Q) and measurement noise covariance matrix (R) at every time step. Performance of the traditional KF is compared with the adaptive KF by introducing a number of input disturbances to an acid gas removal process.

3.1 Literature Review

Kalman (Kalman, 1961) published a recursive solution to the discrete data linear filtering problem. Since that time, the KF has been the subject of extensive research and application. The KF is a set of mathematical equations that provide efficient computational means to estimate the measured/unmeasured state of a process which minimizes the squared error between the actual and the estimated states. KF supports estimation of past, present and future states even in the presence of measurement noise and large mismatch between the model and the actual process. A general idea about KF can be found in Maybeck (1979). More interested readers are referred to the following references Zarchan et al. (2009), Sorenson (1970), Gelb (1974), Grewal and Andrew (1993), Lewis (1986), and Jacobs (1993).

3.1.1 Traditional KF

For traditional KF, the process is described by equations that are in matrix or state-space form as

$$\frac{dx}{dt} = Ax + Bu + w \tag{3.1}$$

In Eq. (3.1) state variables represented by x are a column vector, A is the nonsingular constant process matrix and B is the input matrix. u is a known vector, which is sometimes called the control vector. The random variable w is process white noise, which is also expressed as vector. The process noise covariance matrix Q is related to process noise vector according to

$$Q = E[ww^T] \tag{3.2}$$

where $E[.]$ designates the expected value. Although process noise might not always have physical meaning in this work it is used as a device to represent mismatch between the linear model and nonlinear process as well as unmodeled process dynamics. The traditional KF requires that the measurements be linearly related to the states as

$$y_{noisy} = Cx + v \tag{3.3}$$

In Eq. (3.3) y_{noisy} is the measurement vector, *C* the measurement matrix and *v* is the white measurement noise, which is also expressed as a vector. *w* and *v* are assumed to be uncorrelated. The measurement noise covariance matrix *R* is related to measurement noise vector according to

$$R = E[vv^T] \tag{3.4}$$

where *R* and *Q* are manually tuned. However, the distinguishing feature of the traditional KF is that *R* and *Q* are kept constant during state estimation. Therefore, good guesses for both *Q* and *R* are required to obtain satisfactory filter performance. However, in industrial applications, these matrices are unknown and it is difficult to generate good guesses for them.

The estimation error *e* between actual state *x* and estimated state $\hat{x}$ is defined as

$$e = x - \hat{x} \tag{3.5}$$

and the estimation error covariance matrix, $P = E[ee^T]$ (3.6)

P can be calculated from the following matrix differential equation,

$$\frac{dP}{dt} = -PC^T R^{-1} CP + PA^T + AP + Q \tag{3.7}$$

which in turn is used to compute Kalman gain, *K*.

$$K = PC^T R^{-1} \tag{3.8}$$

Kalman gain in Eq. (3.9) is a set weight on the difference $\left(y_{noisy} - C\hat{x}\right)$. The difference is called the measurement innovation or residual. $C\hat{x}$ is the predicted measurement. The residual of zero means that the y_{noisy} and $C\hat{x}$ are in complete agreement.

$$\frac{d\hat{x}}{dt} = A\hat{x} + Bu + K(y_{noisy} - C\hat{x}) \tag{3.9}$$

Preceding relationships are discretized to derive discrete traditional Kalman filter. Consider discrete measurements with time interval Δt.

$$\Phi = \exp(A\Delta t) \tag{3.10}$$

$$G = \int_0^{\Delta t} \exp(A\sigma)\, B d\sigma \tag{3.11}$$

Φ and G are the discrete transition matrix and discrete input matrix respectively.

The linear stochastic vector-difference equation (Eq. 3.12-3.13) of the discrete dynamic system is shown as

$$x_k = \Phi x_{k-1} + G u_{k-1} + w_{k-1} \tag{3.12}$$

$$z_k = C x_k + v_{k-1} \tag{3.13}$$

The traditional discrete KF estimates the process states based on the predictor-corrector approach. The traditional KF algorithm is shown below

Initial Conditions

$x_0 \; and \; P_0$

Predictor: Time Update Equations

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \tag{3.14}$$

$$P_k^- = A\hat{P}_{k-1}A^T + Q \tag{3.15}$$

Corrector: Measurements Update Equations

$$K_k = P_k^- C^T (CP_k^- C^T + R)^{-1} \tag{3.16}$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - C\hat{x}_k^-) \tag{3.17}$$

$$\hat{P}_k = (I - K_k C)P_k^- \tag{3.18}$$

$\hat{x}_k$, $\hat{x}_k^-$, $\hat{P}_k$, P_k^- , and K denote the estimated state vector, predicted state vector, estimated state covariance matrix, predicted state covariance matrix, and optimal Kalman gain, respectively. Looking at Eq. (3.16), as the measurement error covariance R approaches zero, the actual measurement y_k is trusted more, and the gain K weights the residual more heavily.

$$\lim_{R_k \to 0} K_k = C^{-1} \tag{3.19}$$

On the other hand, predicted state covariance matrix P_k^- approaches zero, the actual measurement y_k is trusted less, and the gain K weights the residual less heavily. Specifically,

$$\lim_{P_k^- \to 0} K_k = 0 \tag{3.20}$$

It implies that the predicted measurement is trusted more while P_k^- approaches zero.

3.1.2 Adaptive KF

In the traditional KF algorithm, the filter parameters (*R* and *Q*) are assumed constant. Because of the dynamic behavior of the process, periodic re-estimation of these matrices might be required. An Adaptive KF can be utilized to accomplish this.

3.1.2.1 Innovation-based Estimation of R

R can be adapted based on innovation sequences (Mehra, 1970, 1971; Mohamed and Schwarz, 1999). This adaptation includes estimations of the variance-covariance matrix ($\hat{C}_v$) of the innovation sequence (v), the difference between the noisy measurements z_k and its predicted values $C\hat{x}_k^-$. The number of samples *m* is referred to as window size. $\hat{C}_v$ may be computed through averaging inside a moving window at each time step (Mohamed and Schwarz, 1999):

$$v_k = z_k - C\hat{x}_k^- \tag{3.21}$$

$$\hat{R}_k = \hat{C}_v - CP_k^-C^T \tag{3.22}$$

$$\hat{C}_v = \frac{1}{m}\sum_{i=1}^{m} v_{k-i}v_{k-i}^T \tag{3.23}$$

The outcomes must be positive definite for the innovation based estimation of *R*. This outcome is not guaranteed in the previous approach as two positive definite matrices are subtracted.

3.1.2.2 Residual-based Estimation of R

The residual based estimation of *R*, as proposed by Wang et al. (1999, 2000), can be used to ensure that the estimated R is positive definite. This includes the estimations of the variance-covariance matrix ($\hat{C}_{\bar{v}}$) of the residual sequence $\bar{v}$, the difference between the noisy measurements and its estimated values. Residual based adaptive Kalman filtering is shown below:

$$\bar{v}_k = z_k - C\hat{x}_k \tag{3.24}$$

$$\hat{R}_k = \hat{C}_{\bar{v}} - C\hat{P}_kC^T \tag{3.25}$$

$$\hat{C}_{\bar{v}} = \frac{1}{m}\sum_{i=1}^{m} \bar{v}_{k-i}\bar{v}_{k-i}^T \tag{3.26}$$

3.1.2.3 Adaptive Estimation of Process Noise Covariance Matrix (Q)

Estimation of the process noise covariance matrix Q depends on the measurement noise covariance matrix R, as the estimation of R requires the predicted state covariance P_k^- and hence Q. If R and $\hat{P}_k$ are assumed to be known, Q can be calculated as (Wang et al., 1999; Ding et al., 2007):

$$Q_k = Q_{k-1}\sqrt{\alpha} \tag{3.27}$$

where α is the ratio between the estimated and predicted innovation covariance (Ding et al., 2007).

$$\alpha = \frac{trace\{\hat{C}_v - R_k\}}{trace\{CP_k^- C^T\}} \tag{3.28}$$

3.2 Algorithm

3.2.1 Adaptation of *R*

Figure 3.1 shows the algorithm used in this work for the residual-based estimation of *R*. The state vector and the state error covariance matrix are initialized to calculate $\hat{x}_k^-$ and P_k^- for the next time step. In the time update equations, *u* is known and a good guess is made for *Q* based on the knowledge of discrepancy between the model and the actual process. A series of measurements is used where the window size for measurement is preset to *m* for calculating adaptive *R*. If the current time instant is less than *m* the algorithm follows the steps $i - iii$ in traditional KF with fixed *R*. Once the window size measurements are available the algorithm follows steps $i - vi$ in residual-based adaptation of *R*.

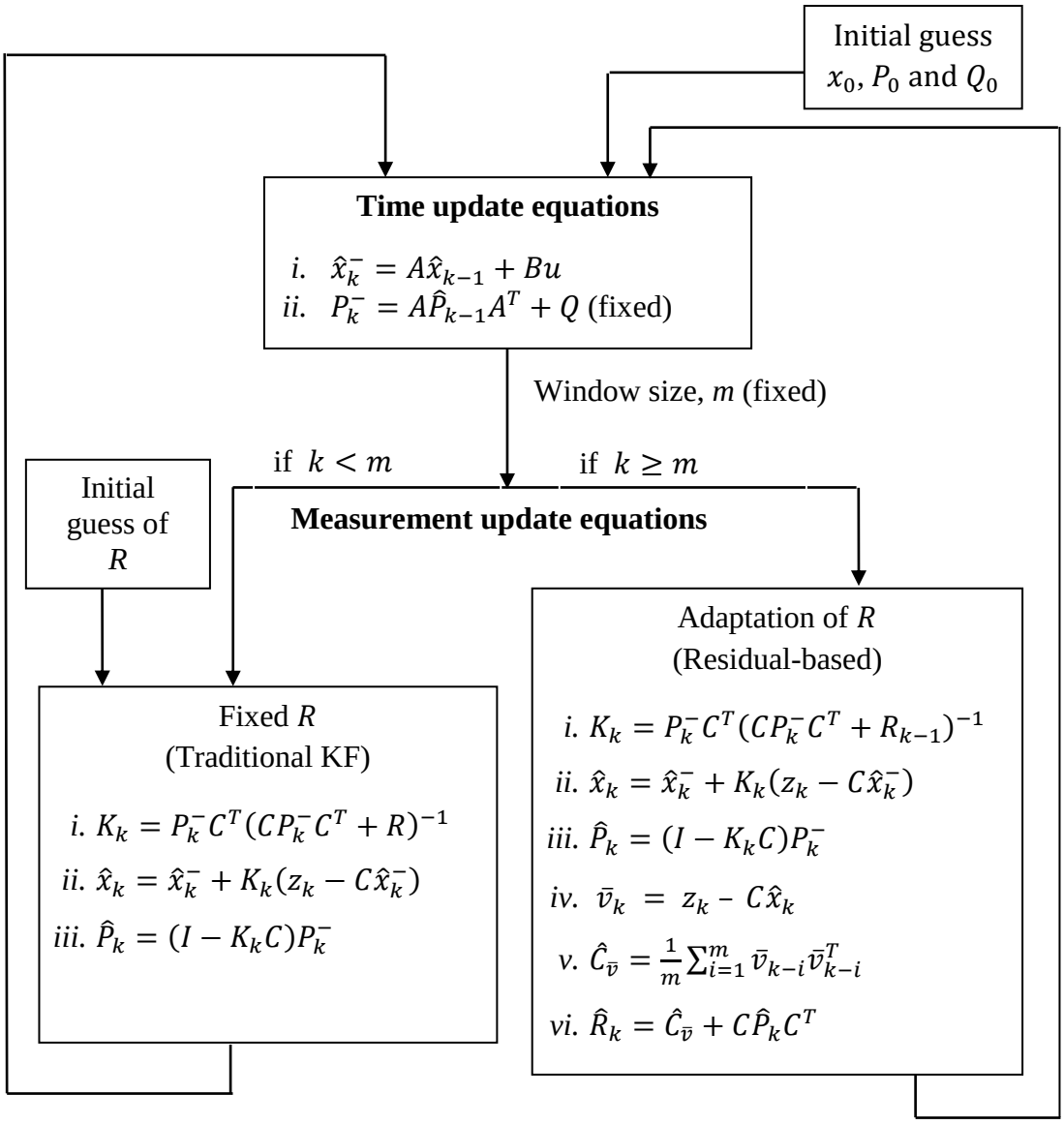

Figure 3.1: Algorithm for residual-based estimation of *R*.

3.2.2 Adaptation of Q

Figure 3.2 shows the algorithm used in this work for adaptive estimation of *Q*. The state is initialized as a zero vector and the state error covariance matrix is initialized as a null matrix. The same value of *Q* as used in the adaptation of *R* is used to make initial guess. The time update equations predict $\hat{x}_k^-$ and P_k^- for the current time step. Predicted state is used to compute innovation sequence. $\hat{C}_v$ is computed through averaging inside a moving window at each time

step. Window size remains variable until the number of sample reaches n. It should be noted that the adaptive estimation of *Q* uses fixed *R* obtained from final iteration of Figure 3.1.

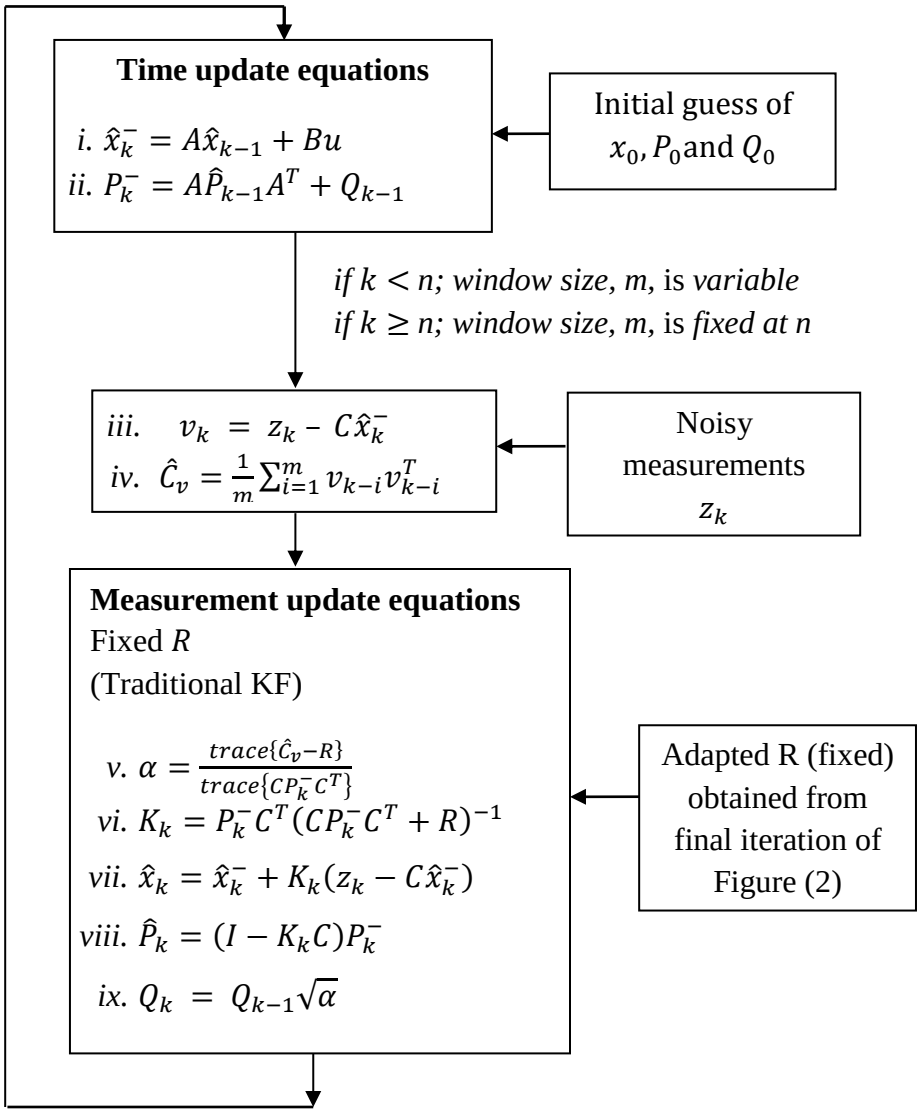

Figure 3.2: Algorithm for adaptive estimation of *Q*.

3.3 Working Approach

Starting with a detailed nonlinear process model of the AGR process in Aspen Plus Dynamics® (APD®), a continuous-time, linear model is generated by running a control design interface (CDI) script that linearizes the nonlinear model around the steady-state operation conditions. The

linear model is discretized in MATLAB® for use in the KF algorithm. The measurement data used in this work are generated by the nonlinear APD process model. The working approach is shown in Figure 3.3.

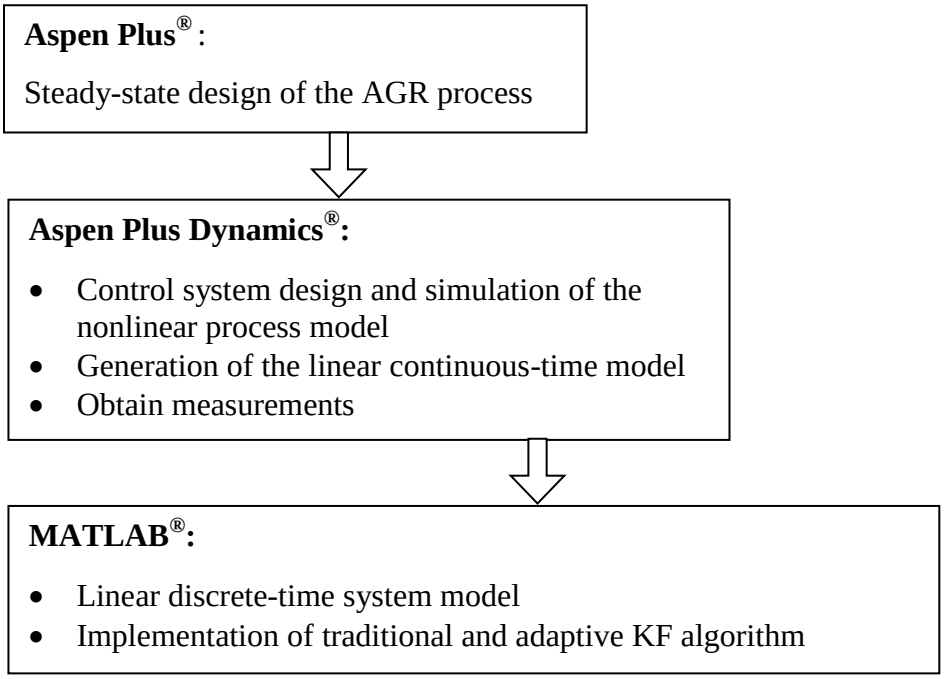

Figure 3.3: Flowsheet of the working approach.

3.4 Case Study

The application of the algorithm presented in Figures 3.1 and 3.2 is illustrated in this study for acid gas removal (AGR) process as part of the IGCC plant with CO_2 capture that has been presented by Bhattacharyya et al. (2011). The AGR process involves extensive heat and mass integration and therefore, is very suitable for the study of the designed estimators in the presence of complex interactions between process variables. Figure 3.4 shows the configuration of the AGR unit and subsequent CO_2 compression system.

The AGR process is a dual-stage unit that is selective to both H_2S and CO_2 capture. Chilled solvent is used to remove H_2S in the first stage followed by a second stage that removes CO_2. Most of the H_2S in the syngas entering the AGR process is absorbed in the semi-lean solvent as it passes through the H_2S absorber. The tail gas from the Claus sulfur capture unit is recycled to the

H_2S absorber. The off-gas from the top of the H_2S absorber is sent to the CO_2 absorber. A portion of the loaded solvent (about 30% in the base case) from the bottom of the CO_2 absorber is chilled, and sent to the H_2S absorber. The remaining portion of the loaded solvent from the bottom of the CO_2 absorber is heated and then flows through the H_2 recovery drum. After that it goes through a series of three flash vessels, high pressure (HP), medium pressure (MP), and low pressure (LP), to recover CO_2 for compression in preparation for storage or sequestration. The semi-lean solvent leaving the LP flash vessel is cooled by exchanging heat with the loaded solvent and is then chilled before returning to the CO_2 absorber. The rich solvent from the bottom of the H_2S absorber is heated and then sent to a flash vessel. The vapor from the flash vessel is recycled back to the H_2S absorber. The bottom stream from the flash vessel goes to the solvent stripper. Make-up solvent is mixed with the stripped solvent and sent to the top tray of the CO_2 absorber.

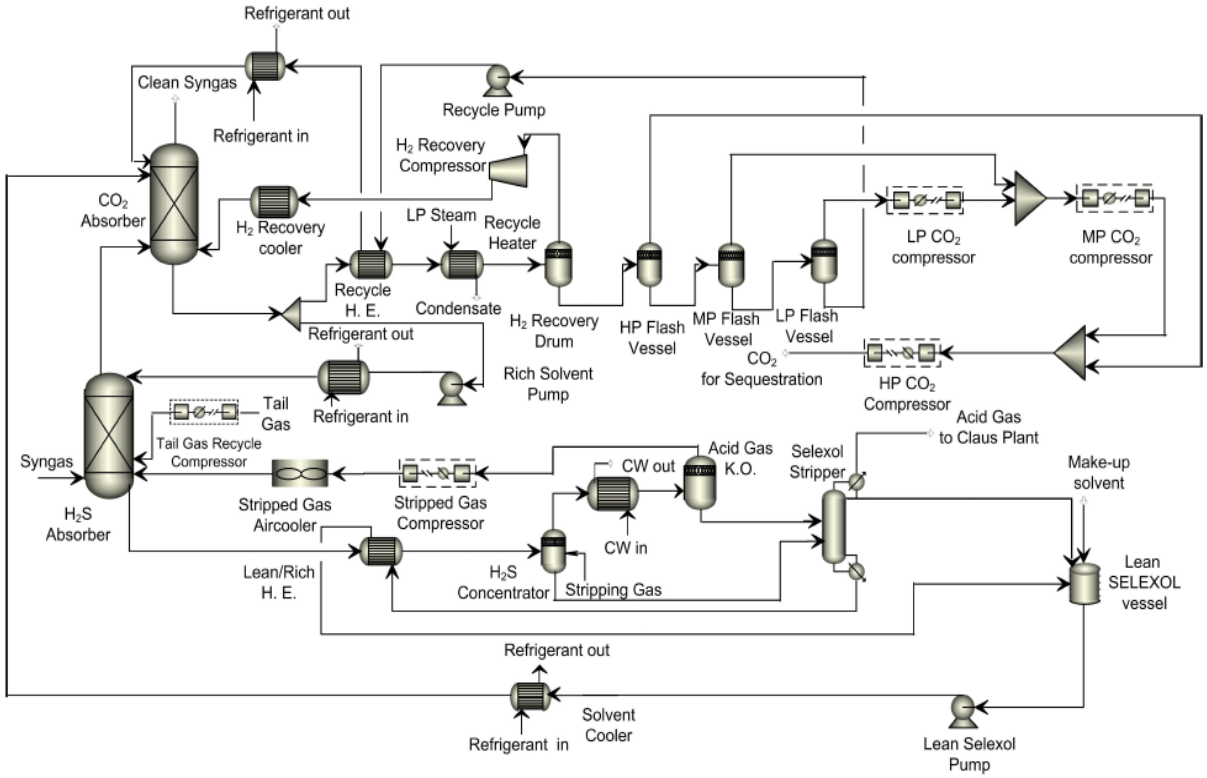

Figure 3.4: Configuration of the AGR and CO_2 compression units (Bhattacharyya et al., 2011).

In the results presented below, focus was the H_2S absorber as shown in Figure 3.4. The H_2S absorber model consists of 332 state variables, 2 input variables (pressure of the feed gas, and opening of the feed gas valve), and 10 output variables (H_2S and CO_2 molar concentration in the 15^{th} and 20^{th} stages of the H_2S absorber, flowrate of the top and bottom outlet streams from the H_2S absorber, H_2S and CO_2 molar concentration in the top and bottom outlet streams from the H_2S absorber).

3.5 Results and Discussion

All the results presented below are generated by introducing a 5.5% step increase in the feed gas flowrate to the H_2S absorber.

3.5.1 Adaptive *R*

The innovation-based estimation of R is discarded as the estimated R is found to be negative definite in certain cases for this process. The results, shown in Figures 3.5-3.6 in terms of deviation variables, are based on the residual-based adaptive estimation of R. In this study, Q is manually tuned and then kept constant for all the cases presented in Figures 3.5-3.6. For the adaptive KF, the initial R is same as the R used in the traditional KF. In these studies, the initial R is chosen such that the performance of the adaptive KF can be tested where the initial guess of R is poor. Figures 3.5 and 3.6 show the performance of the traditional as well as the adaptive KF for estimating CO_2 and H_2S composition in the top outlet stream from the H_2S absorber. In both figures, the traditional KF fails to filter out the noise while the estimation of the adaptive KF matches nearly perfectly with the actual, noise-free data from the nonlinear process model.

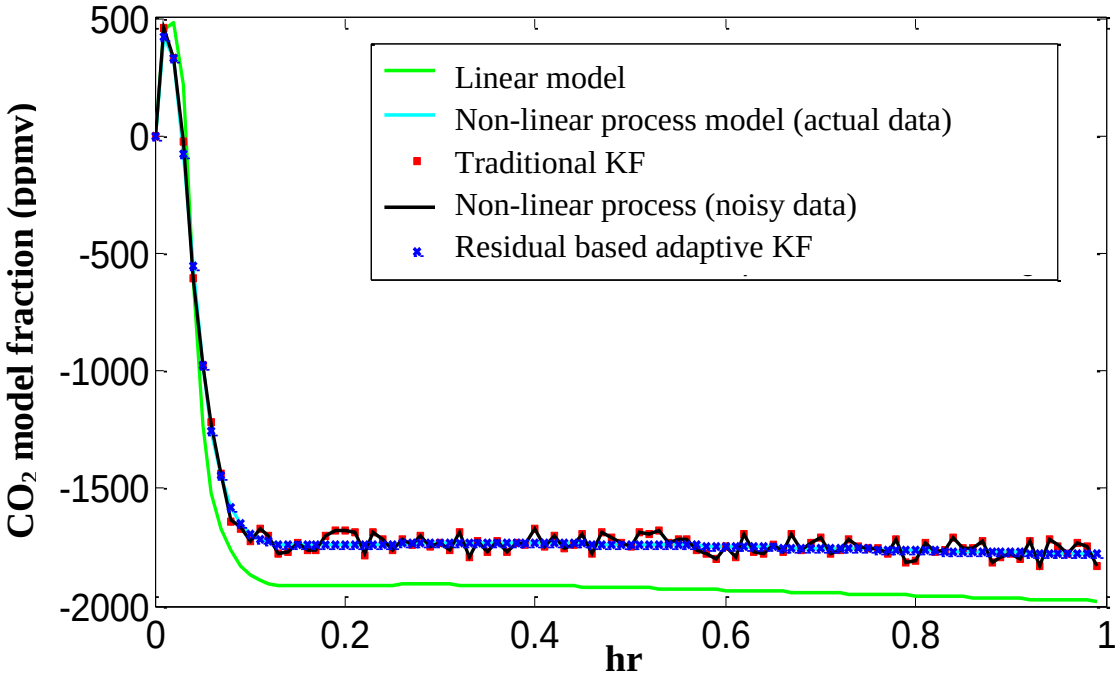

Figure 3.5: Comparison of the filter estimates for the CO_2 composition (deviation variable) in the H_2S absorber top outlet stream.

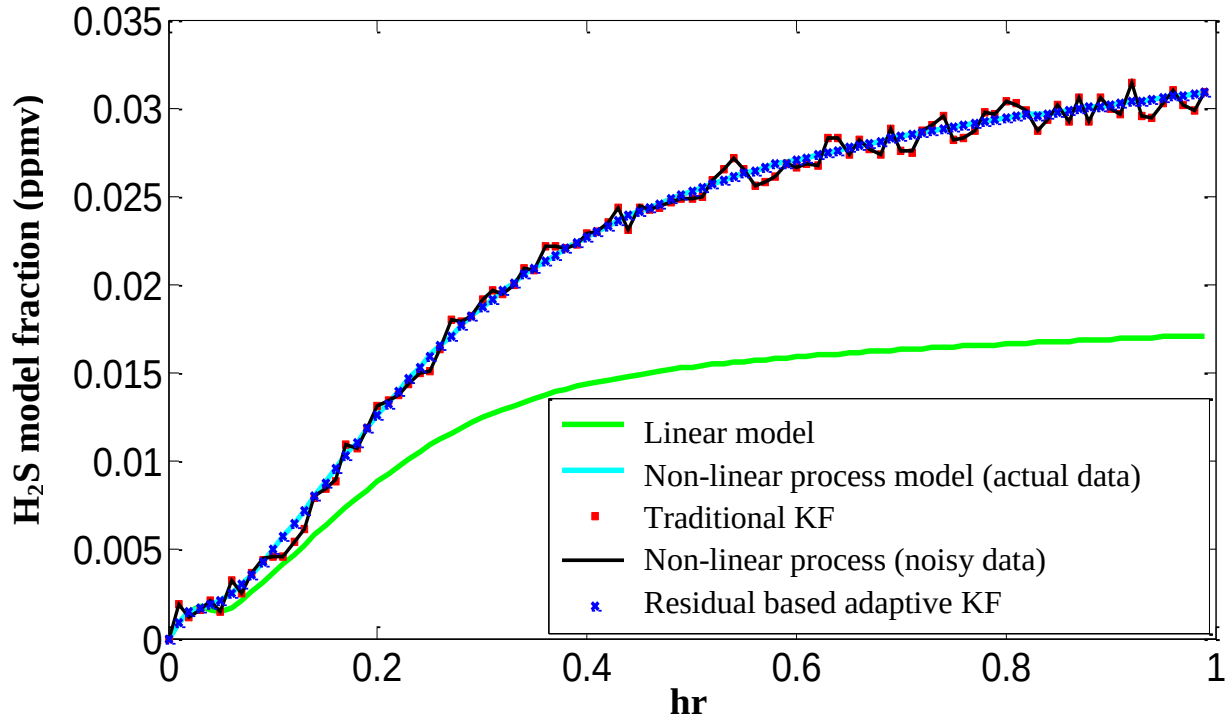

Figure 3.6: Comparison of the filter estimates for the H_2S composition (deviation variable) in the H_2S absorber top outlet stream.

3.5.2 Adaptive Q

Figures 3.7-3.8 show the results when Q is adapted. Here, R is set constant to the value obtained from the final iteration of the residual-based adaptive estimation of R. For the adaptive KF, the initial Q is the same as the Q used in the traditional KF. In these studies, the initial Q is chosen such that the performance of the adaptive KF can be tested where the initial guess of Q is poor. Figure 3.7 compares the estimation from the adaptive and traditional KFs for the flowrate of the H_2S absorber bottom stream. The estimation from the adaptive KF matches very well with the actual data; whereas, the traditional KF with constant Q mainly follows the linear model.

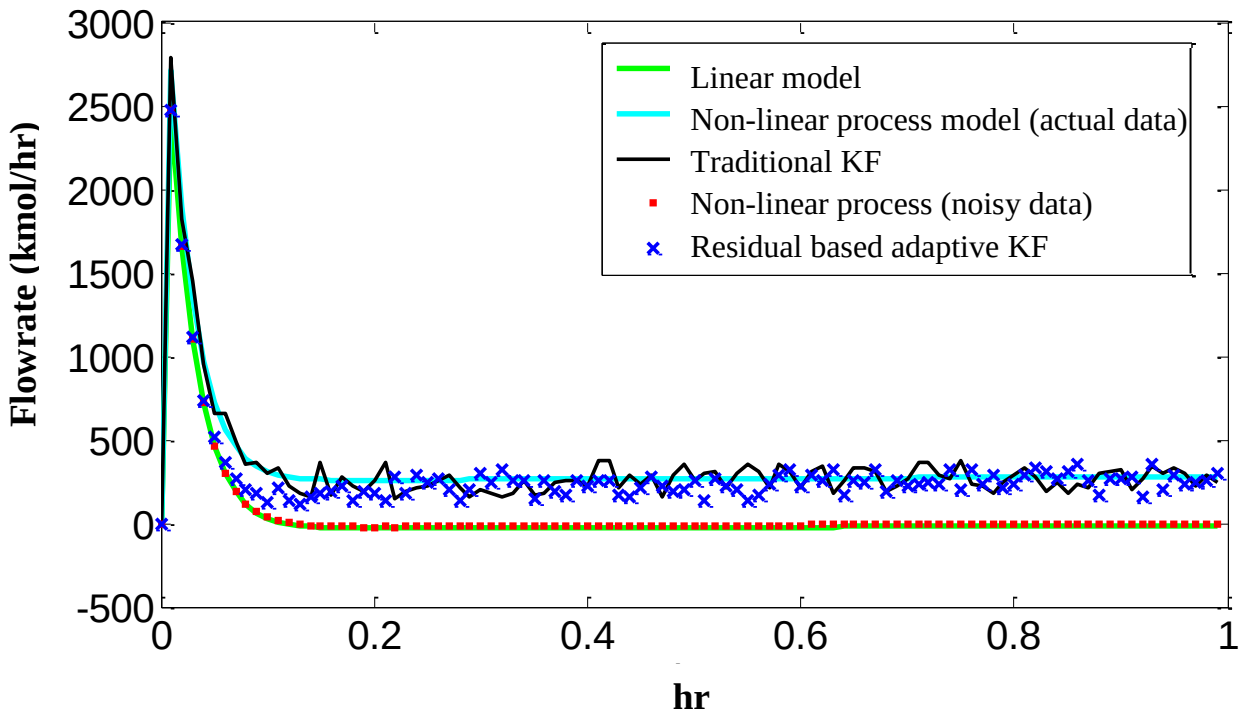

Figure 3.7: Comparison of the filter estimates for the H_2S absorber bottom stream flowrate (deviation variable).

Figure 3.8 compares the estimation of the H_2S mole fraction in the H_2S absorber bottom stream. Even though, the adaptive KF initially follows the linear model, its estimation is much superior to the traditional KF that mainly follows the linear model.

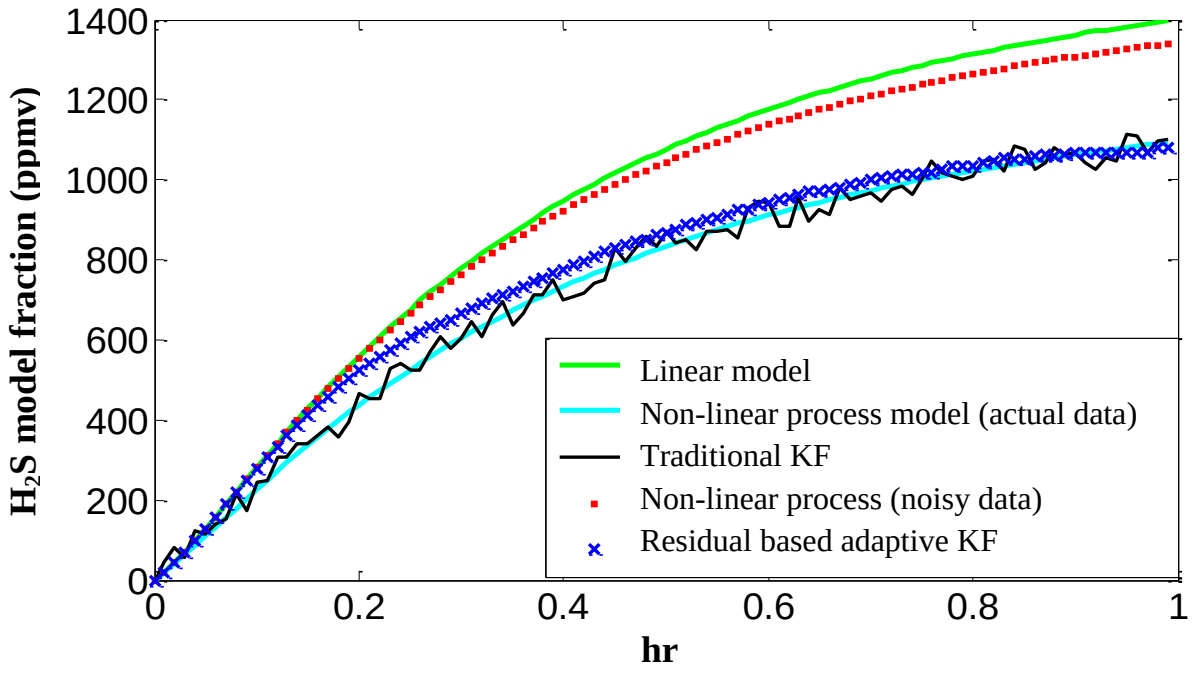

Figure 3.8: Comparison of the filter estimates for the H_2S composition (deviation variable) in the H_2S absorber bottom stream.

3.6 Conclusion

In this chapter, the key variables that capture the environmental performance of an AGR process are estimated by adapting *R* and *Q* separately. *R* is first adapted based on the residual sequence and then it is used to adapt *Q*. Results show that the estimation accuracy of the adaptive KF is much superior to the traditional KF. The adaptive KF estimates the key performance variables very accurately, even in the presence of a high noise-to-signal ratio and large mismatches between the linear and nonlinear process models.

Chapter 4

Steady State SND Algorithm

SND is a constrained optimization problem requiring systematic and effective solution algorithms for determining where best to locate sensors. In this chapter, a SND algorithm is developed for maximizing plant efficiency for an estimator-based control system while simultaneously satisfying accuracy requirements for the desired process measurements. The SND problem formulation leads to a mixed integer nonlinear programming (MINLP) optimization that is difficult to solve for large-scale system applications. A simultaneous solution approach is described where all the constraints are satisfied at the same time. Thus simultaneous approach is not appropriate for problems involving thousands of process variables. Therefore, a sequential approach is developed to solve the MINLP problem where the integer problem for sensor selection is solved using the genetic algorithm while the nonlinear programming problem including convergence of the 'tear stream' in the estimator-based control system is solved using the direct substitution method. The SND algorithm is then successfully applied to a large-scale, highly integrated chemical process.

4.1 Literature Review

Most of the SND algorithms that have been presented in the existing literature have considered static process conditions. These algorithms will be called steady-state SND (SSND) algorithms. Some of the most popular works in the area of SSND have been reviewed in Bagajewicz (2000). Among the earlier works on SSND are those by Vaclavek and Loucka (1976), Kretsovalis and Mah (1987), and Madron and Veverka (1992). A linear steady state process was used by Ali (1993) and Ali and Narasimhan (1993, 1995 and 1996) to introduce the concept of reliability for sensor placement. Some of the recent works are the design of non-redundant observable linear sensor networks (Carnero et al., 2001, 2005) and redundant sensor network (Nabil and Narasimhan, 2012) for minimizing the loss of operational profit due to measurement error.

To the best of our knowledge, there is no SND algorithm in the existing literature for maximizing process efficiency. The main difference in the SND algorithm for maximizing efficiency is due to consideration of the closed loop system. It should be noted that all existing SND algorithms have been developed considering open-loop systems, i.e. the performance of the sensor network does not affect the process. In a closed-loop system, the measurements from the sensor network passes through the estimator, and the controllers take action based on the estimated values. This, in turn, affects the process and therefore the measured variables. As a result the estimator output and the control action differ from before. This continues until the process reaches its new steady state.

Due to the feedback loop in the control system, the resulting system of equations becomes very difficult to converge for any arbitrary set of integer variables (i.e., set of sensors). A sequential optimization algorithm is developed that follows the infeasible path method where a 'tearing' approach is used to solve the feedback loops. The methodology is developed in a way that large-scale systems can be solved efficiently.

In this work, the integer programming problem is solved by GA while other linear and nonlinear constraints are satisfied by a sequential equation solver using a 'tear' stream approach. As discussed below in more detail, this formulation helps in satisfying the linear and nonlinear equality constraints for every combination of integer variables.

Many of the SND algorithms in the existing literature have been applied to small simplified test problems. In this work, the developed methodology is applied to a large, highly integrated acid gas removal (AGR) unit as part of an integrated gasification combined cycle (IGCC) power plant with CO_2 capture. This AGR unit comprises of a number of typical unit operations involving considerable mass and energy integration and, therefore, is a very good industrial case study for the proposed algorithm.

The following organization is adopted in the rest of this chapter. First the SND algorithm for efficiency maximization for an estimator-based control system is developed. This is followed by

a discussion of the solution approaches to the SND problem. Finally, the application of the SND algorithm to the AGR case study is presented.

4.2 Development of the SSND Algorithm

Figure 4.1 shows the estimator-based control system that is used to develop the SSND algorithm. Perturbed by a disturbance, u_d, the estimator receives the noisy measurements, $y_{noisy,\beta}$, from the sensor network and estimates the process variables of interest for use in control ($\hat{y}_{cont,est}$) and monitoring ($\hat{y}_{mon}$). The controller(s) then implement(s) the corrective action on the process based on the estimated controlled variables.

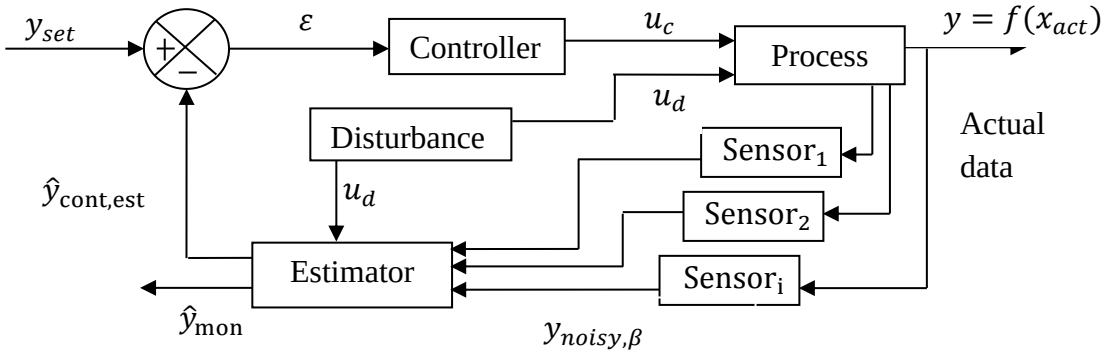

Figure 4.1: Schematic of the estimator-based control system for development of the SSND algorithm.

For developing the SSND algorithm, first the set of equations corresponding to each block of the estimator-based control system is organized. The estimator block in Figure 4.1 is considered to be a continuous Kalman filter. The process and measurement models as appear in Eq. (3.1) and (3.3) are rewritten in terms of actual state

$$\frac{dx_{act}}{dt} = Ax_{act} + Bu + w \tag{4.1}$$

$$y_{noisy} = Cx_{act} + v \tag{4.2}$$

In Eq. (4.1), A $(n \times n)$ and B $(n \times m)$ are the constant nonsingular transition matrix and input matrix, respectively. Eq. (4.2) defines the relationship between the measurement vector (y_{noisy})

and the state vector (x_{act}). C $(l \times n)$ is the measurement matrix. The mismatch between the nonlinear process and the linear state space model is captured by the random variable w, typically known as the process noise vector. The random variable v in Eq. (4.2) represents measurement noise. Process noise (w) and measurement noise (v) are assumed to be uncorrelated, Gaussian, white noise sequences with zero-mean.

Traditional KF is used to estimate the states and disturbances of the process. Table 4.1 shows the equations that characterize the closed loop blocks: estimator, comparator, and controller. Linear differential equations (Eq. 4.3) are used to estimate the states $\hat{x}$ of the controlled variables and other key performance variables of the process in the presence of noisy measurements, y_{noisy} and Kalman gain, K. Kalman gain can be obtained by first integrating the nonlinear matrix differential Riccati equation (Eq. 4.4) for the state covariance matrix, P, and then solving the matrix equation (Eq. 4.5) for the Kalman gain. Q and R are kept constant during state estimation. Since the tuning parameters are unknown, a good guess is crucial for both of them.

Table 4.1: Equations Characterizing the Estimator, Comparator and Controller Block (in Figure 4.1)

<table>
<tr><th>Estimator (KF)</th><th>Comparator</th></tr>
<tr><td>$\frac{d\hat{x}}{dt} = A\hat{x} + Bu + K(y_{noisy} - C\hat{x})$ (4.3)

$\frac{dP}{dt} = -PC_{\beta}^{T}R^{-1}C_{\beta}P + PA^{T} + AP + Q$ (4.4)

$K = PC_{\beta}^{T}{R_{\beta}}^{-1}$ (4.5)</td><td>Estimated measurements:
$\hat{y}_{cont,est} = C_{cont}\hat{x}$ (4.6)
Error function:
$\varepsilon(t) = y_{cont,set} - \hat{y}_{cont,est}$ (4.7)

Controller (proportional- only)

Control action: $\frac{du_c}{dt} = K_c\frac{d\boldsymbol{\varepsilon(t)}}{dt}$ (4.8)</td></tr>
</table>

The comparator receives the estimated measurements (Eq. 4.6) and compares them with the set point of the controlled variables of interest and calculates the error functions, $\varepsilon(t)$ (Eq. 4.7). In

this case, a proportional-only (P) controller has been assumed mainly for simplicity. Eq. (4.8) shows the time variant proportional control action, where K_c is the proportional gain.

For simplicity and reduction of computational expense, the SSND algorithm is developed under steady-state assumptions. In addition to the steady-state versions of the equations shown in Table 4.1, the inequality constraints shown in Eqs. (4.9) and Eq. (4.10) are also considered. In Eq. (4.9), parameter b denotes the budget ($) for the sensor where the left side of the inequality represents total cost of placed sensors for obtaining measurements and c_i denotes the cost of individual sensor i.

$$\sum_{\forall i} c_i \beta_i \leq b \qquad \forall i \in N_s \tag{4.9}$$

$$|y_{ma,act} - C_{ma}\hat{x}| < tol_1 \tag{4.10}$$

β_i takes on a value of 1 if a sensor is placed to measure the process variable, otherwise it is 0. In Eq. (4.10), tol_1 is the tolerance limit vector on the estimation error and the left side of the inequality is the vector of actual minus vector of estimated value ($C_{ma}\hat{x}$) of process monitoring variables as well as active constraints. The objective function is defined as the deviation of the actual efficiency of the plant from the optimal efficiency. The optimal efficiency, η_{opt}, is the maximum efficiency when the plant runs under optimal operating conditions with no estimator and measurement errors. Therefore, η_{opt} is the maximum efficiency that can be attained. The efficiency of the process in the presence of estimator-based control system is denoted by $\eta(x_{act}, \beta)$. It should be noted that one down-side of the steady-state assumption is that the KF is essentially being used to estimate steady-state bias.

The SSND objective is to maximize $\eta(x_{act}, \beta)$ for a given budget for sensors. This is equivalent to the minimization of the squared error between the maximum efficiency and the actual efficiency of the plant with the sensors in place. Therefore, after some substitutions and rearrangement, the SSND problem is given by:

$$
\begin{aligned}
& Min(\eta_{opt} - \eta(x_{act}, \beta))^2 \\
& \text{s.t.} \\
& Ax_{act} + Bu + w = 0 \\
& C_\beta = \left[C_{ij}\right]_{\beta_i \neq 0} ; \quad v_\beta = [v_i]_{\beta_i \neq 0} \\
& i = 1, 2, \ldots \ldots l; \;\; j = 1, 2, \ldots \ldots n \\
& y_\beta = C_\beta x_{act} + v_\beta \\
& AP + PA^T - PC_\beta^T {R_\beta}^{-1} C_\beta P + Q = 0 \\
& K = PC_\beta^T {R_\beta}^{-1} \\
& A\hat{x} + Bu + K\left(y_\beta - C_\beta \hat{x}\right) = 0 \\
& \textstyle\sum_{\forall i} c_i \beta_i \;\leq\; b \\
& \beta_i = 0,1 \qquad \forall i \in N_s \\
& y_{ma} - \; C_{ma}\hat{x} < tol
\end{aligned}
\tag{4.11}
$$

N_s is the set of all candidate sensors. In this formulation, the variable y_{noisy} in Eq. (4.2) is replaced by y_β as the set of available measurements. C_β is the measurement matrix of the available sensors and the corresponding measurement noise is v_β. It should be noted that β is the set of integer variables while the remaining variables are continuous. Therefore the SSND is a mixed integer nonlinear programming (MINLP) problem which can be solved by two solution approaches: a simultaneous solution approach or a sequential solution approach.

4.3 Solution Approach

4.3.1 Simultaneous Solution Approach

In the simultaneous approach, all the constraints are satisfied at the same time. However, the developed algorithm has a large number of variables including the actual states, outputs, estimated states, and the elements of the state covariance matrix, P. If there are *n* state variables and $\sum \beta_i$ integer variables, then the total number of continuous and integer variables in the SSND problem is $2n + 2\sum \beta_i$. In addition, the solution of $n \times n$ matrix Riccati equations and

computation of steady state Kalman gain matrix of identical dimension results in extensive computational complexity. In the case of a very large size problem involving more than a thousand states, this approach becomes computationally very expensive. Furthermore, it becomes very difficult to make an initial guess for the continuous variables, especially for *P* for every possible combination of integer variables (i.e. selected sensors). A bad initial guess can result in high computational expense and in the worst case can lead to failure. One typical approach to solve the MINLP problem is to separate the integer programming (IP) problem from the nonlinear programming (NLP) problem. But again, the convergence of the NLP problem is extremely difficult because of the reason mentioned above. Based on our extensive testing of a number of case studies, this approach is found to be suitable for small problems with very few states and candidate sensor locations. Since our objective is to apply the SSND algorithm to large systems, this approach was not pursued further. Instead, a sequential modular approach described in the following section was developed.

4.3.2 Sequential Solution Approach

In this approach, the MINLP problem for SSND is solved by solving the IP problem by GA while the NLP problem is solved sequentially as described below.

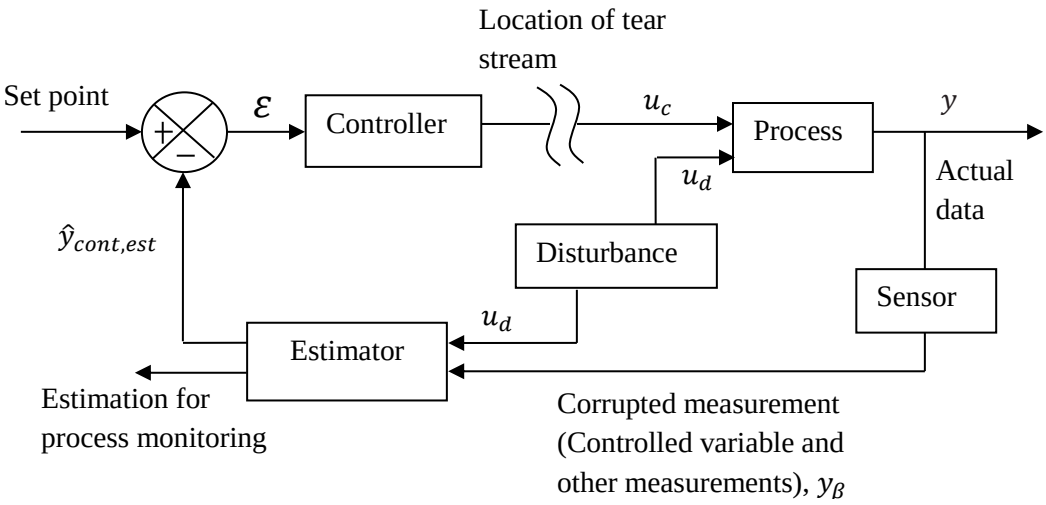

Figure 4.2: Sequential solution approach to the SND problem for the estimator-based control systems.

The proposed sequential approach is similar to the sequential modular approach used for solving process flowsheeting problems. In this approach, each estimator-based control loop is opened by tearing a stream and then the blocks are solved sequentially until a convergence criterion is satisfied. The proposed approach is shown in Figure 4.2. While the tear stream location can be any location such that the loop is opened, the location chosen in this work, as shown in Figure 4.2, helps to reduce the total number of tear variables and helps to generate initial guesses for the tear variables (i.e. the inputs).

Since the NLP problem is solved sequentially, the objective function in the GA is modified to introduce the penalty term for the estimation error in the process monitoring variables and active contraints. Figure 4.3 shows the algorithm for the sequential solution approach.

The SSND algorithm is developed under the assumption of perfect implementation of control action in the feedback control loop, i.e., implementation error due to the actuator and any associated hardware/software is neglected. The developed algorithm is solved using GA. The flowsheet in Figure 4.3 starts with the specification of GA parameters and proceeds with the creation of an initial population in the first generation (denoted by Gen in Figure 4.3). Each solution set β in the population consists of decision variables, i.e., locations of sensors. A counter is set to reduce excessive computational time for those solution sets that fail to satisfy the convergence criterion. The estimator-based feedback control loop starts with $counter = 0$ and an initial guess for the tear stream (u_c). The initial guess is generated by considering the process model and assuming perfect control and measured disturbances. Input, u, is obtained by augmenting the disturbance vector, d, with u_c. In Step 1, the process model is solved to calculate the actual states, x_{act}, using the augmented vector u. In Steps 2 and 3, those rows of the measurement matrix and measurement noise vector that correspond to $\beta_i = 0$ (ith row) are rejected. As a result, the dimension of C_β and v_β reduces to $\sum \beta_i \, by \, n$ and $\sum \beta_i \; by \; 1$, respectively. This is followed by Step 4, where noisy measurements are obtained by using the linear algebraic measurement equations and adding measurement noise v_β. Step 5 involves solving the algebraic Riccati equation to obtain the process noise covariance matrix (P) and it is followed by the calculation of the steady-state Kalman gain matrix (K) in Step 6. Once the steady-state gain is available, the estimated states, $\hat{x}$, can be computed in Step 7 that are then

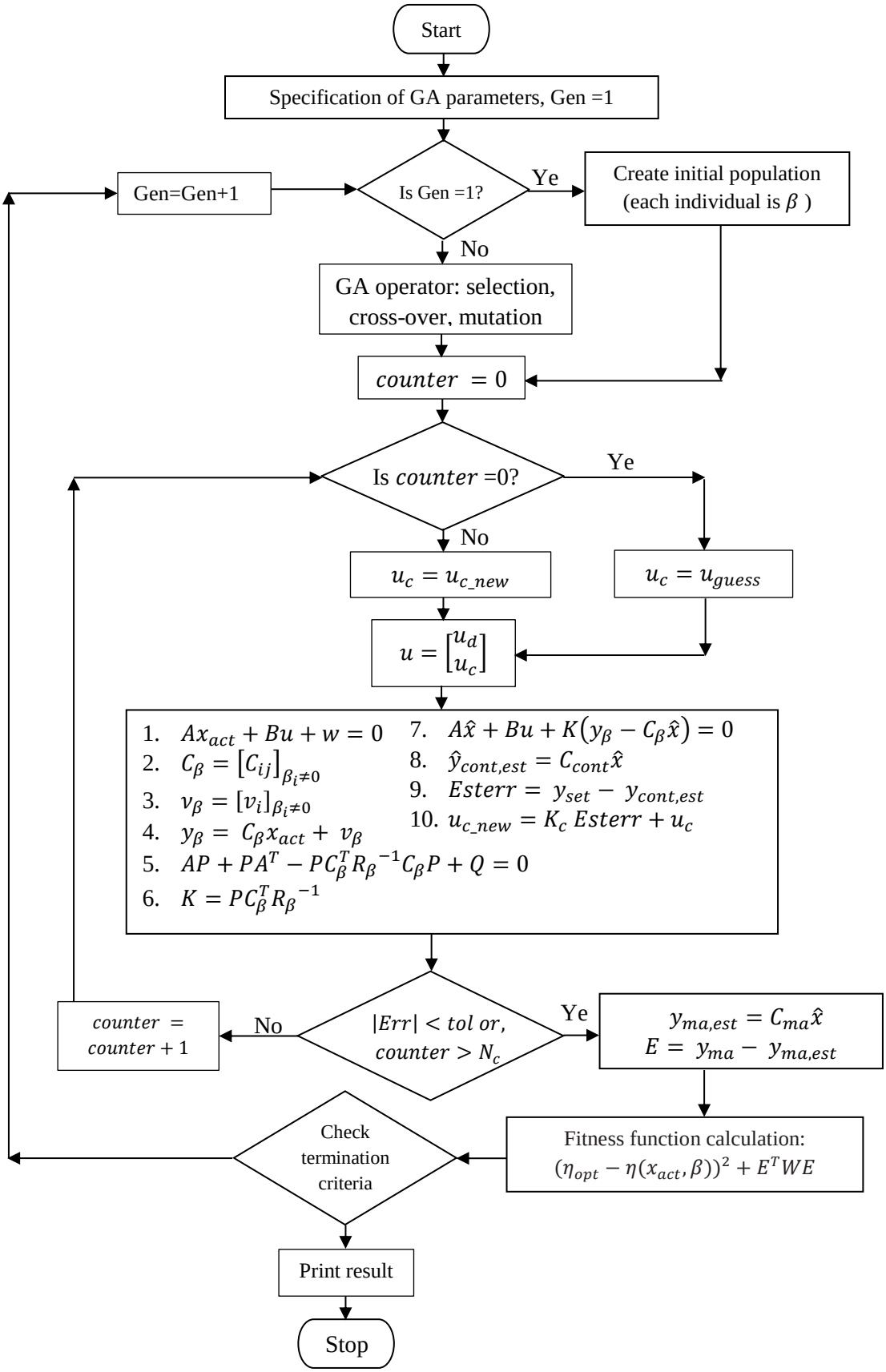

Figure 4.3: Algorithm to simulate feedback control system with an estimator.

used in Step 8, to obtain estimated controlled variables, $\hat{y}_{cont,est}$. In the Step 9, the error is determined from the difference between the setpoint and the estimated controlled variables. Based on this error, in Step 10, the P-only controller computes necessary control action u_{c_new}. Until $|Err|$ satisfies the tolerance or the counter is less than the pre-specified number for the iteration loop, the entire computation loop is repeated with the updated u_c. It should be noted that when $|Err|$ satisfies the tolerance and the tolerance is set at a low enough value, the solution represents the steady state of the entire system. The steady-state solution for a particular candidate set of sensors is achieved after a number of iterations. Then the feasible candidate set is assigned a fitness value based on the objective function. The infeasible set of sensors that does not satisfy the estimation accuracy in monitoring variables as well as in active constraints penalizes the objective function by adding $E^T WE$ where W is a weighting factor. The GA continues until the convergence criterion is satisfied.

The SSND algorithm presented here uses the 'direct substitution' method for tear stream convergence. For highly interacting systems, other algorithms for tear stream convergence such as Broyden's method or Newton's method might be necessary.

4.4 Genetic Algorithm

The GA is based on the principle of biological evolution (Haupt, 2004). GA creates the initial solution sets and ranks them according to their fitness value. Solution sets with higher fitness values survive and act as parents to produce children for the next generation. Breeding is performed based on the pre-specified cross-over, mutation, and selection criteria. Over successive generations, the population "evolves" toward an optimal solution. The proposed SSND problem is very suitable for the GA because:

- The problem is a combinatorial optimization problem.
- The GA can handle the inequality constraints with mixed integer linear programming problem.
- The SSND problem is expected to have many extrema and therefore, a global search is necessary.

4.5 Case Study

This section illustrates the application of the proposed SSND methodology to a large-scale chemical process unit, specifically a selective, dual-stage, chilled Selexol^{TM} solvent-based acid gas removal (AGR) unit as shown in Figure 4.4. A short description of the AGR unit is provided in section 3.4 of previous chapter. Interested readers are referred to Bhattacharyya et al. (2011) for detail description.

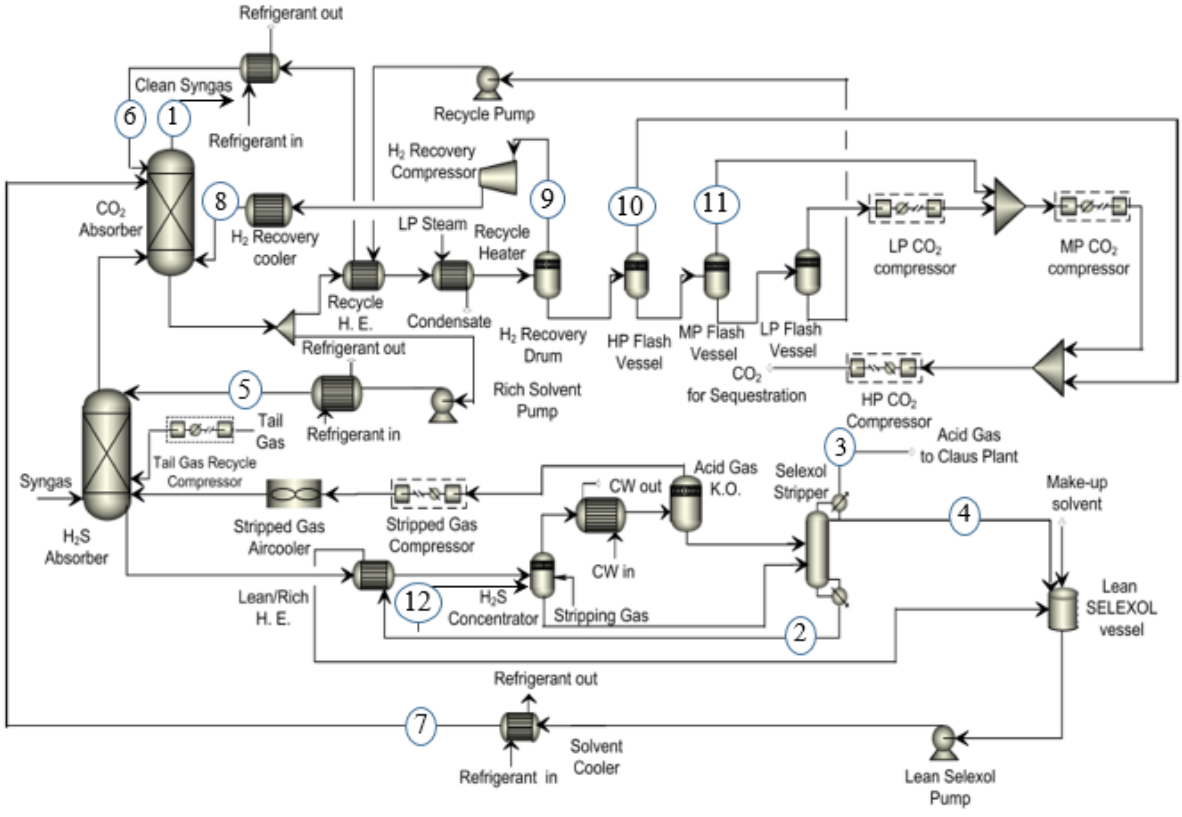

Figure 4.4: Locations of primary control variables (labeled from 1-12) in AGR unit (Bhattacharyya et al., 2011) considered for implementation of SSND algorithm.

For evaluating performance of AGR processes, usually measures such as \$/tonne CO_2 captured or avoided is considered (Bhattacharyya et al., 2011). The dollar cost includes both operating and capital costs. From the sensor placement perspective, since we are mainly interested in the operating costs, amount of CO_2 captured per unit power consumption is considered to be the

measure for efficiency of this AGR process. Thus $\eta(x_{act}, \beta)$ is defined for the AGR unit by the following equation:

$$\eta(x_{act}, \beta) = \frac{F_{CO_2,in}(x_{act}) - F_{CO_2,out}(x_{act})}{aF_{solvent}(x_{act}) + \sum_{c=1}^{3} P_c(x_{act})} \quad (4.12)$$

The numerator in Eq. (4.12) represents the amount of CO_2 captured while the denominator is the MWh power consumption. The variables in Eq. (4.12) are a function of x_{act} in the estimator-based control system.

The primary controlled variables for the AGR process have been identified by Jones et al. (2014) and the locations are labeled from 1-12 in Figure 4.4.

Table 4.2: List of Primary Controller Variables and Pairings (Jones et al., 2014)

	Active Constraints		**Manipulated Variables**
1.	CO_2 Capture	1.	Low Pressure Flash Pressure
2.	Water Content of Solvent at Stripper Bottom	2.	Steam Flowrate
3.	Stripper Pressure	3.	Stripper Vapor Flowrate
4.	Stripper Top Temperature	4.	Stripper Condenser Duty
5.	Semi-lean Solvent Cooler Outlet Temperature	5.	Semi-lean Solvent Cooler Duty
6.	Loaded Solvent Cooler Outlet Temperature	6.	Loaded Solvent Cooler Duty
7.	Lean Solvent Cooler Outlet Temperature	7.	Lean Solvent Cooler Duty
8.	H_2 Cooler Outlet Temperature	8.	H_2 Cooler Duty
	Self-optimizing Controlled Variables		**Manipulated Variables**
9.	Pressure of the H_2 Recovery Unit	9.	H_2 recovery outlet valve position
10.	Pressure of the HP Flash Vessel	10.	HP Compressor Brake Power
11.	Pressure of the MP Flash Vessel	11.	MP Compressor Brake Power
12.	N_2 Flowrate to H_2S Concentrator	12.	Valve opening of the N_2 feed valve

The list of the primary control variables is given in Table 4.2. An estimator-based control system has been implemented for active constraints and self-optimizing controlled variables as shown in Table 4.2. The interested reader is referred to Jones et al. (2014) for details of the primary controlled variables and their selection method.

There are a number of operational constraints in the AGR unit considered as process monitoring variables and estimation accuracy must be satisfied for these variables. In this framework, it is easy to include more monitored variables for which estimation accuracy must be satisfied. However, in this case, for simplicity and testing, only two variables are considered for process monitoring purposes as shown in Table 4.3.

Table 4.3: List of Process Monitoring Variables

Constraint	Value
Maximum Allowable Solvent Temperature	175°C
Minimum Stripper Pressure	276 kPa

Figure 4.5 is the block diagram of the workflow based on the three different software platforms used, namely Aspen Plus® (AP), Aspen Plus Dynamics® (APD) and MATLAB®

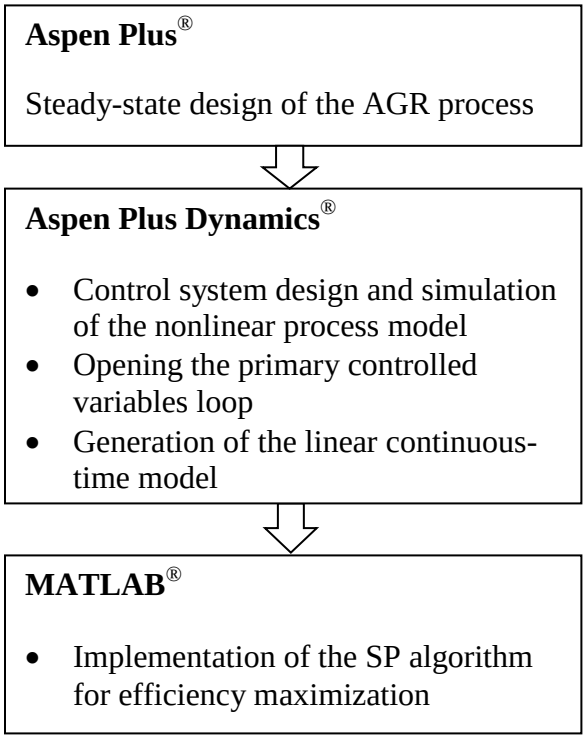

Figure 4.5: SSND workflow

AP® has been used to develop the steady-state process model of the AGR unit. The model is then exported to APD® for designing the control system and obtaining a stable dynamic model. Details about this model can be found in the works of Bhattacharyya et al. (2011) and Jones et al. (2014). Starting with this nonlinear process model of the AGR process in APD®, a continuous-time, linear model is generated by running a control design interface (CDI) script that linearizes the nonlinear model around the steady-state operation conditions. All the primary controlled variable loops are kept open during the linearization of the model. The linear state-space model of the AGR unit, the algebraic measurement equations for candidate sensor locations, the primary controlled variables, and variables that appear in the objective functions are then exported to MATLAB®. It should be noted that even though all primary controlled variables have to be estimated by the measurement framework, sensors are not necessarily placed on all primary controlled variables. This is because measurement of some of these variables can be difficult and/or expensive and can have time delay, high noise, and/or low estimation accuracy. On the other hand, it may be possible to estimate these variables satisfactorily by placing sensors elsewhere in the process and within the budget constraint. The optimal selection is done by the SSND algorithm. In addition, satisfactory estimation of all other variable used for estimation purposes is desired. Therefore, the candidate sensor locations include other variables in addition to the primary controlled variables. The SSND algorithm described above is implemented in MATLAB®.

The AGR process contains 1505 states. Two disturbances, including the change in the syngas flowrate and CO_2 concentration at the inlet of the AGR unit, have been considered. The flowrate disturbance is simulated by changing the inlet pressure of the syngas to the AGR unit. Four different types of commonly used sensors have been considered. These are temperature, pressure, flow, and composition (CO_2 and H_2S) sensors. They are denoted by *T*, *P*, *F*, y_{H_2S} and y_{CO_2}, respectively in the discussion below. The process flowsheet of the AGR unit is reviewed and the candidate sensor locations are identified based on the criteria mentioned below:

For columns including the H_2S and CO_2 absorbers, solvent stripper and H_2S concentrator, candidate *T*, *P*, y_{H_2S} and y_{CO_2} sensor locations are shown in Table 4.4. As all columns are

modeled using an equilibrium-stage assumption, temperatures of the liquid and vapor phases leaving a stage are the same. Therefore, only one *T* and *P* are considered for all these trays.

1. For the heat exchangers, *T* and *P* are measured at both inlet and outlet. Since the mass/molar flowrate and composition does not change across the heat exchangers in the AGR unit, these variables are measured only at the outlet.
2. For each recycle stream, one flow meter is considered.
3. In all mixer blocks, no pressure drop has been considered. Therefore *T*, *F*, y_{H_2S} and y_{CO_2} vary, but *P* is constant.
4. For splitter blocks, *F* changes, but *T*, *P*, y_{H_2S} and y_{CO_2} are constant.
5. Across the pump and valve, only *P* changes. There may be some changes in the temperature but that is neglected.
6. For compressors, both *P* and *T* change across the compressor.

After this analysis, 169 measurements are identified as potential locations for sensor placement. Tables 4.4 and 4.5 show the distribution of these candidate sensors in the AGR unit.

Table 4.4: Candidate Sensor Locations in the Equipment Items in the AGR Unit

Equipment	**Sensors**	
H_2S Absorber	T_2, T_8, T_{14}, T_{20}, T_{26}, P_7, P_{16}, P_{25}, $(H_2S)_5$, $(H_2S)_{16}$, $(H_2S)_{25}$, $(CO_2)_5$, $(CO_2)_{16}$, $(CO_2)_{25}$	14
CO_2 Absorber	T_2, T_8, T_{14}, P_3, P_9, P_{15}, $(H_2S)_2$, $(H_2S)_8$, $(H_2S)_{14}$, $(CO_2)_2$, $(CO_2)_8$, $(CO_2)_{14}$	12
H_2S Concentrator	T_3, T_5, P_4, $(H_2S)_1$, $(H_2S)_5$, $(CO_2)_1$, $(CO_2)_5$	7
Acid Gas K.O.	T, P	2
Selexol Stripper	T_1, T_3, T_7, T_{10}, P_1, P_3, P_9, $(H_2S)_3$, $(H_2S)_9$, $(CO_2)_3$, $(CO_2)_9$	11
	Total	46

Table 4.5: Candidate Sensor Locations in the Process Streams in the AGR Unit

	Streams				
Sensors	**T**	**P**	**F**	**H_2S analyzer**	**CO_2 analyzer**
No.	50	36	18	3	16
Total	123				

As mentioned previously, P-only control has been considered in this work. It is required to obtain the tuning parameters of estimator-based controllers so that the closed-loop system in MATLAB® remains similar to the APD® model. The tuning parameter, in this case the proportional gain, is determined by using both Ziegler-Nichols (1942) and Cohen-Coon (1953) approximate model tuning rules. It should be noted that for the APD® model, PID tuning parameters were obtained by Ziegler-Nichols and Cohen-Coon rules. In this case, the Ziegler-Nichols tuning rule outperforms the Cohen-Coon rule as latter one is more aggressive. The Ziegler-Nichols tuning rule is similar to the APD® model. Table 4.6 shows the rules that have been used to calculate the gain for the proportional controller.

Table 4.6: Approximate Model Tuning Rules (Ogunnaike, 1994)

Tuning Rules	**Controller Type**	K_c
Ziegler-Nichols	Proportional controller	$\frac{1}{K_p}\left(\frac{\tau}{\alpha}\right)$
Cohen-Coon	Proportional controller	$\frac{1}{K_p}\left(\frac{\tau}{\alpha}\right)\left[1+\frac{1}{3}\left(\frac{\alpha}{\tau}\right)\right]$

Process gain (K_p), time constant (τ) and time delay (α) are obtained from the nonlinear process model in APD®. Table 4.7 presents the list of controllers and the corresponding tuning parameters obtained from the aforementioned tuning rules.

Table 4.7: Tuning Parameters

Primary Controlled Variables	K	τ (min)	α (min)	K_c (Ziegler-Nichols)	K_c (Cohen-Coon)
CO_2 Capture	17.48	10.94	3.41	0.18	0.20
Water Content of Solvent at Stripper Bottom	0.127	66.14	0.457	1132.77	1135.37
Stripper Presser	0.102	0.355	0.06	57.86	61.12
Stripper Top Temperature	241.02	0.095	0.06	0.0066	0.0079
Semi lean Solvent Cooler Outlet Temperature	0.556	0.0862	0.06	2.58	3.18
Loaded Solvent Cooler Outlet Temperature	1.72	0.0884	0.06	0.854	1.05
Lean Solvent Cooler Outlet Temperature	11.36	0.0765	0.06	0.112	0.14
H_2 Cooler Outlet Temperature	72.59	0.108	0.06	0.0248	0.0294
Pressure of the H_2 Recovery Unit	0.161	5.87	0.432	83.97	86.03
Pressure of the HP Flash Vessel	0.0173	3.81	0.543	403.37	422.55
Pressure of the MP Flash Vessel	0.0194	3.41	0.496	353.76	370.89
N_2 Flowrate to H_2S Concentrator: F_{N_2}	115.85	0.237	0.06	0.0342	0.037

As mentioned earlier, four types of sensors were considered: flow, pressure, temperature, and composition sensors. Flow sensors are further classified based on the phase of the stream and range of the flowrate. Table 4.8 shows the types of sensors, range, % inaccuracy, typical cost range, and the cost used in the work. The data provided in this table have been obtained from Liptak and Liptak (2003). The cost of the sensors includes the price for measuring device, transmitter, other accessories as well as installation cost.

Table 4.8: Cost of Sensors (Liptak, 2003)

Types		Range	Inaccuracy	$ Cost	Considered
Flow sensor (plate + flanges + flanged meter + transmitter)	Gas phase	5–20 cm pipe	± 0.25 *to* ± 0.5 % of actual flow	$1500-3500	$3400
	Gas phase	20-50 cm pipe		$3500-8000	$7000
	Liquid phase	1-35 cm pipe		$1000-6000	$5300
	Liquid phase	70-100 cm pipe		$10,000-15,000	$14,000
Pressure measurement device (integral with a transmitter)		0 to 69 bars	0.1 to 1% of span	$1500-3700	$2500
Temperature (thermocouple integral with a transmitter)		-174.4 to $2337^{o}C$	± 1 to $2.8^{o}C$	$700 - 2000	$1000
H_2S analyzer (includes installation cost)		0 to 500ppm	1% of full scale	$65,000-145,000	$70,000
CO_2 analyzer (explosion-proof NDIR analyzer with recorder, includes installation cost)		0 to 50ppm	1% to 2% of full scale	$10,000	$10,000

Due to the large size of the SSND problem, considerable computational time is required to solve the constrained optimization problem on a single computer processor. For reducing the computation time, parallel computing was performed using the Distributed Computing Server (DCS®) and the Parallel Computing® toolbox from Mathworks®. The proposed algorithms were

implemented in a MATLAB® program running on a computer cluster with 32 Intel® Xeon® 2.10 GHz processors with 64 GB RAM.

The methods for the GA operators and the values for the parameters are intuitively chosen in accordance with the scale of the problem using the guidelines provided in the literature (Haupt and Haupt, 2004). Table 4.9 shows the key parameters in the GA specification.

Table 4.9: Set-up Parameters in GA

Parameters	
Generations	250
Selection	Stochastic uniform selection method
Crossover	Scattered crossover method
Population size	75
Mutation rate	0.01

4.5 Results

The efficiency of the AGR unit, defined as the amount of CO_2 capture per unit power consumption, is considered as the objective function in the SND algorithm. The maximum efficiency of the AGR unit as calculated from the dynamic model in APD® is 766.18 mol CO_2 capture/MWh when the plant runs under optimal operating conditions with no estimator and measurement errors. However, the value of the objective function without any measurements (i.e., estimator only) is the minimum value of the efficiency and is found to be 715.65 mol CO_2 capture/MWh. It is noted that the difference between the maximum and the minimum efficiency defined this way is a measure of the goodness of the process model.

Table 4.10 presents the results of eleven case studies for different budgets ($) for the sensor network. These case studies show that as the budget increases the number of sensors increases. Consequently, the number of available measurements also increases and/or costlier sensors are selected, which in turn can provide higher estimation accuracy for the process variables of interest as well as increased efficiency of the plant. Most of the CPU time is consumed for

solving the matrix Riccati equation involving process covariance matrix of dimension 1505×1505. In addition, as this problem considers 169 potential sensor locations, a large combinatorial problem is solved for each budget constraint. It is also observed from the case studies that the computational time depends on the initial population used by the GA. For the Cases 1-6 in Table 4.10, the computational time increases as the budget decreases. This is because of the higher number of sensors that can be considered without violating the budget constraint. For the lower budget cases, i.e. Cases 7-11, the initial population is created using the solution set of sensors obtained from the higher budget case studies. The computation time is significantly less for the lower budgets case studies. Using the DCS® and Parallel Computing® toolbox reduces the computation time by a factor of 6 compared to the non-parallel case studies.

Table 4.10: Number of Sensors and the Value of the Objective Function for Different Budgets

Cases	Budget (Cost of Sensors, $)	Number of Sensors	Efficiency ($molCO_2/MWh$)
1	431,900	75	766.0058
2	322,600	66	766.0058
3	229,400	64	766.0058
4	187,900	62	766.0058
5	149,000	56	765.11
6	118,700	46	762.6422
7	71,200	25	758.2536
8	63,700	25	756.7611
9	60,200	23	752.0415
10	59,700	24	750.0721
11	42,500	17	742.8588

Figure 4.6 shows how the optimal objective function value changes with the change in the budget. The figure shows that beyond 149K budget (Case 5), the value of the objective function changes negligibly.

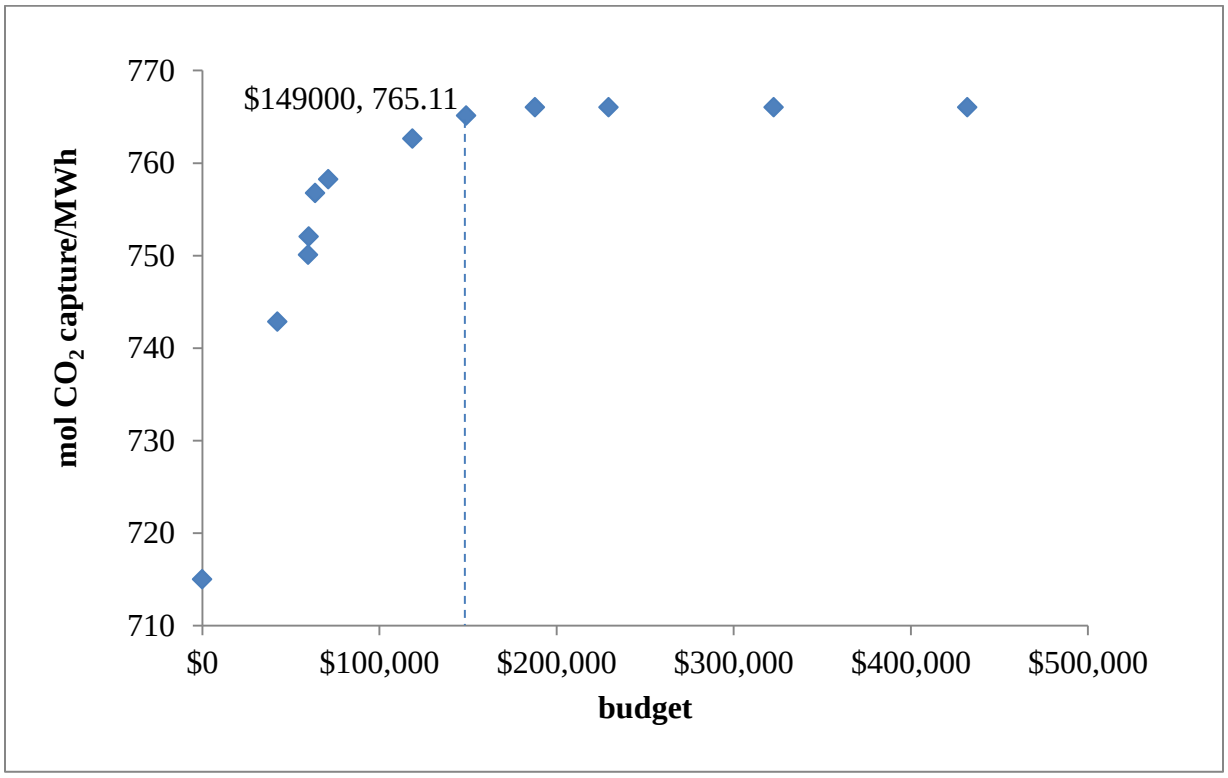

Figure 4.6: Objective function vs. cost of sensors.

Figures 4.7-4.9 illustrate the underlying reason for the increase in efficiency as the budget for sensors increases. In short, the impact of the budget on the estimation accuracy of a few key input-output variables is presented. As the budget increases, the estimation accuracy of the controlled variables improves. As a result, the values of the manipulated variables approach the values that were obtained for the maximum efficiency case. The vertical axes denote deviation variables and are calculated with respect to the values that were obtained for the maximum efficiency case.

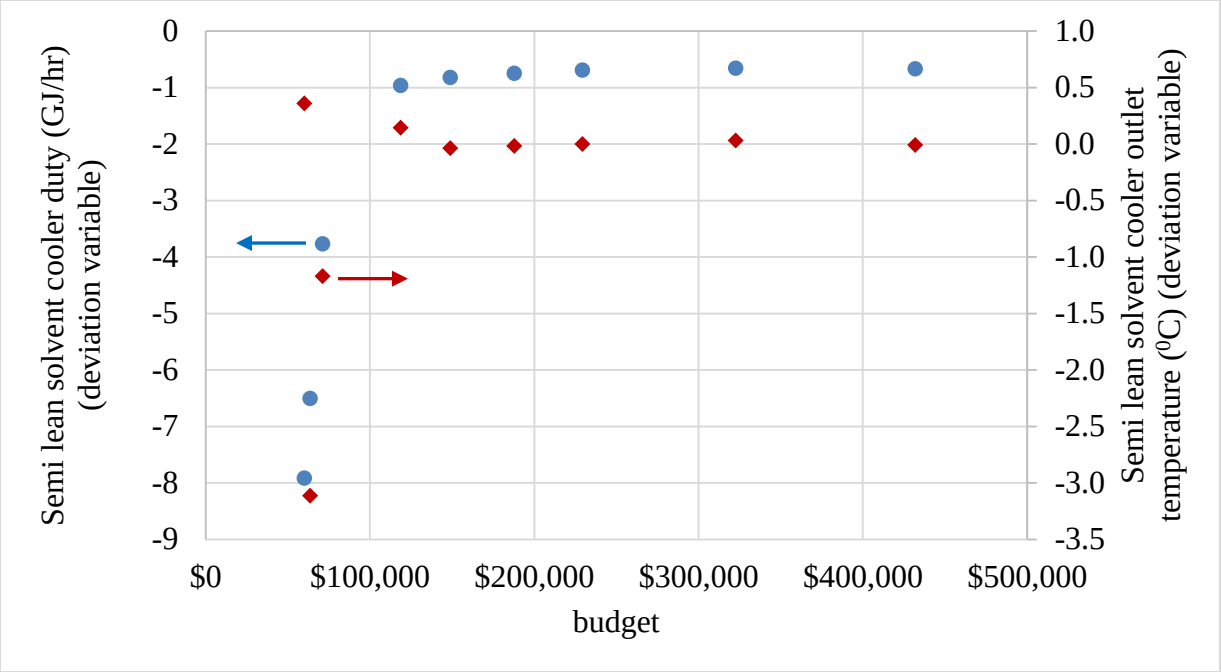

Figure 4.7: Manipulated variable: semi-lean solvent cooler duty (left); Controlled variable: semi lean solvent cooler outlet temperature (right) against budget.

Figure 4.7 shows how the semi-lean solvent cooler duty changes with the increase in sensor cost. In the semi-lean solvent cooler, the NH_3-refrigerant is used for chilling the solvent. As shown on the right-hand side axis of Figure 4.7, for lower budgets, the estimate of the temperature at the outlet of the refrigeration cooler deviates more on the negative side, i.e., it leads to a cooler temperature which is suboptimal. As a result, higher refrigeration duty is required at lower budget leading to loss in efficiency. Regardless of the noise in the measurements obtained from sensors, the estimated value of the controlled variable approaches to the optimal value as the sensors budget increases beyond $149K as shown in Figure 4.7.

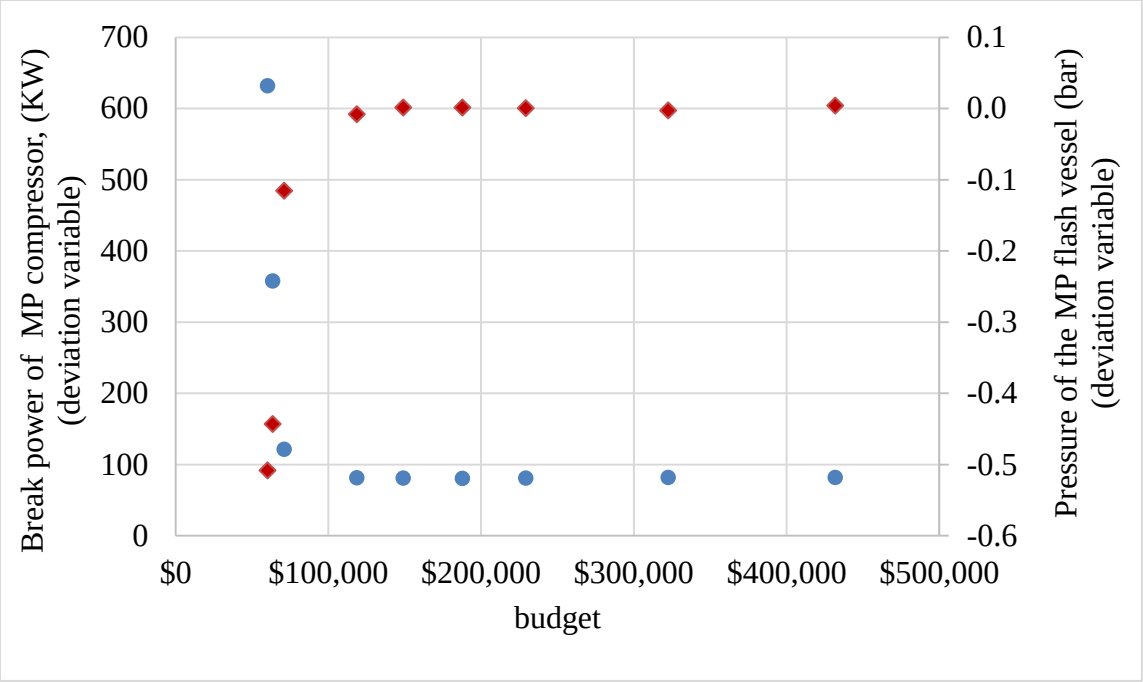

Figure 4.8: Manipulated variable: power of the MP compressor (left); Controlled variable: pressure of the MP flash vessel (right) against budget.

Figures 4.8-4.9 show how the compressor brake power changes as budget changes. Figure 4.8 and Figure 4.9 show the brake power of the MP and HP CO_2 compressors, respectively. In both the cases, at lower budget, the estimated flash vessel pressures are lower than the optimal value. As a result, the compressors consume more power than the optimal case leading to decrease in the efficiency. As before, even in the presence of process and measurement noise, the estimated value of pressure approaches the optimal value as the sensors budget increases beyond $149K (labeled in Figures 4.8-4.9).

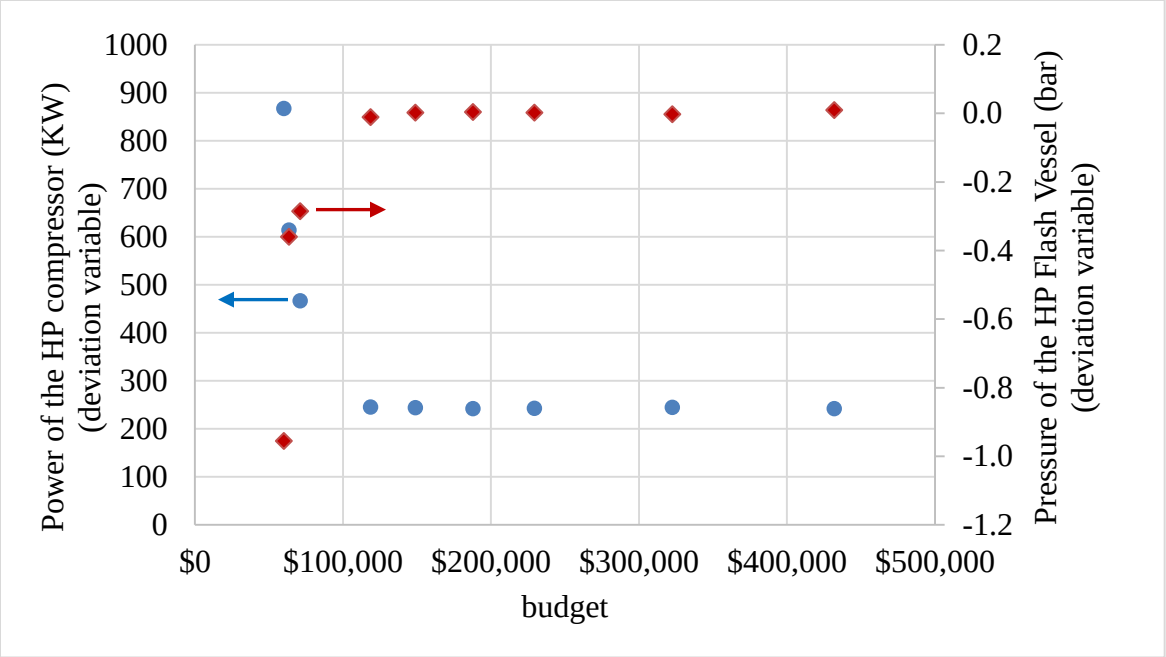

Figure 4.9: Manipulated variable: power of the HP compressor (left); Controlled variable: pressure of the HP flash vessel (right) against budget.

Table 4.11 shows the optimal set of sensors for $149K. It should be noted that if the number of controlled variables and/or variables for monitoring purposes are changed, the optimal budget is expected to change.

Table 4.11: Optimal Set of Sensors

Temperature Sensor	Pressure Measuring Device
1. [a]H_2S Absorber$_2$	30. [a]H_2S Absorber$_{16}$
2. [a]H_2S Concentrator$_5$	31. [a]CO_2 Absorber$_9$
3. Acid Gas K.O. Drum Vapor Outlet	32. Syngas Cooler Inlet
4. [a]CO_2 Absorber$_8$	33. Off Gas from Top of H_2S Absorber
5. [a]CO_2 Absorber$_{14}$	34. Clean Syngas at the Top of CO_2 Absorber
6. [a]Selexol Stripper$_1$	35. Semi-lean Solvent Cooler Inlet
7. [a]Selexol Stripper$_3$	36. Rich Solvent Heater Inlet
8. [a]Selexol Stripper$_7$	37. Rich Solvent at Selexol Stripper Inlet
9. Inlet to H_2O K.O. Drum	38. Lean Solvent at the inlet to CO_2 Absorber
10. Off Gas from Top of H_2S Absorber	39. Inlet to H_2 Recovery Flash Vessel
11. Off Gas Cooler Outlet Temperature	40. H_2 Recovery Flash Vessel
12. Off Gas at the Inlet to CO_2 Absorber	41. Stripped Gas Compressor Outlet
13. Clean Syngas at the Top of CO_2 Absorber	42. Acid Gas K.O. Drum Liquid Outlet
14. Rich Solvent at H_2S Absorber Bottom	43. H_2 Recovery Compressor Outlet
15. Rich Solvent Heater Inlet	44. HP Flash Vessel Outlet
16. H_2 Recovery Flash Vessel Outlet	45. Outlet of 1st LP CO_2 Compressor
17. HP Flash Vessel Outlet	46. Outlet of 2nd MP CO_2 Compressor
18. MP Flash Vessel Outlet	47. Glycol Absorber Top Outlet
19. Semi-lean Solvent Cooler Inlet	**Flow Sensor**
20. H_2S Concentrator Vapor Outlet	48. Semi-lean Solvent to H_2S absorber
21. Stripped Gas Compressor Outlet	49. H_2S Concentrator Vapor Outlet
22. 1st LP CO_2 Compressor Outlet	50. H_2O K.O. Drum Bottom Outlet
23. 2nd LP CO_2 Compressor Outlet	**CO_2 Analyzer**
24. 5th LP CO_2 Compressor Outlet	51. [a]Liquid Phase in H_2S Absorber$_{16}$
25. 1st HP CO_2 Compressor Outlet	52. [a]Liquid Phase in Selexol Stripper$_{16}$
26. 1st MP CO_2 Compressor Inlet	53. H_2S Absorber Bottom
27. Vapor of CO_2 Flash Vessel	54. LP Flash Vessel Bottom
28. Liquid of CO_2 Flash Vessel	55. Acid Gas K.O. Drum Liquid Outlet
29. Tail Gas to H_2S Absorber	56. MP Flash Vessel Vapor

a. Subscript at the end of location denotes stage number

4.6 Conclusions

A SSND algorithm is developed in this work for maximizing plant efficiency using an estimator-based control system while estimating other variables of interest for a given sensor network budget. We have considered two solution approaches for the SSND problem. A concise description is presented for the simultaneous solution approach that can be used for small-scale processes with a few unit operations. A sequential approach is developed for solving the SSND

problem for large-scale highly-integrated plants. In this approach, the GA is used to solve the IP problem while the NLP problem is solved by using a tear-stream approach. The direct substitution method is used to solve the 'tear stream' in the estimator-based control system. The SSND algorithm is then implemented for a highly-integrated AGR unit as part of an IGCC power plant with pre-combustion CO_2 capture. For solving this large-scale problem, a MATLAB® cluster is used for parallel computation leading to significant reduction in computation time. The results show that as the budget for sensors increases, the number of sensors used and the plant efficiency achieved both increase until a threshold is reached beyond which the budget has minimal impact on plant efficiency. The study also shows that an estimation error in the primary controlled variables, when selected from an economic perspective, can lead to loss in efficiency. However, a further decrease in estimation error below a certain threshold is wasteful since the sensor network budget increases while having minimal impact on the plant efficiency. This SSND algorithm is currently developed for grassroots plants, but can be readily enhanced for retrofitting. However, a SSND algorithm can result in a suboptimal transient efficiency profile. Therefore, in the next chapter we have developed a dynamic model-based SND (DMSND) algorithm for maximizing process efficiency.

Chapter 5

Dynamic Model-Based SND Algorithm

A dynamic model-based sensor network design (DMSND) algorithm has been developed for maximizing system efficiency for an estimator-based control system. The algorithm synthesizes the optimal sensor network in the face of disturbances or setpoint changes. Computational expense of the large-scale combinatorial optimization problem is significantly reduced by parallel computing and by using a combination of three novel strategies: multi-rate sampling frequency, model order reduction, and use of an incumbent solution that enables early termination of evaluation of infeasible sensor sets. The developed algorithm is applied to an AGR unit as part of an IGCC power plant with carbon capture. Even though there are more than one thousand process states and more than one hundred candidate sensor locations, the optimal sensor network design problem for maximizing process efficiency could be solved within couple of hours for a given budget.

5.1 Literature Review

DMSND algorithms are limited in the existing literature. Kadu et al. (2008) have considered a discrete linear time invariant system with multi-rate extension of the basic Kalman filtering algorithm to show the effect of various measurement sampling rates on state estimation. To find Pareto optimal solutions for the optimal sensor network, they solved dual objective functions including maximizing the quality of estimates and minimizing the measurement cost subject to a constraint on system detectability. Mellefont and Sargent (1978) developed an implicit enumeration algorithm using a linear stochastic system for selection of measurements to be used in optimal feedback control. This algorithm minimizes both the measurement cost and a quadratic function of the covariance of state prediction error with minimum number of measurements.

Computational expense is an issue for solving large-scale SND problems. Due to this difficulty, Chmielewski et al. (2002) have developed an alternative SND formulation to obtain the minimum cost sensor network. The authors improved computational efficiency by converting the nonlinear programming problem into a convex program through the use of linear matrix inequalities. They applied the SND approach to both steady-state and dynamic processes subject to single/multiple constraints on precision, gross-error detectability, resilience, and reliability. Nguyen and Bagajewicz (2008) have proposed a rigorous equation-based tree search method for designing nonlinear sensor networks but its performance is not satisfactory when dealing with large-scale problems ($\geq$ 35 measured variables and $\geq$ 25 balance equations). Later on, the same authors have proposed an approximate method (Nguyen and Bagajewicz, 2013) to solve a large-scale problem with 35 variables and 28 balance equations where the equation-based tree search method was used for initialization but still optimality of the solution is not guaranteed. Singh and Hahn (2006) presented an optimal SND approach where due to the computational expense; they have performed the calculation of empirical Gramians for all sensor locations outside the optimization loop.

DMSND algorithms are computationally very expensive due to the study of the transient behavior of the process for each candidate set of sensors. This computational expense significantly increases as the number of state variables and the number of candidate sensor variables increases. In particular, the solution of the matrix Riccati equation takes significant computational time. However, for the DMSND algorithm to be usable for large-scale industrial applications, an efficient algorithm is desired that can be solved within reasonable run times using standard computing resources. With this incentive, a computationally efficient DMSND algorithm for the estimator-based control system has been developed in this work for maximizing the efficiency of large-scale processes. In this DMSND algorithm, the Kalman filter (KF) is used for estimating process states (Paul et al., 2013) and particular focus is given to its convergence properties. In addition, several strategies have been developed for significantly reducing the computational expenses for solving large-scale DMSND problems.

Therefore, in this current work we have developed a computationally efficient DMSND algorithm for maximizing process efficiency.

5.2 Features of the DMSND Algorithm

Distinguishing features of the DMSND algorithm developed in this work over the SSND algorithm presented earlier (Paul et al., 2015) are given below:

- ***Feedback loop****:* The SSND algorithm considers the single steady-state operating point in the analysis and uses an infeasible path solution method involving trial and error to satisfy a tolerance limit on the tear stream for the estimator-based control system. In contrast, the DMSND algorithm resembles the real-world dynamic scenario expected for an estimator-based control system, i.e., it implements control actions by repeated feedback through the estimator until a new steady-state condition is reached.

- ***Transient response****:* The DMSND algorithm provides satisfactory estimation accuracy of all desired process variables during transients unlike the SSND algorithm that provides satisfactory estimation accuracy only under steady-state operation. It should be noted that satisfactory estimation accuracy under steady-state condition does not necessarily yield optimal performance during transient operation.

- ***Strategies for reducing computational expense***: Due to the presence of the feedback loop in the estimator-based control system and due to the large number of state variables and candidate sensors, the computational expense of solving the DMSND problem is significantly greater than for the SSND problem. Thus, strategies for reducing computational expense are the key to solving this large-scale DMSND problem successfully. To this end, three strategies have been developed and applied in this work. First, use of the reduced order model (ROM) in the DMSND algorithm significantly reduces the computational cost. Second, an approach, called multi-rate discretization (MRD) has been developed in this work. Furthermore, the best feasible solution known at any point during iteration (the incumbent solution) is used to terminate evaluation of infeasible sensor sets. These approaches will be expanded on later in this paper. To the best of our knowledge, none of these strategies have been considered in the open literature for DMSND algorithms. As will be seen later in this paper, the computational advantage obtained by using these strategies makes the solution of the large scale

combinatorial problem feasible with minimal loss of accuracy. In fact, it is observed that the DMSND algorithm is solved much faster than the SSND algorithm presented previously by the authors (Paul et al., 2015).

Multiple strategies have been developed for reducing computational expenses for DMSND algorithm. Subsequently, the discrete-time DMSND algorithm for efficiency maximization is developed by incorporating these strategies. Finally, the algorithm is applied to an AGR unit as part of an IGCC power plant.

5.3 Development of DMSND Algorithm

5.3.1 Continuous Estimator-Based Control System

The estimator-based control system as shown in Figure 4.1 is used to develop the DMSND algorithm. The algorithm considers the dynamic response of the estimator-based control loops subject to disturbances and/or setpoint changes. The set of equations governing each block of the estimator-based feedback loop is the same as Eq. (3.1), (3.3), (3.7)-(3.9) presented in Chapter 3. The error is calculated by Eq. (5.1) which is then used to calculate control action. DMSND algorithm assumes PI control law.

Comparator: Estimated measurements: $\hat{y}_{cont,est} = C_{cont}\hat{x}$ (5.1)

Error function: $\varepsilon(t) = y_{cont,set} - \hat{y}_{cont,est}$ (5.2)

Proportional-Integral (PI) Controller: $u = K_c\left[\varepsilon(t) + \frac{1}{\tau_I}\int_0^t \varepsilon(t)dt\right]$ (5.3)

It should be noted that the SSND algorithm in Paul et al. (2015) used a proportional-only controller.

5.3.2 Objective Function

Eq. (5.4) denotes the objective function where $\Delta\eta_{est,i}$ denotes the deviation from the optimal plant efficiency at any time instant i. k denotes the time instant that it takes for the process to reach its

new steady-state following the introduction of a disturbance (due to disturbance rejection by the controllers) or a change in the controller setpoint(s). The integer problem is solved subject to the constraints on budget and estimation accuracy as denoted by Eqs. (5.6) and (5.7), respectively, where c_i denotes the cost of a sensor of type *i*.

$$Min \sum_{i=1}^{k} \left(\Delta\eta_{est,i}\right)^2 \tag{5.4}$$

$$\sum_{\forall i} c_i\beta_i \ \leq \ b \quad ; \quad \beta_i = 0,1 \qquad \forall i \in N_s \tag{5.6}$$

$$\sum_{i=1}^{k} Esterr_i^{\,2} < tol_1 \tag{5.7}$$

The variables in Eq. (5.4) are a function of x_{act} in the estimator-based control system and are defined as:

$$\Delta\eta_{est} = \eta_{opt} - \eta(x_{act}, \beta) \tag{5.8}$$

$$Esterr = y_{ma,act} - \hat{y}_{ma} \tag{5.9}$$

As shown in Eq. (5.8), $\Delta\eta_{est}$ is the difference between the optimal efficiency (η_{opt}) and $\eta(x_{act},\beta)$, the efficiency obtained in the estimator-based system. Eq. (5.9) is used to calculate the estimation error in key variables, denoted by *Esterr*, by comparing the actual values with the estimated values of process variables.

5.4 Strategies for Reduction in Computational Expenses

As stated above, due to the feedback loop in the estimator-based control system, the computational expense is very high especially because the high-dimensional nonlinear matrix Riccati equation (Eq. 4.4) is solved for every discrete point in time for each candidate set of sensors until the process variables reach new steady state. Thus, for a process involving thousands of states, solving the DMSND problem is extremely time-consuming. To reduce the computational time, several strategies have been considered and are presented below.

5.4.1 Multi-Rate Discretization (MRD)

Multiple-measurement sampling times (Holsapple et al., 2007) can be considered for reducing computational expense. Intuitively the rich work in the area of numerical integration of stiff equations can be utilized for selecting the sampling times. However, there are number of complications for selecting sampling times while solving the SND problem in a process with an estimator-based control system. First and foremost, this is a differential algebraic equation (DAE) system with integer variables. Thus many of the tools available in the existing literature cannot be readily applied. Furthermore, the discrete-time system has one-epoch latency between the process and the estimator-based control system. Second, as the sampling intervals of the estimator, process, and the controllers are expected to be the same for the current work, stability of the closed-loop system including the estimator-based control system should be accounted for while selecting the sampling time. Because of the integer variables, i.e. selection of different sensors at different iterations, the desired profile of sampling intervals needs to be determined for each combination of sensors. Another complication is that if the discrete-time model has been obtained by using numerical data collected at given intervals, it may be impractical to reidentify the model for every combination of sensors (trillions of them) as the sampling interval changes. Even for a linear model, the calculation of the matrix exponential required for discretization as the interval changes can be computationally prohibitive. Thus the benefit of MRD on reduction of computational expense may not be realized if the sampling interval is adopted for every combination of sensors. Furthermore, as the process and measurement noises are considered to be Gaussian white noise with zero-mean, adequate sampling is needed to represent the distribution well during simulation. While all of the above methods can be very involved tasks and may not yielding much benefit on computational expense, the following heuristics worked quite well for the comprehensive example problem described in Section 5.7. The heuristic approach is based on the rate of change of the process variables in the face of disturbance or setpoint change. During fast transients, a higher sampling rate is needed thereby increasing computational expense. Once the process approaches the new steady-state condition, the rate of change in controlled variables decreases considerably and therefore, the sampling frequency are decreased accordingly.

5.4.2 Reduced Order Model (ROM)

Reduced order models that approximate a full-order model can considerably lower the computational cost. Depending on the methodology used for ROM development, the state variables in the ROM may not represent true states. Therefore, it should be ensured that the dynamic and steady-state responses of the ROM for the desired output variables are satisfactory in comparison to the full-order model. Several methods are available in the open literature and applications of these methods are problem specific. Two widely used methods for order reduction of very large-scale linear dynamic systems are the balanced truncation method and the Hankel norm approximation method. Both methods are based on Hankel singular value (HSV) decomposition. Interested readers are referred to the rich literature in this area (Glover, 1984; Safonov et al., 1990; Antoulas, 1999; Antoulas and Sorenseen, 2001; Willcoxand Peraire, 2002; Meyer, 1990). Both methods guarantee two of the most important ROM properties: (*i*) preserving stability of the original system and (*ii*) satisfying the global error bound (Eq. 5.10).

$$\|S - S_r\|_\propto \leq 2(\sum_{i=r+1}^{n} \sigma_i) \tag{5.10}$$

where $S_r(A_r,B_r,C_r)$ denotes the ROM of the original system $S(A,B,C)$. σ_i denotes the HSVs of the full-order process model. r is the reduced order. The optimal order of the ROM is selected either directly or indirectly by choosing a cut-off value for σ_r to obtain the r^{th} ROM. One common approach is to look for large "gaps" in the relative magnitude of σ_{r-1}/σ_r. The cut-off value for σ_r should be selected such that the constraint(s) considered for model order reduction is satisfied. One common approach is to consider a constraint as shown in Eq. (5.11) that ensures that the squared difference between the full-order model and ROM outputs satisfies the tolerance.

$$\sum_{j=1}^{m} \sum_{i=0}^{p} (y(i,j) - y_r(i,j))^2 \leq tol \tag{5.11}$$

In Eq. (5.11), p denotes number of samples, m denotes the number of measurements, $y(i,j)$ is the i^{th} sample of the j^{th} measurement for the full-order system and $y_r(i,j)$ is the i^{th} sample of the j^{th} measurements for the ROM.

5.4.3 Termination Using Incumbent Solution

An incumbent solution is defined as the best value of the integral deviation from the optimal efficiency that has been obtained so far. For a given set of sensors, if the integral deviation from the optimal efficiency exceeds the incumbent solution, then the sensor set being evaluated is inferior to the sensor set corresponding to the incumbent solution and therefore, the evaluation of this sensor set is terminated. If the current result is superior to the existing incumbent solution, then the current results become the new incumbent solution. Let a system subjected to a disturbance/setpoint change take time t_{ss} to reach a new steady state with an incumbent solution equal to $\Delta\eta_{est,incum}$. If an inferior sensor set exceeds the incumbent solution at the termination time t_{term} and results in $\sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2$, then the optimization problem is reformulated as:

$$Min\left[\sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2+\lambda_1\sum_{i=1}^{k}Esterr^2+\lambda_2\left(\frac{\Delta\eta_{est,incum}}{t_{ss}}-\frac{\sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2}{t_{term}}\right)^2\right] \quad (5.12)$$

Note that the constraint Eq. (5.7) is eliminated due to this reformulation of the objective function. The third term in Eq. (5.12) $\frac{\Delta\eta_{est,incum}}{t_{ss}}-\frac{\sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2}{t_{term}}$ is the penalty term for exceeding the incumbent solution. Considering convergence issues and computational expense, the estimation error and the penalty term are included in the objective function. In Eq. (5.12), λ_1 and λ_2 are the weighting factors for estimation error and incumbent solution, respectively.

5.5 Discrete-Time Estimator-Based Control System

A discrete-time version of the estimator-based control system is developed and used in the DMSND algorithm.

$$\Phi=\exp(A\Delta t) \quad (5.13)$$

$$G=\int_0^{\Delta t}\exp(A\sigma)\,Bd\sigma \quad (5.14)$$

Φ and G are the discrete transition matrix and discrete input matrix respectively.

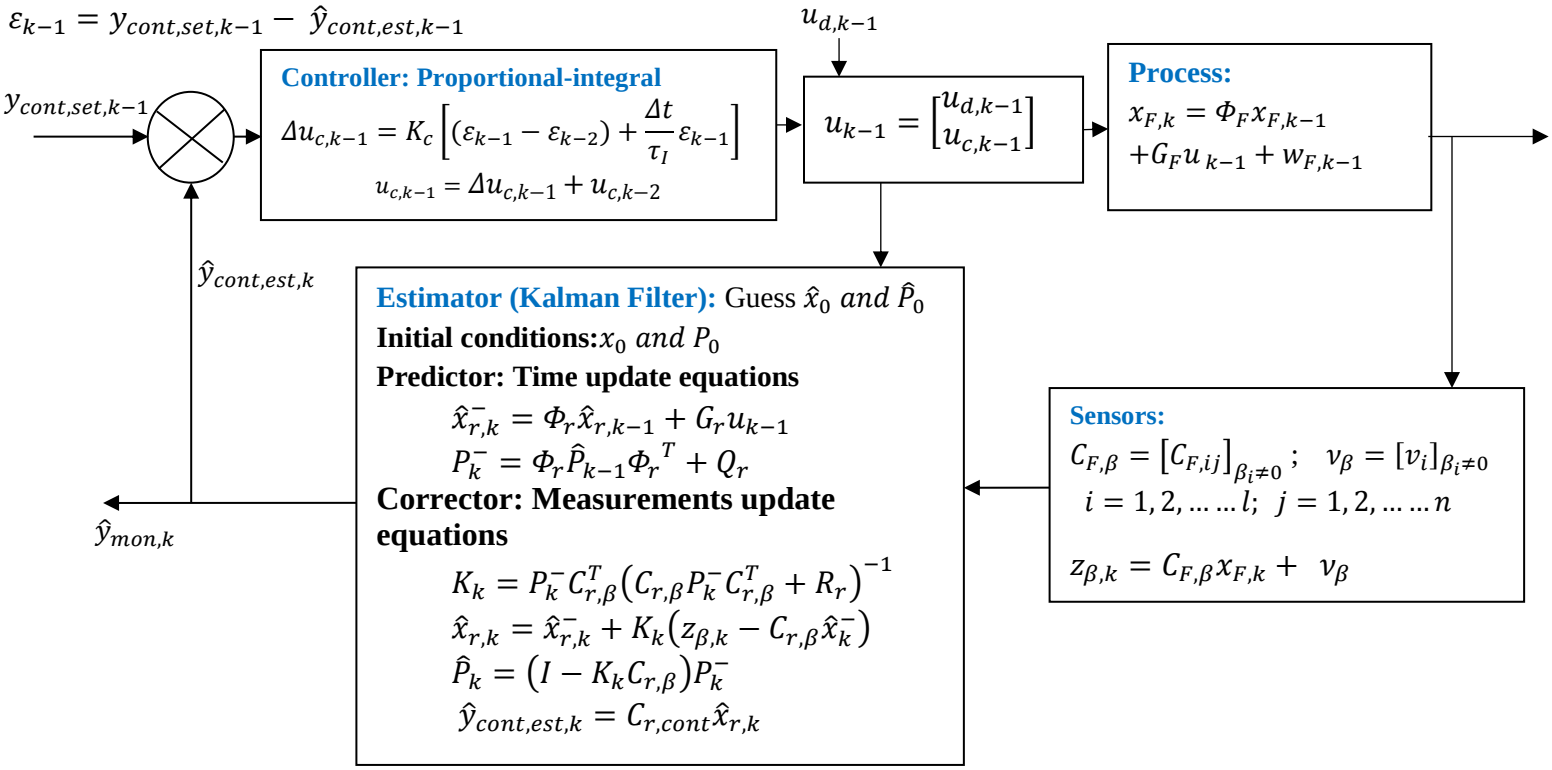

Figure 5.1: Reduced-order estimator-based control-loop configuration in conjunction with full-order process model.

The process block diagram is shown in Figure 5.1 and is characterized by full n^{th}-order linear model and the additive n^{th}-order process noise W_F. The estimator in this case considers reduced-order matrices: discrete process matrix Φ_r, input matrix G_r, and output matrix C_r. Usually, good guesses for the process and measurement noise variance-covariance matrices, $Q \in \Re^n$ and $R \in \Re^m$, respectively, are required for achieving the desired estimation accuracy for the variables of interest. The reduced-order matrices in the estimator block provide computational efficiency while solving matrix Riccati equation. In addition, a good guess is only needed for the reduced order process noise variance-covariance matrix Q_r (diagonal matrix of dimension r) instead of Q. Even though Q_r is a lower dimensional matrix in comparison to the full-order model, it is difficult to provide a good estimate as the states in the ROM do not represent true states. As exact knowledge of Q (and similarly Q_r) and R are very difficult, if not impossible (especially that of Q), and Q and R are expected to evolve in the real life as the plant keeps operating, the sensor network has to still perform satisfactorily as long as the errors are within certain bounds.

This is typically a consideration made after sensor placement and therefore not considered in the DMSND algorithm. The authors of this paper (Paul et al., 2013) have looked into possibility of adapting *Q* and *R* for a case similar to the example considered here where optimal performance of the filter was obtained even in the presence of inaccurate knowledge of *Q* and *R*. Interested readers are referred to our previous work for more information.

Finally, the DMSND problem results in following optimization problem:

$$Min\left[\sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2+\lambda_1\sum_{i=1}^{k}Esterr^2+\lambda_2\left(\frac{\Delta\eta_{est,incum}}{t_{ss}}-\frac{\sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2}{T_{term}}\right)^2\right]$$

s.t.

$$x_{F,k}=\Phi_F x_{F,k-1}+G_F u_{k-1}+w_{F,k-1} \tag{5.15}$$

$$C_{F,\beta}=\left[C_{F,ij}\right]_{\beta_i\neq 0}\;;\quad \nu_\beta=[v_i]_{\beta_i\neq 0}\;;\quad i=1,2,\ldots\ldots l;\;\; j=1,2,\ldots\ldots n$$

$$z_{\beta,k}=C_{F,\beta}x_{F,k}+\;\nu_\beta$$

$$\hat{x}_{r,k}^{-}=\Phi_r\hat{x}_{r,k-1}+G_r u_{k-1}$$

$$P_k^{-}=\Phi_r\hat{P}_{k-1}{\Phi_r}^T+Q_r$$

$$K_k=P_k^{-}C_{r,\beta}^T\left(C_{r,\beta}P_k^{-}C_{r,\beta}^T+R_r\right)^{-1}$$

$$\hat{x}_k=\hat{x}_k^{-}+K_k\left(z_{\beta,k}-C_r\hat{x}_k^{-}\right)$$

$$\hat{P}_k=\left(I-K_kC_{r,\beta}\right)P_k^{-}$$

$$\hat{y}_{cont,k}=C_{r,\beta}\hat{x}_k$$

$$\varepsilon_k=y_{cont,set}-\hat{y}_{cont,est,k}$$

$$\Delta u_{c,k}=K_c\left[(\varepsilon_k-\varepsilon_{k-1})+\frac{\Delta t}{\tau_I}\varepsilon_k\right]$$

$$u_{c,k}=\Delta u_{c,k}+u_{c,k-1}$$

$$u_k=\begin{bmatrix}u_{d,k}\\u_{c,k}\end{bmatrix}$$

$$\textstyle\sum_{\forall i}c_i\beta_i\;\leq\;b\quad;\quad\beta_i=0,1\qquad\forall i\in N_s$$

5.6 DMSND Algorithm

The DMSND algorithm shown in Figure 5.2 includes all of the strategies mentioned in the previous section for reducing computational expense. The optimization problem is solved using the GA. In the first generation of GA, the algorithm creates an initial population consisting of candidate sets (β) of sensors. For every candidate set, the initial control action $u_{c,k-1}$(deviation variable) is zero. As the disturbances perturb the process or the controller setpoint(s) is(are) changed, controlled variables deviate from the setpoint. Estimated controlled variables are obtained from the estimator given the available measurements (selected by GA) and discrete process model. The estimator, controller equations, and process model are solved sequentially at every discrete point in time in the face of disturbance/setpoint change. Control action continues until the new steady-state condition is reached. Over the period of process response, the efficiency profile is calculated corresponding to a sensor set that is then scored by a fitness function (objective function). Some sets of sensors with higher fitness values are classified as elite and the GA proceeds to the next generation. The remaining sets of sensors in the next generation are obtained by selection, cross-over and mutation based on their fitness. The set of candidate sensors evolves over the successive generations and the algorithm is halted once it satisfies the termination criteria. The final set of sensors obtained from this algorithm yields optimal efficiency given the budget constraints and constraints on estimation accuracy. It should be noted that the lower budget case studies comply with the budget constraints but fail to satisfy the estimation accuracy.

The incorporation of the incumbent solution concept in the DMSND algorithm brings additional advantages in optimization. At the end of every time instant, the cumulative deviation of the efficiency from the optimal value is calculated and compared with the incumbent. If for a candidate set, the cumulative deviation exceeds the incumbent, $\Delta\eta_{est,incum}$, further computation is terminated for that set. Otherwise the sequential control action proceeds in a regular fashion and ends up at the new fitness value once it reaches the new steady-state time t_{term}. The new fitness value is then used as the new incumbent for the subsequent evaluation of the candidate set of sensors.

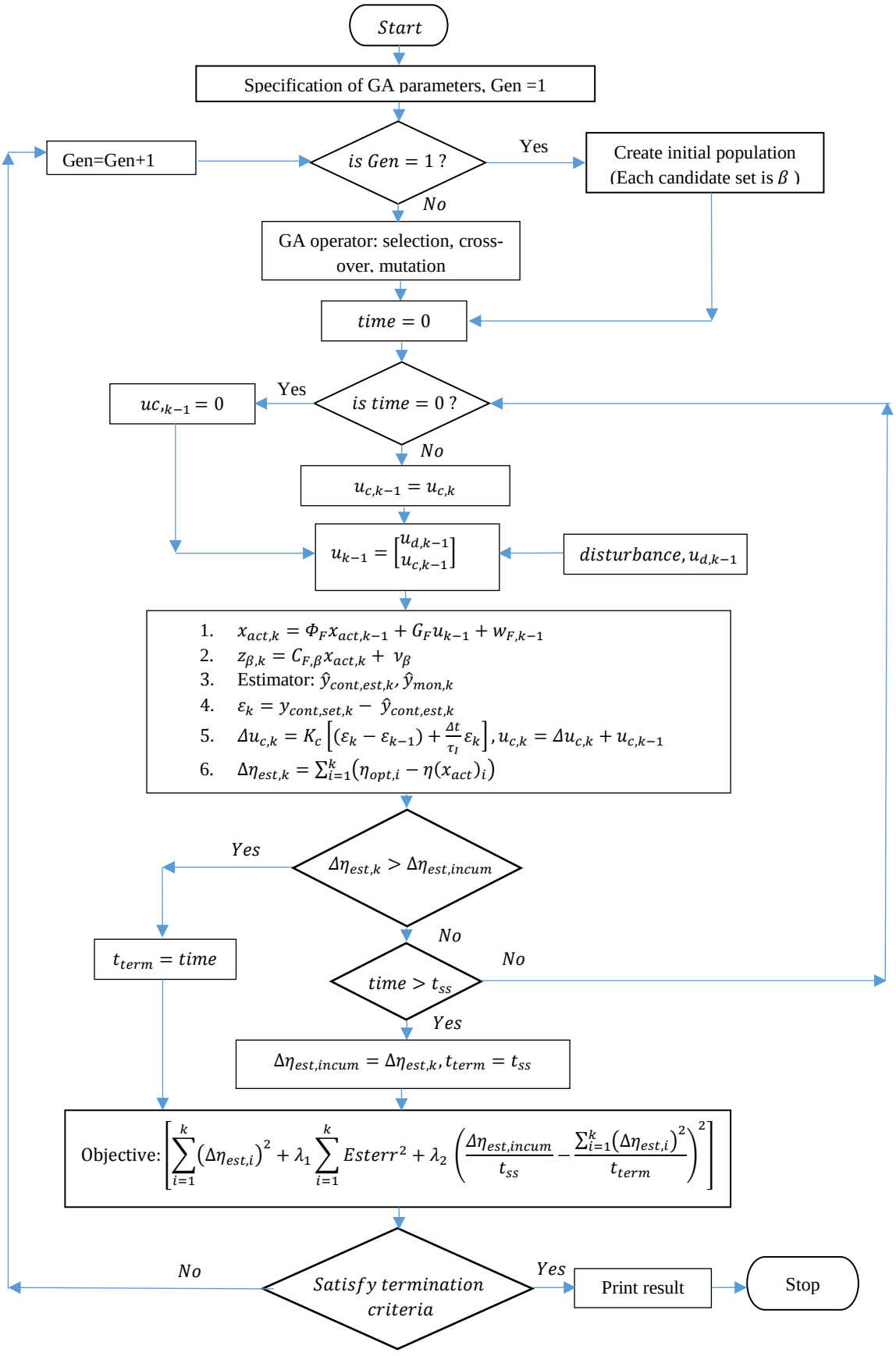

Figure 5.2: Flowchart of the DMSND algorithm incorporating termination using incumbent solution.

The designed algorithm results in an MINLP problem. The optimization problem has been decomposed such that the integer problem is solved at upper level by using GA (Haupt and Haupt, 2004).

5.7 Case Study

This section illustrates the application of the DMSND algorithm on AGR unit in Figure 5.3. A short description of AGR process is given in section 3.4 of Chapter 3. Details of the modeling of this unit can be found in Bhattacharyya et al. (2011).

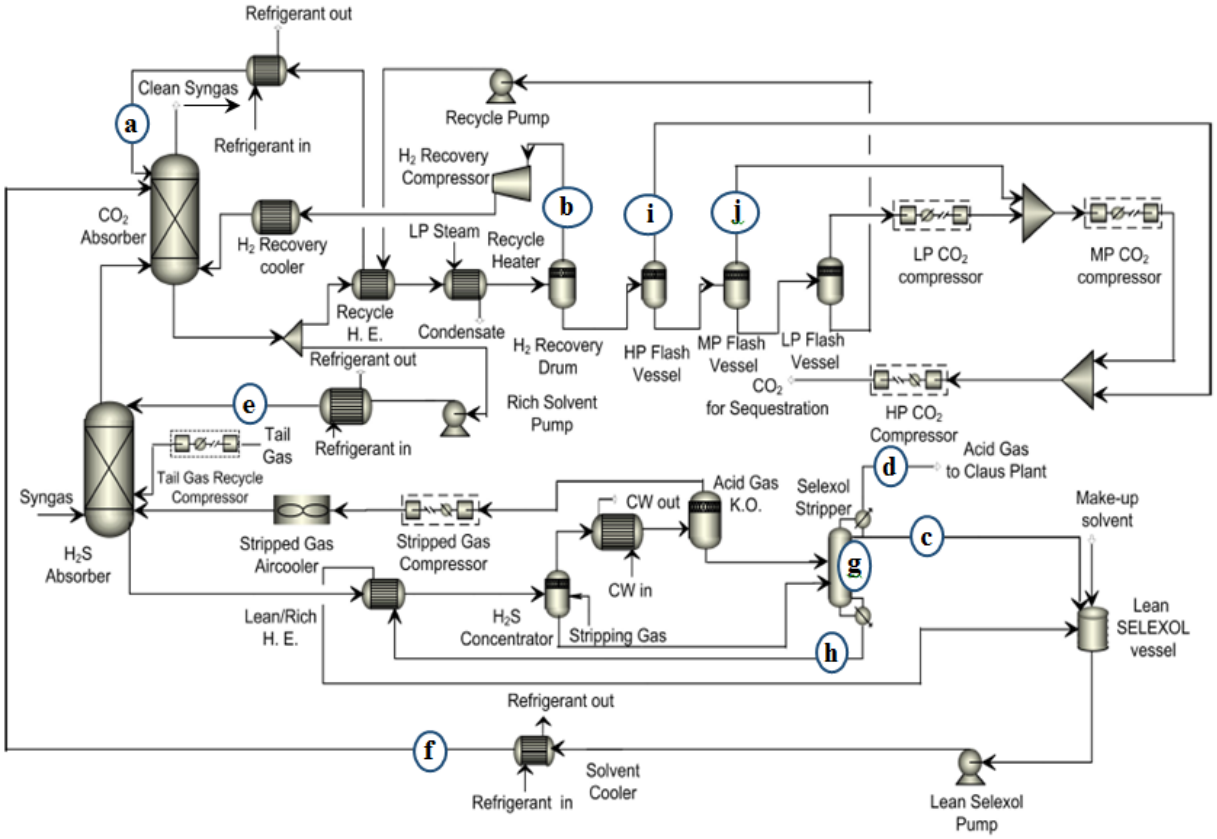

Figure 5.3: Locations of primary control variables (labeled from a-j) in AGR unit (Bhattacharyya et al., 2011) considered for implementation of DMSND algorithm.

The DMSND algorithm involves using three software platforms: Aspen Plus®, Aspen Plus Dynamics® (APD), and MATLAB®. The steady-state AGR process model is designed in Aspen Plus and then exported to APD for control system design and dynamic simulation. The nonlinear model in APD is linearized around the nominal operating condition by using a featured built-in tool called control design interface (CDI). It should be noted that all the primary controlled variable loops are kept open during the linearization while the loops are closed in MATLAB during the SP algorithm implementation. Thus the model obtained from APD is that of the open-loop system. The CDI script calls the controlled variables as output and control variables as input. APD generates the linear state-space model of the AGR unit, which along with the algebraic measurement equations for candidate sensor locations, the primary controlled variables, and variables that appear in the objective functions, are then exported to MATLAB®. Implementation of the DMSND algorithm for optimization is performed in MATLAB®.

Even though all primary controlled variables have to be estimated by the measurement framework, sensors are not necessarily placed on all primary controlled variables. This is because measurement of some of these variables can be difficult and/or expensive and can have time delay, high noise, and/or low estimation accuracy. On the other hand, it may be possible to estimate these variables satisfactorily by placing sensors elsewhere in the process and within the budget constraint.

The AGR process model has 1505 state variables. Variations in the syngas flowrate and CO_2 composition in the syngas to the unit are considered as disturbances. The flowrate disturbance is simulated by changing the inlet pressure of the syngas to the AGR unit. It should be noted that even though the study considered two disturbances, the SP algorithm is generic and additional disturbances and change in the controller setpoints can be readily implemented.

Figueroa et al. (2008) have investigated how CO_2 capture affects plant efficiency. Efficiency of the AGR unit is defined in Eq. (5.16) where the numerator represents the amount of CO_2 captured while the denominator is the MWh power consumption.

$$\eta(x_{act}, \beta) = \frac{F_{CO_2,in}(x_{act}) - F_{CO_2,out}(x_{act})}{aF_{solvent}(x_{act}) + \sum_{c=1}^{3} P_c(x_{act})} \tag{5.16}$$

Four different types of commonly used sensors have been considered: temperature (*T*), pressure (*P*), flow (*F*), and composition (z_{H_2S} and z_{CO_2}). The AGR process flowsheet is studied and 126 candidate sensors are identified for potential placement. Tables 5.1 and 5.2 show the distribution of these candidate sensors in the AGR unit.

In our previous paper (Paul et al., 2015), every sixth tray of the H_2S absorber was considered for placement of a temperature sensor while in this work we have considered every fourth tray due to a lower price of temperature sensors. For the H_2S absorber, this results in a total of 15 candidate sensor locations, one more than considered in Paul et al. (2015). For the other three separation columns, a broader distribution of trays has been considered for locating candidate sensors; however, the total number of candidate sensors for each remains unchanged. As shown in Table 5.1, a total of 47 candidate sensor locations are identified for the AGR equipment. For the AGR process streams, the measurements, which are not sensitive to disturbance changes or, dependent on input changes, are eliminated. As shown in Table 5.2, this results in a total of 79 candidate sensor locations in the AGR process streams in this work, compared to the 117 candidate stream locations in Paul et al. (2015). Eliminating these stream measurements, results in an additional decrease in computational expense.

Table 5.1: Candidate Sensor Locations in the Equipment Items in the AGR Unit

Equipment	Sensors	No.
H_2S Absorber	T_4, T_8, T_{12}, T_{16}, T_{20}, T_{24}, P_7, P_{16}, P_{25}, $(H_2S)_5$, $(H_2S)_{16}$, $(H_2S)_{25}$, $(CO_2)_5$, $(CO_2)_{16}$, $(CO_2)_{25}$	15
CO_2 Absorber	T_4, T_8, T_{12}, P_3, P_9, P_{15}, $(H_2S)_2$, $(H_2S)_8$, $(H_2S)_{14}$, $(CO_2)_2$, $(CO_2)_8$, $(CO_2)_{14}$	12
H_2S Concentrator	T_3, T_5, P_4, $(H_2S)_1$, $(H_2S)_5$, $(CO_2)_1$, $(CO_2)_5$	7
Acid Gas Knockout	T, P	2
Solvent Stripper	T_1, T_4, T_8, T_{10}, P_1, P_3, P_9, $(H_2S)_3$, $(H_2S)_9$, $(CO_2)_3$, $(CO_2)_9$	11
		Total = 47

Table 5.2: Candidate Sensor Locations in the Process Streams in the AGR Unit

Sensor Types	T	P	F	z_{H_2S}	z_{CO_2}
No. of Sensors	29	20	14	2	14
Total	79				

The range of inaccuracy of commercially available sensors can be found in Liptak (2003). Estimator-based control action is implemented for the primary controlled variables that have been identified by Jones et al. (2014) for this AGR unit. Table 5.3 shows the values of the tuning parameters for these controllers obtained using the Cohen-Coon method (Cohen and Coon, 1953; Ogunnaike and Ray, 1994). The tuning parameters remain unchanged for all iterations.

Table 5.3: Controller Tuning Parameters

Controller Label in Figure 5.3	**Controlled variable**	**Proportional gain (K_c)**	**Reset time (τ_I), min**
a	Loaded Solvent Cooler Outlet Temperature	0.199	4.21
b	Pressure of the H_2 Recovery Unit	5.598	1.37
c	Stripper Pressure	1.016	0.21
d	Stripper Top Temperature	0.001	0.02
e	Semi-lean Solvent Cooler Outlet Temperature	0.262	0.11
f	Lean Solvent Cooler Outlet Temperature	0.004	0.009
g	Stripper Temperature T_{11}*	4.15e-5	20
h	Water Concentration in Solvent at Stripper Bottom	9.285	30
i	Pressure of High-Pressure (HP) Flash Vessel	30.516	1
j	Pressure of Medium-Pressure (MP) Flash Vessel	17.754	1

* = Subscript denotes stage number

To reduce computation expense, the strategies mentioned above are implemented when applying the DMSND algorithm to the AGR unit. For determining the MRD strategy, the initial

measurement sampling time is kept small and chosen on an *ad hoc* basis. It is observed that the primary controlled variables currently under consideration require almost seven hours to reject the impact of the applied disturbance and more than 90% of the transient occurs within the first four hours. Therefore, for the first four hours, the sampling frequency is set to 200 times an hour (Δt_1 = 0.005hr) and then decreased to 50 times an hour (Δt_2 =0.02hr). By using the larger discretization time, the MRD strategy reduces the computational expense by about 75% after *time* =4 hr. This heuristic approach is found to be satisfactory for the current problem for all combinations of sensors studied in this work.

For ROM, the balanced model truncation using the square root method is considered. Figure 5.4 shows the semi-log plot of squared error, $\sum_{j=1}^{m}\sum_{i=0}^{p}(y(i,j)-y_r(i,j))^2$ vs. order of the model (k). The optimal ROM is found to be of order 82 with the tolerance set at 1×10^{-3} for Eq. (5.11).

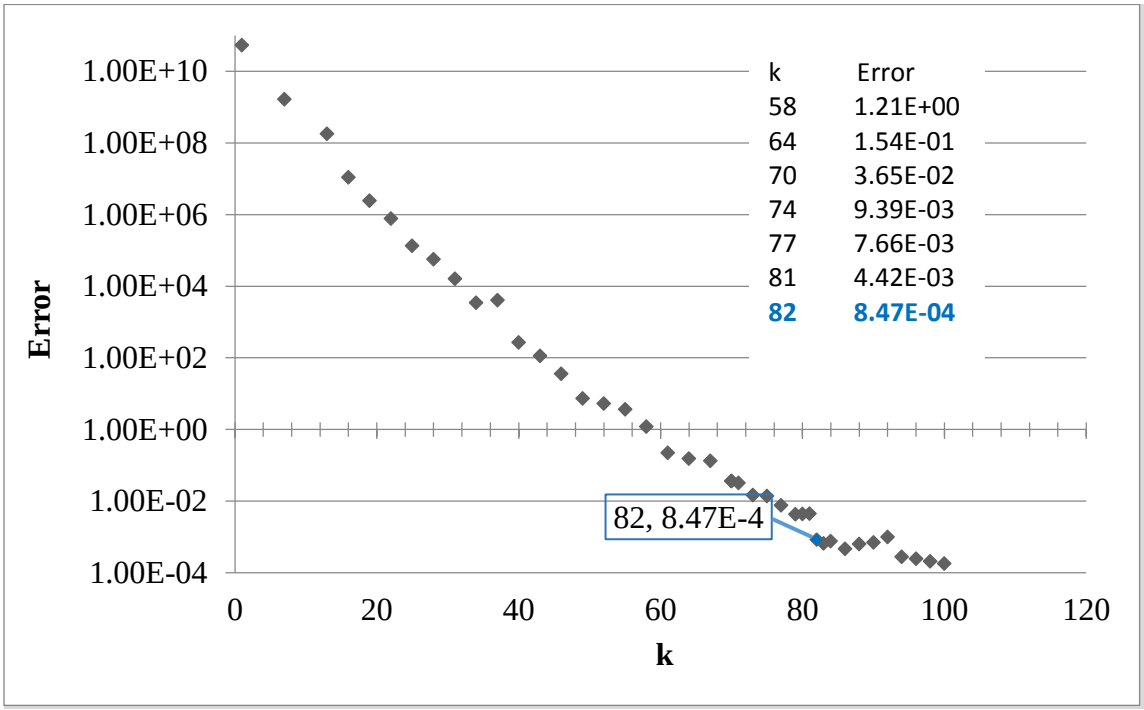

Figure 5.4: Approximation error as a function of reduced order (k).

In this AGR case study, the optimal efficiency profile in terms of moles of CO_2 captured per MWh of power consumed (Eq. 5.16) is obtained by using the nonlinear model of the optimally

designed AGR unit in the absence of any noise. The developed sensor placement algorithm is implemented by using the GA available in the global optimization toolbox in MATLAB. GA creates initial population of the candidate set of sensors and over the successive generation it reaches the optimal set of sensors.

In this work, the value of the weighting factors in Eq. (5.15) is chosen by trial and error by considering the desired weight for estimation accuracy versus incumbent solution. In this case study, following values are considered: $\lambda_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1.0 \times 10^{-3}. \end{bmatrix}$ and $\lambda_2 = 1.0 \times 10^{-3}$.

5.8 Results

The process efficiency, defined by Eq. (5.16), is calculated over the period of the process response. The integral squared error (ISE) due to the deviation from optimal efficiency is obtained. Pareto optimal solutions shown in Figure 5.5 are obtained by plotting $\log_{10}$(ISE) as a function of budget (cost of sensors). It shows that beyond a threshold budget (S_5), there is only a small decrease in ISE/budget. Thus the measurements corresponding to the threshold budget are considered to be the optimal set.

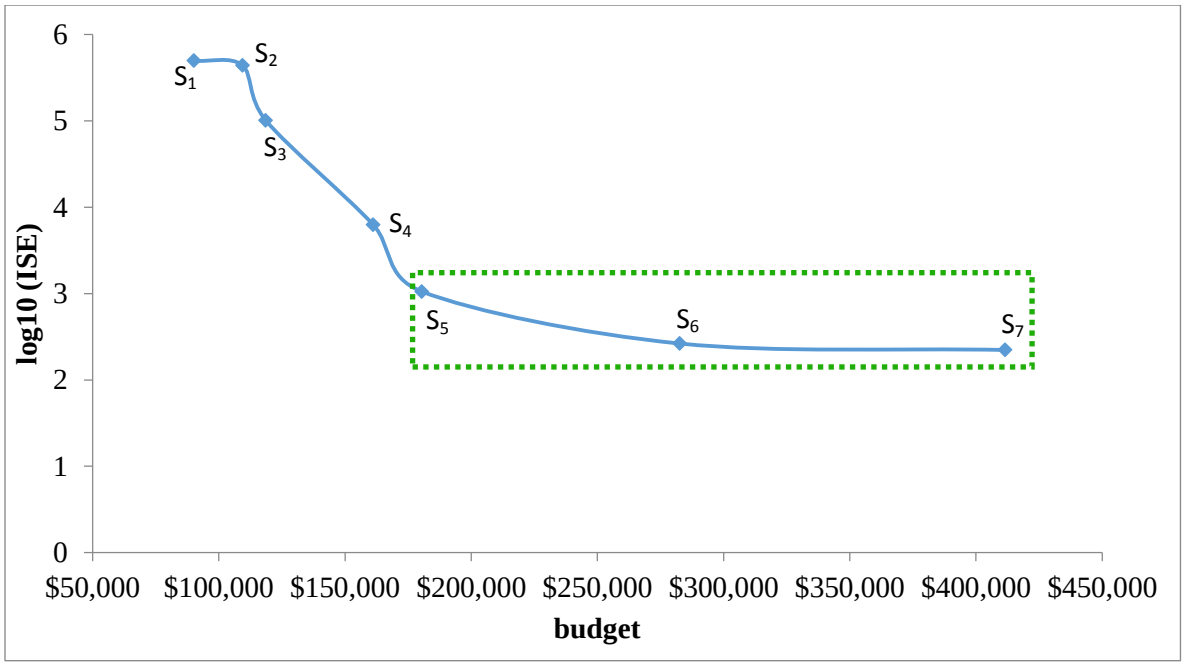

Figure 5.5: Pareto ISE plot for optimal sensor sets for different budgets as calculated by the DMSDN algorithm.

It should be noted that the results presented for each budget consider estimation accuracy on process operational constraints, the maximum allowable solvent temperature of 175°C, and minimum stripper pressure of 276 kPa. It is expected that the budget will increase substantially if grater estimation accuracy is desired for a larger number of variables.

Table 5.4 presents detail comparison of the results for different budgets, in terms of number of sensors, types, etc.

Table 5.4: Analysis of Sensor Sets Obtained from the DMSDN Algorithm for S_3- S_7 Budgets

Sensors	**Cost($)/ sensor**	**Available locations**	**Solution Sensor Sets**				
			S_3 $118,600	**S_4** $161,200	**S_5** $180,400	**S_6** $282,600	**S_7** $411,600
Temperature	1,000	45	24	22	27	24	26
Pressure	2,200	31	13	16	17	13	18
Flow	4,000	14	4	6	9	10	4
H_2S Analyzer	70,000	12	0	0	0	1	3
CO_2 Analyzer	10,000	24	5	8	8	12	12
Selected locations		**126**	**46**	**52**	**61**	**60**	**63**

Figure 5.5 shows that as the budget is increased, it leads to diminishing returns. At lower budgets, a smaller number of measurements are selected. Thus, estimation of controlled variables suffers and results in poor control performance which in turn affects the efficiency. With the increase in budget, process efficiency approaches the optimal performance profile. At higher budgets, solution sets S_5- S_7 show that locations and types decrease ISE only marginally and the total number of sensors is not very different but the locations are different. In addition, it is observed that the number of H_2S analyzers keeps increasing with higher budgets. As shown in Table 5.4, it can be noted that the H_2S analyzers are much more costly than any other type of sensors including CO_2 analyzers.

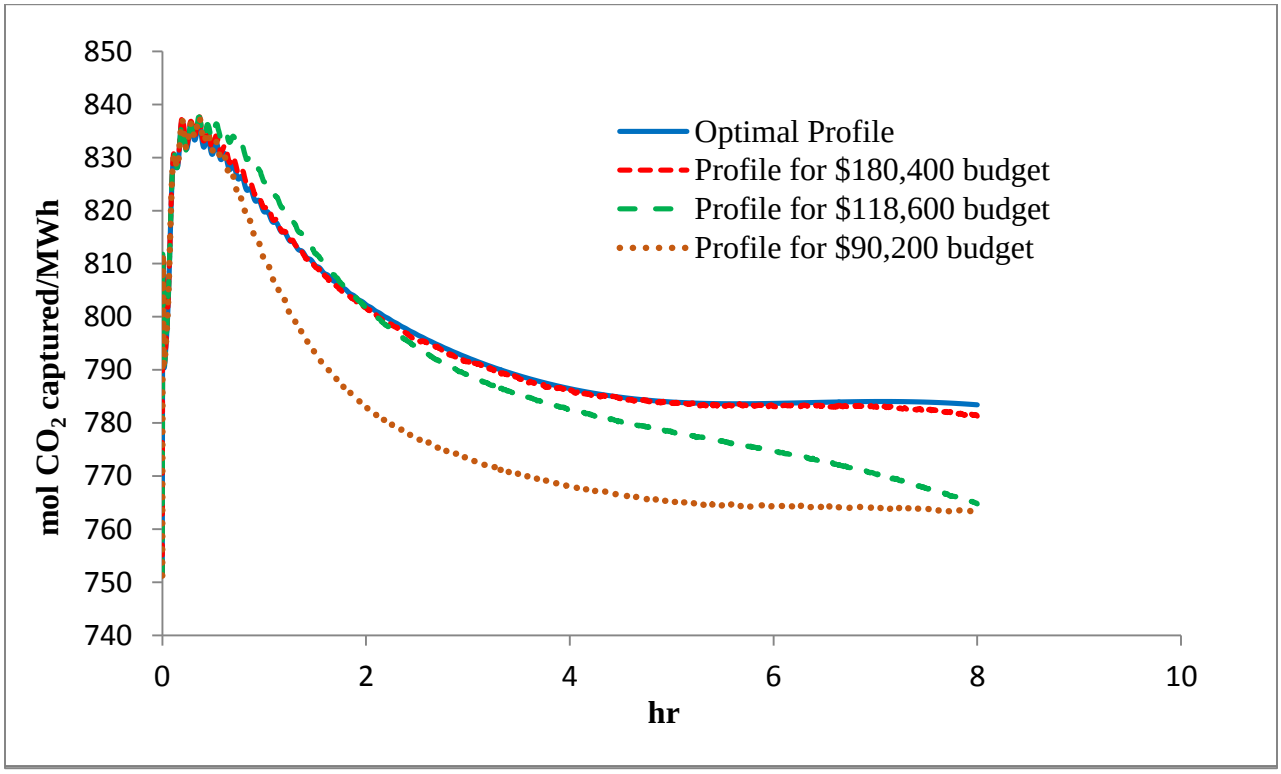

Figure 5.6: Comparison of DMSDN estimator-based efficiency for different budgets with the optimal efficiency.

Figure 5.6 shows the comparison of the efficiency profile for different budgets. The optimal efficiency profile is calculated in the absence of measurement noise and without opening the primary controlled variables. It is observed that the profile for $180,400 corresponding to S_5 in Figure 5.5 closely follows the optimal performance. The profiles for lower budget cases deviate substantially from the optimal profile. In Table 5.4, it is noted that the optimal set, S_5 does not include H_2S analyzer. This is mainly due to the higher cost of an H_2S sensor. It should be noted that one H_2S analyzer is equivalent to the cost of 70 temperature sensors. It is observed that a number of temperature and pressure sensors is selected by the DMSND algorithm at a lower cost than a single H_2S sensor still achieving satisfactory estimation accuracy.

The impact of different disturbances such as total pressure (D_P), flowrate of CO_2 (D_{CO_2}) and H_2S (D_{H_2S}) in the inlet syngas is presented in Table 5.5. In Table 5.5, the measure of deviation from the optimal profile is given in terms of integral absolute error (IAE).

Table 5.5: Impact of Single and Multiple Disturbances for Optimal Sensor Set S_5

Disturbances	**IAE** (**mol CO_2capture/MWh)hr**
D_P	3.17
D_P and D_{CO_2}	4.97
D_P, D_{CO_2} and D_{H_2S}	4.94

5.8.1 Transient Performance of the SSND vs. DMSND

Figure 5.7 shows the performance comparison of the sensor network obtained by SSND algorithm (Paul et al., 2015) and DMSND algorithm.

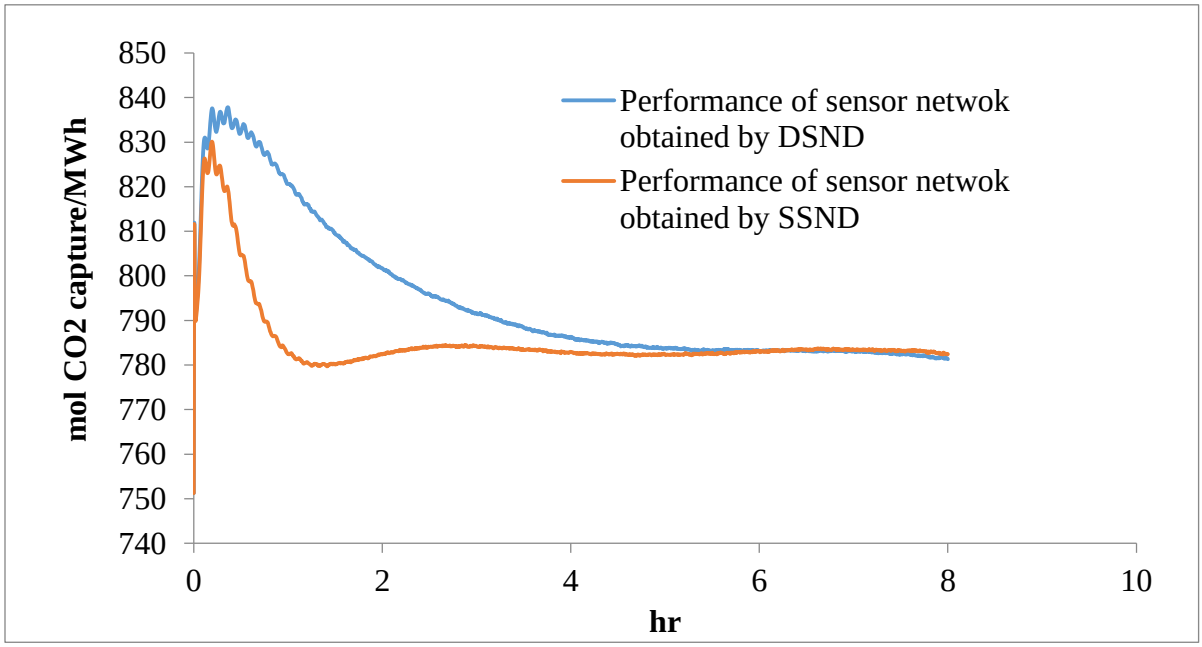

Figure 5.7: Comparison of the efficiency profile using the sensor sets obtained by the SSND and DMSND algorithms.

The efficiency obtained using the sensor sets from the SSND algorithm approaches the optimal efficiency at steady state as would be expected, but the transient profile is significantly inferior to that obtained using the DMSND algorithm.

5.8.2 Impact of Strategies for Reducing Computational Time

It is found that the computational time for a single sensor network budget is significantly reduced by using the incumbent solution strategy. Figure 5.8 shows one of the cases where the incumbent solution interrupts the evaluation of an inferior sensor set in the first generation of GA.

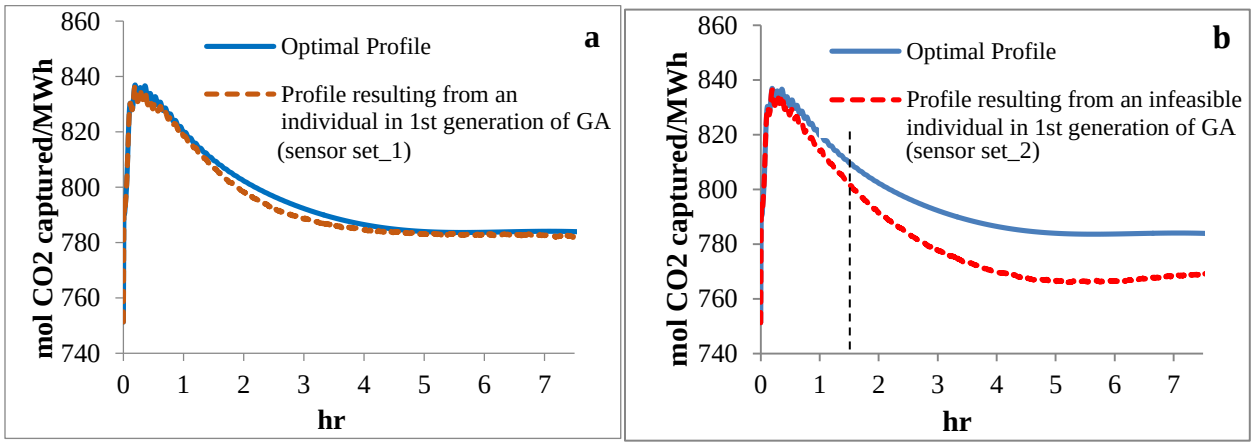

Figure 5.8: AGR efficiency profiles obtained for different candidate set of sensors.

For instance, assume that a Sensor set_1 in Figure 5.8a results in a profile that deviates from the optimal profile by ISE of 8.45×10^3. In this case, 8.45×10^3 is used as incumbent (better fitness) at that instant and in Figure 5.8b the evaluation of the next Sensor set_2 is interrupted after 1.615 hr (marked by dash line) due to exceeding the incumbent ISE. However, the efficiency profile for the additional time (after 1.615 hr in Figure 5.8b) was shown to illustrate that further evaluation of the sensor set would have resulted in significant deviation from the optimal profile. The incumbent ISE gets updated towards the lower value over the successive generation. It should be noted that, in this case, Figure 5.8a and Figure 5.8b both show infeasible sensor sets.

5.8.3 Impact of Various Strategies for Reducing Computational

The impact of the various strategies for reducing DMSND computational time for the AGR case study is presented in Table 5.6. In Table 5.6, 'FOM' denotes full-order model and 'worker' denotes the number of processors used in the parallel computation.

Table 5.6: Impact of Various Strategies for Reducing Computational Time for AGR Case Study

Lower level (each candidate sensor set)	
Strategies used to reduce computation time	***Computation time***
None (*FOM + constant sampling rate*)	17 min 19 sec
MRD-only	11 min 28 secs
ROM-only	5.71 sec
MRD+ROM	3.02 sec
Upper level (case study: each budget) (Above strategies are inclusive)	
Strategies used to reduce computation time	***Computation time (approximate)***
Case study without incumbent (*parallel computation with 4 workers*)	8 hr 30 min
Case study with incumbent (*parallel computation with 4 workers*)	3 hr 30 min – 4 hr
Case study with incumbent (*computation with 1 worker, i.e. no parallelization*)	9-10 hr

5.8.4 Performance Comparison of ROM with FOM

Figure 5.9 shows that the efficiency profile obtained using ROM is very similar to that obtained using the FOM. The integral absolute errors (IAEs) for the full- and reduced-order models have been compared at the top-right corner in Figure 5.9. Again the difference is very small.

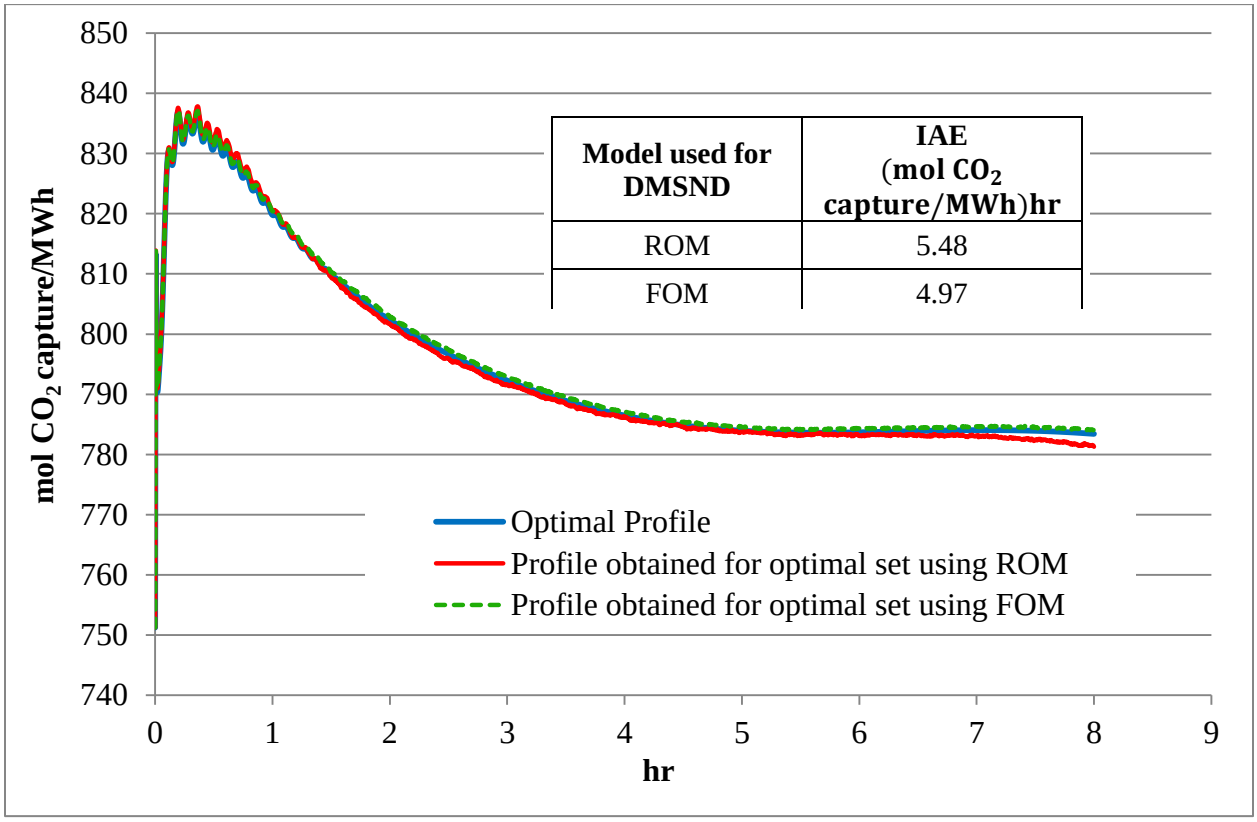

Figure 5.9: Comparison of efficiency profile for the optimal sensor set using ROM and FOM.

Table 5.7 shows the list of 61 measurements in optimal sensor set S_5 obtained from the DMSND algorithm. The set includes 27 temperature sensors, 17 pressure sensors, 9 flow sensors, and 8 CO_2 analyzers. The SSND algorithm in previous work (Paul et al., 2015) found an optimal set of 56 measurements. The comparison shows that a total of 22 measurements (shown as bold in Table 5.7) are the same for both the algorithms. At several locations, the DMSND algorithm placed a sensor similar to the SSND algorithm (*italic*) but selected a different type. It is also

observed that the DMSND algorithm selected several stages in the towers similar to the SSND algorithm but it selected different stage numbers.

It should be noted that the solution to the SND problem is not necessarily unique. Furthermore, the optimal set of sensors can change depending on the disturbances and set point changes and their magnitudes and characteristics. The developed algorithm is generic and can be readily used to study the impact of such changes.

Table 5.7: List of Optimal Set of Sensors

Temperature Sensor

1. [8]H_2S absorber*
2. [12]H_2S absorber*
3. [20]H_2S absorber*
4. **Acid gas knockout drum vapor outlet**
5. **[8]CO_2 absorber***
6. [12]CO_2 absorber*
7. **[1]Solvent stripper***
8. [10]Solvent stripper*
9. Syngas cooler outlet
10. **Off gas from top of H_2S absorber**
11. *Semi-lean solvent cooler inlet*
12. **Rich solvent heater inlet**
13. Solvent Stripper bottom outlet
14. **H_2 recovery flash vessel outlet**
15. **MP flash vessel outlet**
16. **H_2S concentrator vapor outlet**
17. **Stripper gas compressor outlet**
18. Stripped gas cooler outlet
19. Steam to solvent stripper
20. H_2 recovery cooler outlet
21. 2nd MP CO_2 compressor outlet
22. **Vapor of CO_2 flash vessel**
23. *Glycol absorber top outlet*
24. **1st MP CO_2 compressor inlet**
25. 3rd MP CO_2 compressor outlet
26. **1st HP CO2 compressor outlet**
27. **Tail gas to H_2S absorber**

Pressure Measuring Device:

28. [7]H_2S absorber*
29. [25]H_2S absorber*
30. *[4]H_2S concentrator**
31. [3]CO_2 absorber*
32. **[9]CO_2 absorber***
33. *[15]CO_2 absorber**
34. *[1]Solvent stripper**
35. *[9]Solvent stripper**
36. **Lean/rich heat exchanger inlet (rich solvent)**
37. Lean/rich heat exchanger inlet (lean solvent)
38. Lean Selexol pump outlet
39. **H_2 recovery flash vessel**
40. *MP flash vessel outlet*
41. **Acid gas K.O. bottom outlet**
42. H_2 recovery cooler outlet
43. *1st MP CO_2 compressor inlet*
44. *1st HP CO_2 compressor outlet*

Flow sensor

45. *Rich solvent heater inlet*
46. *H_2S concentrator liquid outlet*
47. **H_2S concentrator vapor outlet**
48. Solvent Stripper bottom outlet
49. *H_2 recovery inlet*
50. HP flash vessel inlet
51. *LP flash vessel bottom*
52. Inert stripping gas to H_2S concentrator
53. **H_2O K.O. drum bottom outlet**

CO_2 analyzer

54. [14] *Liquid phase in CO_2 absorber**
55. *Clean syngas*
56. **Liquid phase in H_2S absorber bottom**
57. *H_2S concentrator liquid outlet*
58. **MP flash vessel vapor**
59. **LP flash vessel liquid**
60. *H_2S concentrator vapor outlet*
61. *H_2 recovery compressor inlet*

*Superscript at the beginning of each location denotes stage number

5.9 Conclusion

In this work, a DMSND algorithm has been developed for efficiency maximization of an estimator-based control system. The computational expense is significantly reduced by developing three strategies: MRD, ROM, and use of an incumbent solution to terminate evaluation of infeasible sensor sets. The algorithm is applied to a large-scale acid gas removal unit as part of an IGCC power plant with CO_2 capture. It is observed that the computational time for a single sensor network budget is reduced by more than half by using the incumbent solution. For a given set of sensors, the use of MRD and ROM reduces the DMSND algorithm computation time from about 17 min to 3 sec for a typical sensor set. The use of the ROM results in a very minor deviation from the efficiency profile obtained by using the full-order model. The results show that beyond a threshold budget for the sensor network, the efficiency obtained using the estimator-based control system approaches the optimal efficiency. With further increase in budget, there is minor change in the efficiency profile. It is observed that the sensor network obtained using the SSND algorithm can result in suboptimal transient performance even though the steady-state performance approaches the optimal efficiency as expected.

Chapter 6

Nonlinear Dynamic Model-Based SND Algorithm

The SND algorithms that have been presented in earlier chapters have been developed using a linear process model. Also, a single objective has been considered for optimization. In this chapter, a nonlinear dynamic model-based SND (NDMSND) algorithm is developed for multi-objective optimization for an estimator-based control system while satisfying accuracy requirements for key process variables. A lexicographic approach is used for multi-objective optimization. First the process efficiency is maximized followed by the minimization of sensor budget. The NDMSND algorithm is developed using an unscented KF (UKF) for estimating the key process variables. The NDMSND algorithm can be used to determine optimal location, number and type of sensors for a highly nonlinear system for which a linear model can lead to inaccuracies.

6.1 Literature Review

There are very few works published in the area of NDMSND. Wouwer et al. (2000) have presented an approach to the selection of optimal sensor locations using a nonlinear distributed parameter model of a catalytic fixed-bed reactor for on-line estimation of states and unknown parameters. Karim et al. (2008) have presented a SND methodology for nonlinear continuous-stirred tank reactor using EKF for dynamic data reconciliation. They use genetic algorithm to solve the constrained optimization problem. Alonso et al. (2004) have studied optimal location and type of sensors in a low dimensional nonlinear convection-diffusion-reaction process through an efficient guided search algorithm that minimizes orthonormality distortion. Georges (1995) has used an approach based on nonlinear observability functions (Scherpen, 1993) for determining sensor location. Lopez and Alvarez (2004) have presented geometric approach to determine the degree of estimability for nonlinear systems. However, geometric approaches (Isidori,1995; Hermann and Kerner, 1977) are computationally expensive for determining sensor locations. Nguyen and Bagajewicz (2008) have investigated NDMSND using an equation-based tree search method for the design of a nonlinear sensor network. Singh and Hahn (2005) have

performed an observability analysis of a system over an operating region for placing a single sensor. The authors later extended the analysis and considered measurement redundancy for placing multiple sensors (Singh and Hahn, 2006).

There is hardly any work on SND for efficiency maximization using nonlinear, dynamic models. The main difference in the SND algorithm for efficiency maximization is due to consideration of the estimator-based control system as mentioned earlier. In previous chapters, the SND algorithms have been developed where the cost of sensor sets are used as inequality constraints. Thus the optimization terminates once the sensor set reaches the maximum efficiency and the cost of the sensor set is below or equal to some pre-specified budget. However this does not guarantee that the sensor sets obtained are cost optimal. There may exist other set of sensors that can achieve equal or almost the same efficiency at much lower budget. Therefore, multi-objective optimization where process efficiency is maximized at the minimum budget seems appropriate. A lexicographic approach is used to solve the multi-objective SND problem. First, optimization is performed to maximize process efficiency for a given budget. Then the cost of the sensors is minimized subject to the maximum efficiency obtained from the first optimization.

The NDMSND algorithms are computationally very expensive due to the study of the nonlinear transient behavior of the process for each candidate set of sensors. This computational expense significantly increases as the number of state variables and the number of candidate sensor variables increase. In particular, the solution of the matrix Riccati equation takes significant computational time. However, for the NDMSND algorithm to be usable for large-scale industrial applications, an efficient algorithm is desired that can be solved within reasonable run times using standard computing resources. With this incentive, a reduced order nonlinear model is identified for developing computationally efficient NDMSND algorithm for maximizing the efficiency of large-scale processes. In this NDMSND algorithm, the UKF is used for estimating nonlinear process variables and particular focus is given to its convergence properties.

Usually, nonlinear states are estimated by extended KF (EKF) (Anderson and Moore, 1979). In the last two decades, UKF (Julier and Uhlmann, 2004) has emerged as a popular alternative to EKF. Similar to EKF, UKF also implicitly assumes the prior density as Gaussian whose mean

and covariance are obtained by propagating a set of sigma points. This results in a better approximation of the moments (Julier and Uhlmann, 2004) and also avoids numerical issues related to the linearization step of EKF. Several modifications have been proposed in the literature to account for the non-Gaussianity. These include Gaussian Sum Filter (GSF) (Sorenson and Alspach, 1971; Soderstrom, 2002), Gaussian Sum UKF (Straka et al., 2011), Gaussian Sum Particle Filter (Dunik and Simandl, 2005) and Unscented GSM (Kottakki et al., 2014). These approaches are based on the result that a Gaussian sum can approximate any density to an arbitrary degree of accuracy (Sorenson and Alspach, 1971).

In this chapter, a NDMSND algorithm has been developed that results in a MINLP problem. The NDMSND algorithm is solved by splitting the problem into two parts. The integer programming part is solved by GA while other linear and nonlinear equations in Figure 6.1 are solved sequentially. As discussed below in more detail, this formulation helps in satisfying the linear and nonlinear equality constraints for every combination of integer variables.

This following organization is adopted in this paper. First the nonlinear state estimation algorithms, namely EKF and UKF, are presented. Then the method of system identification and multiobjective optimization using lexicographic ordering are discussed briefly. Subsequently, the NDMSND algorithm for efficiency maximization for an estimator-based control system is presented. This is followed by a discussion on the solution approach to the NDMSND problem. Finally, the application of the NDMSND algorithm to the AGR case study is presented.

6.1.1 EKF:

For a general nonlinear system,

$$x_k = f(x_{k-1}, u_{k-1}) + w_{k-1} \tag{6.1}$$

$$z_k = h(x_k) + v_k \tag{6.2}$$

x_k and z_k are the state and measurement vectors, respectively. $f(.)$ and $h(.)$ are the process and measurement nonlinear vector functions. Random vectors w and v are the model uncertainties and measurement noise which are both assumed to be zero-mean, white noise with known covariance and uncorrelated with the initial state x_0.

The following assumptions are made:

$E[w_k] = 0$ $E[w_k w_k^T] = Q_k$ $E[w_k w_j^T] = 0$ for $k \neq j$ $E[w_k x_0^T] = 0$ for all k

$E[v_k] = 0$ $E[v_k v_k^T] = R_k$ $E[v_k v_j^T] = 0$ for $k \neq j$ $E[v_k x_0^T] = 0$ for all k

$E[w_k v_j^T] = 0$ for all k and j

The designed algorithm for nonlinear state estimation using EKF is presented below:

The initial conditions are:

$x_0^a = \mu_0$ with error covariance P_0

x_0^a is the initial optimal estimate states

Predictor:

$$x_k^f = f(x_{k-1}^a) \tag{6.3}$$

$$P_k^f = J_f(x_{k-1}^a) P_{k-1} J_f^T(x_{k-1}^a) + Q_{k-1} \tag{6.4}$$

where x_k^f is forecast nonlinear states, P_k^f is forecast error covariance, J_f is Jacobian of $f(.)$, and Q_k is process noise covariance matrix

Corrector:

$$x_k^a = x_k^f + K_k(z_k - h(x_k^f)) \tag{6.5}$$

$$K_k = P_k^f J_H^T(x_k^f)\left(J_H(x_k^f) P_k^f J_H^T(x_k^f) + R_k\right)^{-1} \tag{6.6}$$

$$P_k = (I - K_k J_H(x_k^f)) P_k^f \tag{6.7}$$

where P_k is posterior error covariance, J_H is Jacobian of $h(.)$, and R_k is measurement noise covariance matrix.

6.1.2 UKF:

The Unscented Transformation (UT) is the central technique of the UKF for the nonlinear function *y = f(x),* where *x* and *y* are *L* × 1 vectors, and *f* represents the nonlinear functions. Here, *x* is a random variable which is typically assumed to be normally distributed (Gaussian) with mean, $\bar{x}$, and covariance, P_x. The UT provides a statistical alternative to the analytical

linearization approach using Jacobian matrices which is used in the EKF. The UT uses a small set of deterministically selected points, called sigma-points, which are selected, based on the *a priori* conditions, i.e., the points are selected from the assumed prior distribution. The spread of these points or the confidence level from the prior distribution is determined based on the selected scaling parameters for the UT. The values of the scaling parameters affect the spread of the sigma-points as well as the weight vectors that are used in reconstructing the *a posteriori* (after the transformation) statistics.

The scaling of the UT can be fully represented by three scaling parameters (Julier and Uhlmann, 1997; Wan and van der Merwe, 2002). The primary scaling parameter, α_{sc}, determines the spread of the sigma-points. Smaller α leads to a tighter (closer) selection of sigma-points, while larger α_{sc} gives a wider spread of sigma-points. The secondary scaling parameter, β_{sc}, is used to include information about the prior distribution (for Gaussian distributions, β_{sc} = 2 is optimal). The tertiary scaling parameter, κ_{sc}, is usually set to 0 (Julier and Uhlmann, 1997). Using these three scaling parameters, an additional scaling parameter, λ, and weight vectors, W_m (mean) and W_c (covariance) are defined.

Off-line calculations (*each step completed once before filtering*):

Define scaling parameters and weight vectors

$$\lambda = \alpha_{sc}^2(L + \kappa) - L \tag{6.8}$$

$$W_0^m = \frac{\lambda}{L+\lambda} \tag{6.9}$$

$$W_0^c = \frac{\lambda}{L+\lambda} + 1 - \alpha_{sc}^2 + \beta_{sc} \tag{6.10}$$

$$W_i^m = W_i^c = \frac{1}{2(L+\lambda)}\ , i = 1,2, \dots\dots, 2L \tag{6.11}$$

Initialization (*each step completed once before filtering*):

$$Q_k = E[w_k w_k^T], \quad R_k = E[v_k v_k^T] \tag{6.12}$$

$$\hat{x}_0 = E[x_0],\ P_0 = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] \tag{6.13}$$

Executing the filter recursively (*each step at every discrete-time*):

Step 1: Generate the Sigma-Points

$\sqrt{P_{k-1}} = chol(P_{k-1})$ (*Lower Cholesky decomposition*)

$$x_{k-1} = \begin{bmatrix} \hat{x}_{k-1} & \hat{x}_{k-1} + \left(\sqrt{L+\lambda}\sqrt{P_{k-1}}\right)i & \hat{x}_{k-1} - \left(\sqrt{L+\lambda}\sqrt{P_{k-1}}\right)i \end{bmatrix} \quad (6.14)$$

Step 2: Prediction Transformation

$$x_{k|k-1}^{(i)} = f\left(x_{k-1}^{(i)}, u_k\right), \;\; i = 1,2, \ldots \ldots 2L \quad (6.15)$$

$$\hat{x}_{k|k-1} = \sum_{i=0}^{2L} W_i^m \, x_{k|k-1}^{(i)} \quad (6.16)$$

$$P_{k|k-1} = Q_{k-1} + \sum_{i=0}^{2L} W_i^c \, (x_{k|k-1}^{(i)} - \hat{x}_{k|k-1}) \, (x_{k|k-1}^{(i)} - \hat{x}_{k|k-1})^T \quad (6.17)$$

Step 3: Observation Transformation

$$\psi_{k|k-1}^{(i)} = h\left(x_{k|k-1}^{(i)}, u_k\right) \quad (6.18)$$

$$\hat{y}_{k|k-1} = \sum_{i=0}^{2L} W_i^m \, \psi_{k|k-1}^{(i)} \quad (6.19)$$

$$P_k^{yy} = R_{k-1} + \sum_{i=0}^{2L} W_i^c \, (\psi_{k|k-1}^{(i)} - \hat{y}_{k|k-1}) \, (\psi_{k|k-1}^{(i)} - \hat{y}_{k|k-1})^T \quad (6.20)$$

$$P_k^{xy} = \sum_{i=0}^{2L} W_i^c \, (\mathrm{x}_{k|k-1}^{(i)} - \hat{x}_{k|k-1}) \, (\psi_{k|k-1}^{(i)} - \hat{y}_{k|k-1})^T \quad (6.21)$$

Step 4: Measurement update

$$K_k = P_k^{xy} (P_k^{yy})^{-1} \quad (6.22)$$

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - \hat{y}_{k|k-1}) \quad (6.23)$$

$$P_k = P_{k|k-1} - K_k P_k^{yy} K_k^T \quad (6.24)$$

6.1.3 System Identification

For the nonlinear SND, the nonlinear process model needs to be identified by using the input-output data. Several methodologies have been developed for nonlinear model identification in different application areas.

6.1.3.1 Neural Network Method

One of the widely used methods for identifying nonlinear model is the neural network (Narendra, 1997). The neural network model coupled with the linear state space model can be used for representing nonlinear, dynamic systems (Sentoni et al., 1998). In this work, a linear in parameter (LIP) method is used for nonlinear model identification.

6.1.3.2 Linear in Parameter Method

The LIP method is widely used in different areas of application because of its simple structure and well-developed characteristic. The general form of LIP can be written as

$$y(k) = x(k)H \tag{6.25}$$

$$\text{Where, } H = (X^T X)^{-1} X^T Y \tag{6.26}$$

y(k) is the output at time instant *k*. *x(k)* is the regression vector. The parameter vector *H* is estimated by least squares method.

A nonlinear process model of any degree can be obtained using the LIP method. NAARX (Nonlinear additive auto-regressive with exogenous input) models are represented by:

$$y(k) = \sum_{i=0}^{g} f_i\big(u(k-i)\big) + \sum_{j=1}^{h} s_j\big(y(k-j)\big) \tag{6.27}$$

In Eq. (6.27) f and s are the polynomials of order N_1 and N_2. The input memory g and the output memory h are determined by number of sampling. In this work, NAARX model including cross-terms is considered:

$$y(k) = \sum_{i=0}^{g} H_1(i, n_1)u(k-i)^{n_1} + \sum_{j=1}^{h} H_2(j, n_2)y(k-j)^{n_2} + \sum_{i=0}^{g}\sum_{j=1}^{h} H_3(i,j)u(k-i)u(k-j) + \cdots \sum_{i=0}^{g} \dots \sum_{l_1=0}^{g} H_{n_1+1}(i, \dots, l_1)u(k-i) \dots u(k-l_1), \quad n_1 = 1{:}N_1, n_2 = 1{:}N_2 \tag{6.28}$$

6.1.4 Multi-Objective Optimization

Multi-objective optimization techniques are used to handle problems where more than one objective is to be maximized or minimized. This type of problem usually has at least two conflicting objectives. It is difficult to reach their optimal values simultaneously. Attaining one will result in degrading the other(s). Interactions among different objectives give rise to a set of compromised solutions, largely known as trade-off solutions. There exist several multi criteria decision problems where the Pareto-optimality is not guaranteed (Miettinen, 2002). A lexicographic approach is one such technique that can guarantee Pareto-optimality of multi-objective optimization problems. Several mathematical nonlinear lexicographic optimizations have been reported by Behringer (1977). This type of optimization is studied by arranging objective functions in lexicographic order i.e. one goal is more important than other. First the most important goal is optimized. Therefore, a lexicographic optimization can be written as:

$$\text{Min } f_1(x),\ f_2(x), \dots\ f_k(x) \tag{6.29}$$

s.t.

$$x \in S$$

where $f_1(x)$ is the most important objective function and $f_k(x)$ is the least important. In lexicographic ordering, if the most important objective function has a unique solution, then the other objectives do not impact the solution obtained in the optimization of most important function.

In the NDMSND algorithm, more importance is given to process efficiency maximization than to minimization of budget for sensors. Thus, first an optimal set of sensors is determined that results in minimum deviation from optimal efficiency subject to budget constraints and

estimation accuracy. Later, the budget for the sensor is minimized subject to the previously obtained optimal value for the efficiency.

6.2 Nonlinear Model-Based SND Algorithm Development

The estimator-based control system that is used to develop the NDMSND algorithm is shown in Figure 6.1. The process is perturbed by a disturbance u_d. The estimator receives the noisy measurements, $y_{noisy,\beta}$, from the sensor network and estimates the controlled variables ($\hat{y}_{cont,est}$) and the variables for monitoring ($\hat{y}_{mon}$) process performance. The controller(s) then implement(s) the corrective action on the process based on the estimated controlled variables.

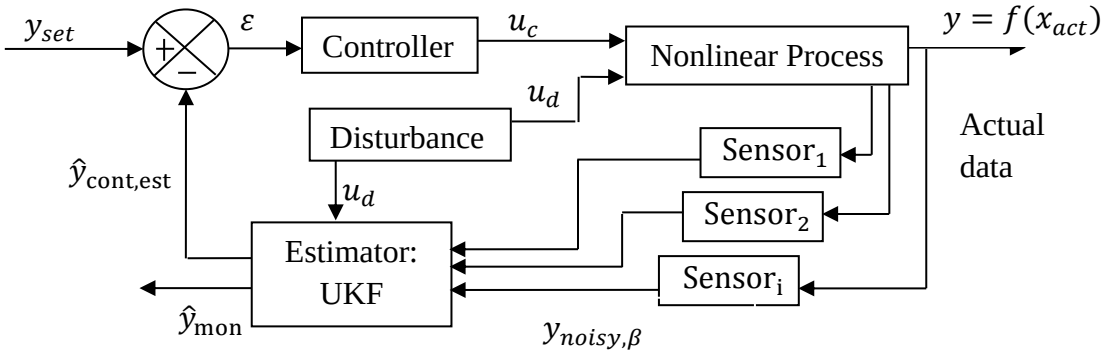

Figure 6.1: Schematic of the UKF-based control system for development of the NDMSND algorithm.

For developing the NDMSND algorithm, first the set of equations corresponding to each block of the estimator-based control system is organized. The estimator block in Figure 6.1 is considered to be a UKF.

The NDMSND algorithm is designed for multi-objective optimization where process efficiency is maximized followed by minimization of budget for sensors. Thus priority-based optimization is performed by using lexicographic ordering as shown below:

STEP A1

$$Min \sum_{i=1}^{k} \left(\Delta\eta_{est,i}\right)^2 \tag{6.30}$$

$$\sum_{\forall i} c_i\beta_i \leq b \quad ; \quad \beta_i = 0,1 \qquad \forall i \in N_s \tag{6.31}$$

$$\sum_{i=1}^{k} Esterr_i{}^2 < tol_1 \tag{6.32}$$

STEP A2

$$Min \sum_{\forall i} c_i\beta_i \tag{6.33}$$

$$\sum_{i=1}^{k} \left(\Delta\eta_{est,i}\right)^2 \leq \text{Optimal value obtained from STEP A1} \tag{6.34}$$

$$\sum_{i=1}^{k} Esterr_i{}^2 < tol_1 \tag{6.35}$$

In STEP A1, the integer problem is solved subject to the constraints in each block of Figure 6.1. The additional constraints on the budget and the estimation accuracy are represented by Eqs. (6.31) and (6.32), respectively, where c_i denotes the cost of a sensor of type i. The variables in Eq. (6.30) are a function of x_{act} in the estimator-based control system.

In STEP A2, the budget for the sensors is minimized subject to the same constraints as those in STEP A1 except that the additional constraints on the deviation in efficiency and the estimation accuracy as denoted by Eqs. (6.34) and (6.35), respectively. $\Delta\eta_{est}$ and $Esterr$ are defined as,

$$\Delta\eta_{est} = \eta_{opt} - \eta(x_{act}, \beta) \tag{6.36}$$

$$Esterr = y_{ma,act} - \hat{y}_{ma} \tag{6.37}$$

As shown in Eq. (6.36), $\Delta\eta_{est}$ is the difference between the optimal efficiency (η_{opt}) and $\eta(x_{act},\beta)$, the efficiency obtained in the estimator-based system. Eq. (6.37) is used to calculate the estimation error in key variables, denoted by *Esterr*, by comparing the actual values with the estimated values of process variables.

The constraint on estimation accuracy in Eq. (6.32) is eliminated and the objective function is reformulated for STEP A1 as follows,

STEP B1

$$Min\left[\lambda_1 \sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2 + \lambda_2 \sum_{i=1}^{k} Esterr^2\right] \tag{6.38}$$

$$\sum_{\forall i} c_i\beta_i \;\leq\; b \quad ; \quad \beta_i = 0{,}1 \qquad \forall i \in N_s$$

In Eq. (6.38), λ_1 and λ_2 are the weighting factors for deviation in efficiency and estimation accuracy, respectively.

Similarly, STEP A2 is reformulated as,

STEP B2

$$Min\left[\acute{\lambda_1} \sum_{\forall i} c_i\beta_i \; + \acute{\lambda_2} \sum_{i=1}^{k} Esterr^2\right] \tag{6.39}$$

$$\sum_{i=1}^{k}\left(\Delta\eta_{est,i}\right)^2 \;\leq\; k' \times \text{Optimal value obtained from STEP 1}$$

In Eq. (6.39), $\acute{\lambda_1}$ and $\acute{\lambda_2}$ are the weighting factors for budget and estimation accuracy, respectively. In STEP B2, the deviation in efficiency remains within a bound for the goal factor k'.

6.3 NDMSND Algorithm

The NDMSND algorithm shown in Figure 6.2 is solved using the GA. In the first generation of GA, the algorithm creates an initial population consisting of candidate sets (β) of sensors. For every candidate set, the initial control action $u_{c,k-1}$(deviation variable) is zero. As the disturbances are introduced, and/or the process or the controller setpoint(s) is(are) changed, controlled variables deviate from the setpoint. Estimated controlled variables are obtained from the estimator (UKF) given the available measurements (selected by GA) and nonlinear process model. The estimator, controller equations, and process model are solved sequentially at every discrete point in time in the face of disturbance/setpoint change. Control action continues until the new steady-state condition is reached. Over the period of process response, the efficiency profile is calculated corresponding to a sensor set that is then scored by a fitness function (objective function). Some sets of sensors with higher fitness value are classified as elite and the GA proceeds to the next generation. The remaining sets of sensors in the next generation are obtained by selection, cross-over, and mutation based on their fitness. The set of candidate sensors evolves over the successive generations and the algorithm is halted once it satisfies the

termination criteria. The final set of sensors obtained from this algorithm yields optimal efficiency given the budget constraints and constraints on estimation accuracy. It should be noted that the lower budget case studies may fail to satisfy the estimation accuracy. This step does not guarantee the minimization of budget.

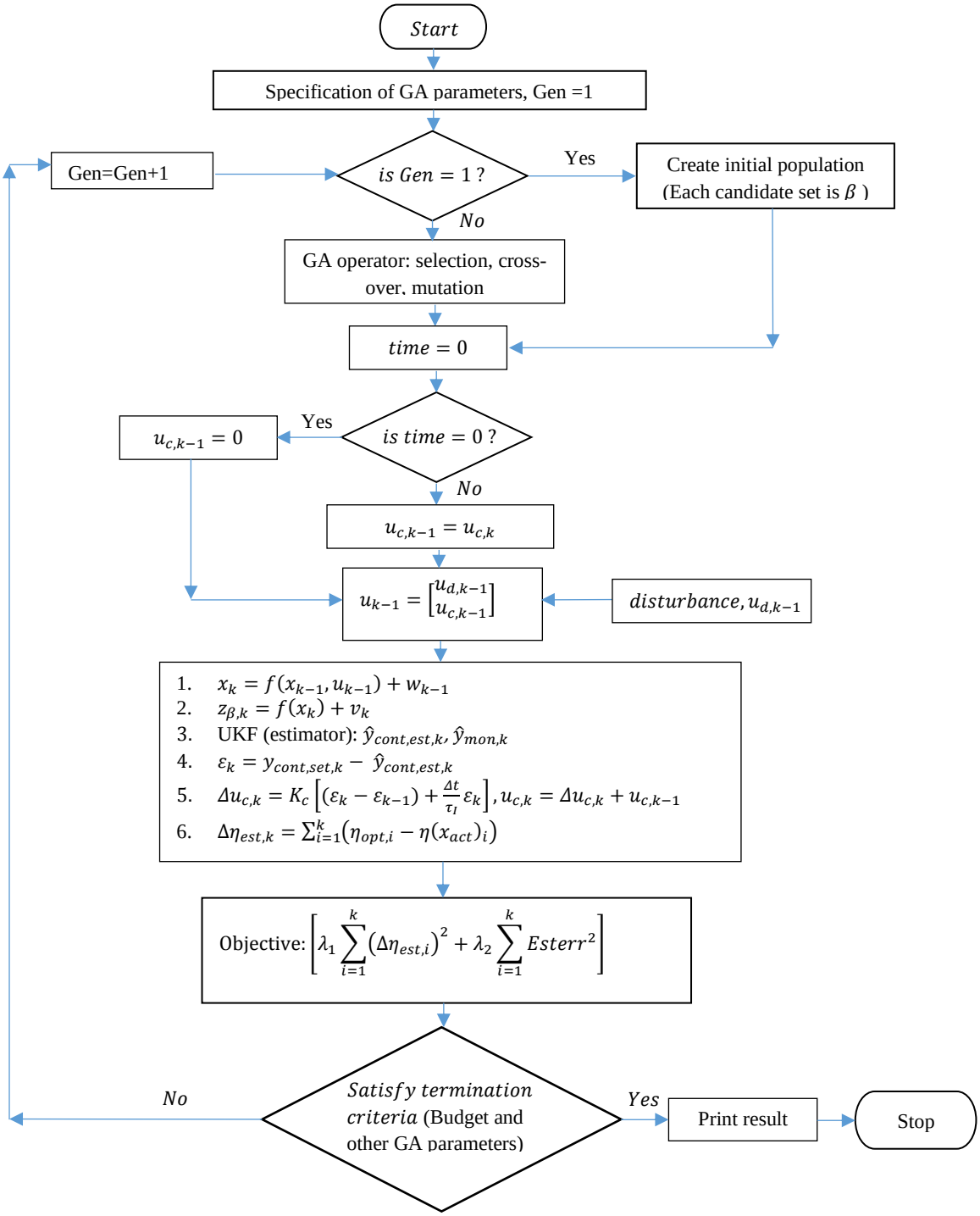

Figure 6.2: Flowchart of the NDMSND algorithm for maximizing efficiency.

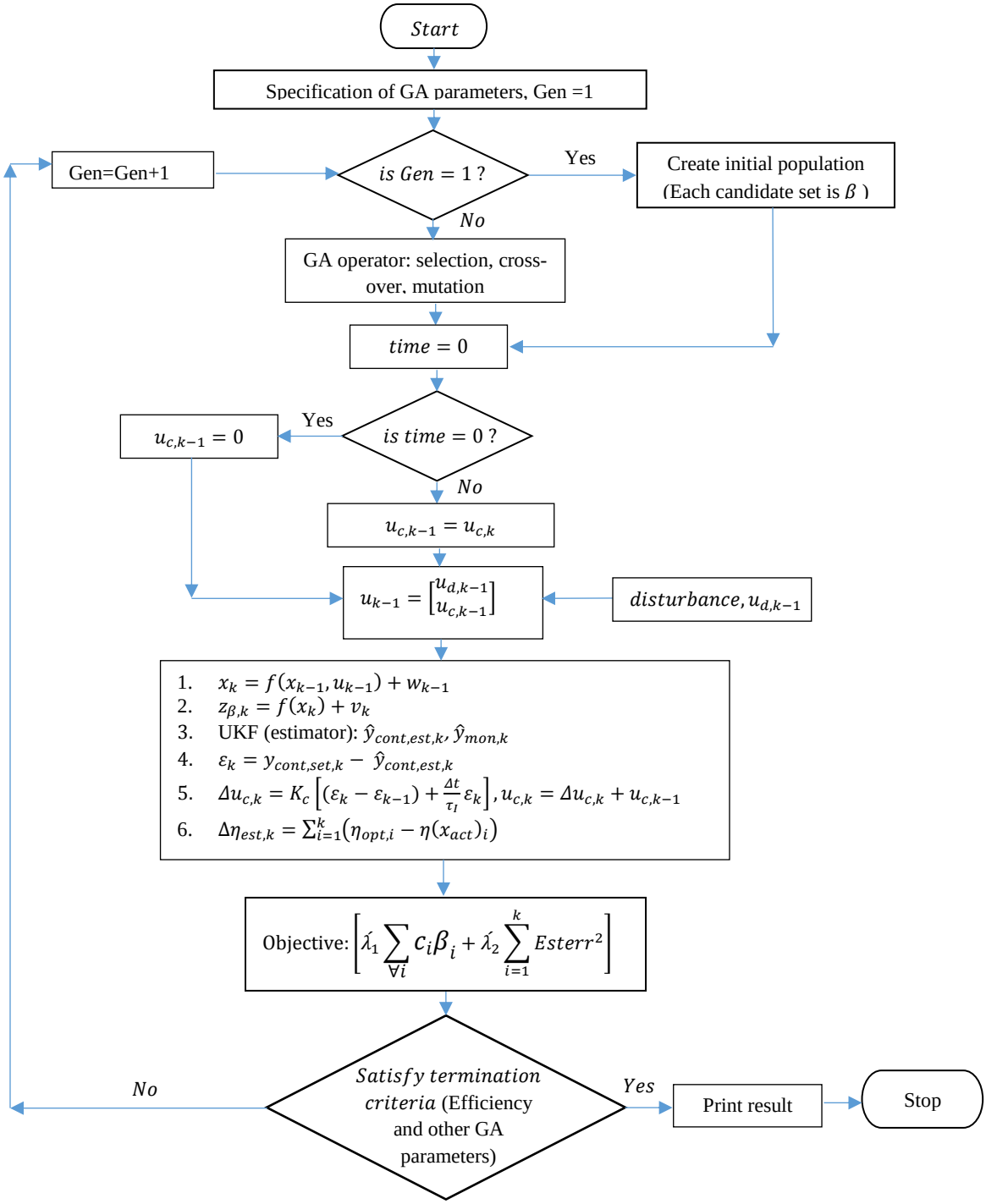

Figure 6.3: Flowchart of the NDMSND algorithm for minimizing budget for sensors.

Figure 6.3 shows the algorithm for minimizing budget for sensors. The estimation accuracy and the efficiency obtained for optimal set of sensors for different budgets are used as constraints. The same optimal set of sensors is obtained if the solution is unique and no lower budget can achieve the previously obtained maximum efficiency within the given tolerance.

6.4 Case Study

This section illustrates the application of the NDMSND algorithm in the nonlinear AGR process as part of an IGCC power plant with CO_2 capture (see Section 3.4). The nonlinear model of the AGR process is identified by a NAARX model as mentioned before. The nonlinear models, involving 62 dominant state variables, are 2nd degree polynomials including cross terms. Measurement equations are also 2nd degree polynomials obtained by NAARX model. Additive Gaussian white noise is used for process model to capture the un-modeled process dynamics and for measurement equations to characterize measurement noise. Variations in the syngas flowrate are considered as the disturbance. The flowrate disturbance is simulated by changing the inlet pressure of the syngas to the AGR unit. It should be noted that even though the study considered only one disturbance, the SND algorithm is generic and additional disturbances and change in the controller setpoints can be readily implemented.

Efficiency of the AGR unit is defined in Eq. (6.40) where the numerator represents the amount of CO_2 captured while the denominator is the MWh power consumption.

$$\eta(x_{act}, \beta) = \frac{F_{CO_2,in}(x_{act}) - F_{CO_2,out}(x_{act})}{aF_{solvent}(x_{act}) + \sum_{c=1}^{3} P_c(x_{act})} \tag{6.40}$$

Four different types of commonly used sensors have been considered: temperature (*T*), pressure (*P*), flow (*F*), and composition (z_{H_2S} and z_{CO_2}).

The AGR process flowsheet is studied and 42 potential sensors are being considered in the current formulation of the NDMSND algorithm. A smaller number of measurement locations is considered for reducing computational expense while using UKF to estimate the states. Table 6.1 shows the distribution of these candidate sensors in the AGR unit.

Table 6.1: Candidate Sensor Locations in the AGR Unit

Temperature Sensors	**Flow Sensors**
1. Syngas Cooler Inlet	23. Inlet to H_2O K.O. Drum
2. Inlet to H2O K.O. Drum	24. Clean Syngas at the Top of CO_2 absorber
3. Top Outlet of H_2O K.O. Drum	25. Off Gas from Top of H_2S Absorber
4. Clean Syngas at the Top of CO_2 absorber	26. H_2S Concentrator Vapor Outlet
5. Stripped Gas Cooler Outlet	27. Stripped Solvent at the Bottom of Selexol Stripper
6. Off Gas from Top of H_2S Absorber	28. Top Outlet of Acid Gas K.O
7. Rich Solvent at H_2S Absorber Bottom	**CO_2 Analyzers**
8. Rich Solvent at inlet to H_2S Concentrator	29. H_2O K.O. Drum Bottom Outlet
9. N_2 Gas Flow to H_2S Concentrator	30. Off Gas from Top of H_2S Absorber
10. H_2S Concentrator Vapor Outlet	31. H_2S Concentrator Vapor Outlet
11. Stripped Solvent at the Bottom of Selexol Stripper	32. Top Outlet of Acid Gas K.O
12. Stripped Solvent at the Bottom of Selexol Stripper	33. Selexol Stripper Top Outlet
Pressure Sensors	**H_2S Analyzers**
13. Inlet to H_2O K.O. Drum	34. H_2O K.O. Drum Bottom Outlet
14. Top Outlet of H_2O K.O. Drum	35. Loaded solvent at the bottom of CO_2 absorber
15. Clean Syngas at the Top of CO_2 absorber	36. Off Gas from Top of H_2S Absorber
16. Loaded solvent at the bottom of CO_2 absorber	37. Rich Solvent at H_2S Absorber Bottom
17. Rich Solvent at inlet to H_2S Concentrator	38. H_2S Concentrator Vapor Outlet
18. Stripped Solvent at the Bottom of Selexol Stripper	39. Stripped Solvent at the Bottom of Selexol Stripper
19. Top Outlet of Acid Gas K.O	40. Top Outlet of Acid Gas K.O
20. Valve Outlet at the Bottom of Acid Gas K.O. Drum	41. Valve Outlet at the Bottom of Acid Gas K.O. Drum
21. Selexol Stripper Top Outlet	42. Selexol Stripper Top Outlet
22. Stripped Solvent at the Bottom of Selexol Stripper	

The range of inaccuracy of commercially available sensors can be found in Liptak (2003). Estimator-based control action is implemented for the primary controlled variables that have been identified by Jones et al. (2014) for this AGR unit.

The developed NDMSND algorithm is implemented by using the GA available in the global optimization toolbox in MATLAB. GA creates an initial population of the candidate set of sensors and over the successive generation it reaches the optimal set of sensors.

In this work, the value of the weighting factors in Eq. (6.38) is chosen by trial and error by considering the trade-off between the desired weight for estimation accuracy versus deviation in efficiency. In this case study, the following values are considered: λ_1 =0.1 and $\lambda_2 = \begin{bmatrix} 1 & 0 \\ 0 & 10 \end{bmatrix}$.

6.5 Results

6.5.1 Nonlinear AGR Model

The identified nonlinear AGR model matches the process transient response of the rigorous AGR model in APD® within acceptable limits. Figures 6.5a and 6.5b show the response in process variables, COS molar holdup in cooler located immediately after the HP CO_2 compressor and H_2O molar holdup on first tray of CO_2 absorber, respectively, subject to sequential step change in multiple disturbances for the pressure (Figure 6.4a) and the CO_2 flowrate (Figure 6.4b) in inlet syngas.

Figures 6.7a and 6.7b show the response in process variables, CO_2 molar holdup on first tray of CO_2 absorber and CO_2 molar holdup on fifth tray of SELEXOL stripper, respectively, subject to a single step change in the same disturbances as shown in Figures 6.6a and 6.6b.

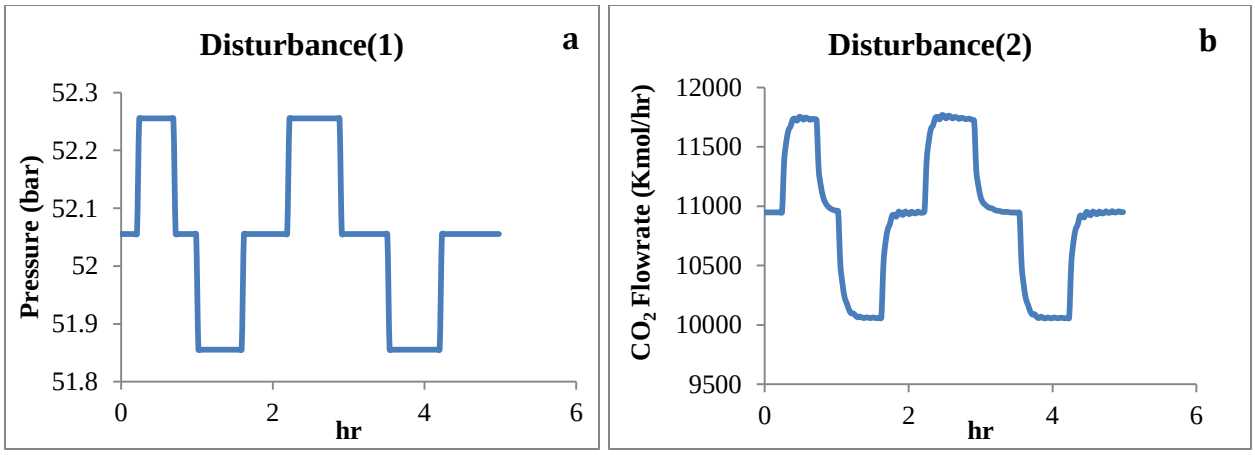

Figure 6.4: Sequential change in disturbances: (a) Pressure and (b) CO_2 flowrate in inlet syngas.

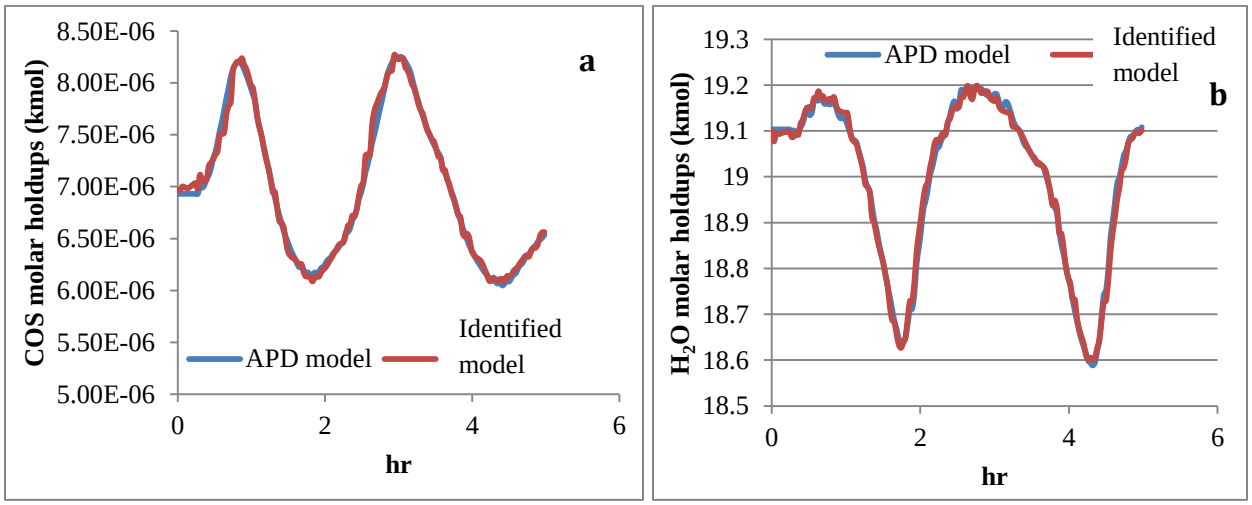

Figure 6.5: Response in (a) COS molar holdup in cooler located immediately after HP CO_2 compressor and (b) H_2O molar holdup on first 1st tray of CO_2 absorber.

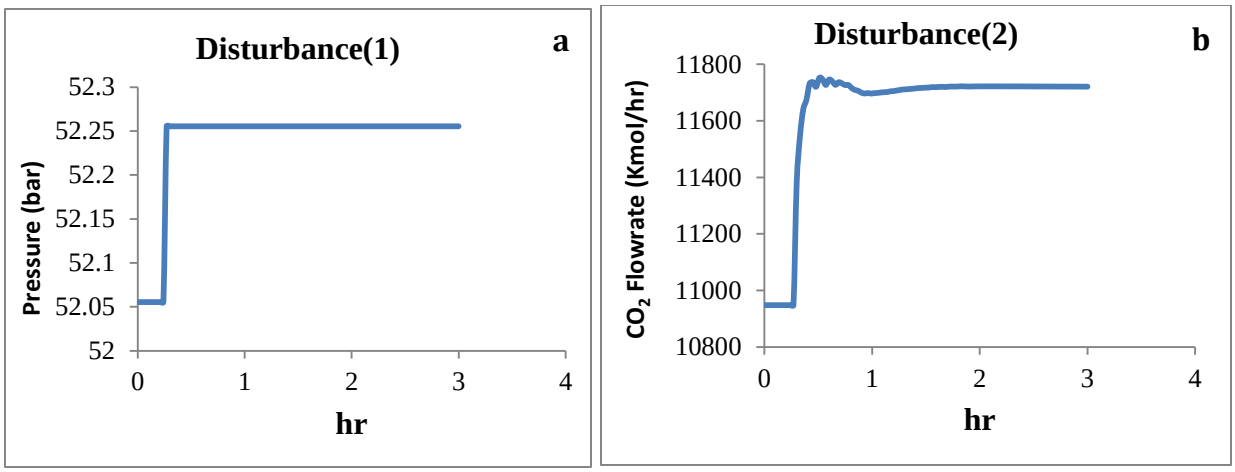

Figure 6.6: Step change in disturbances: (a) Pressure and (b) CO_2 flowrate in inlet syngas.

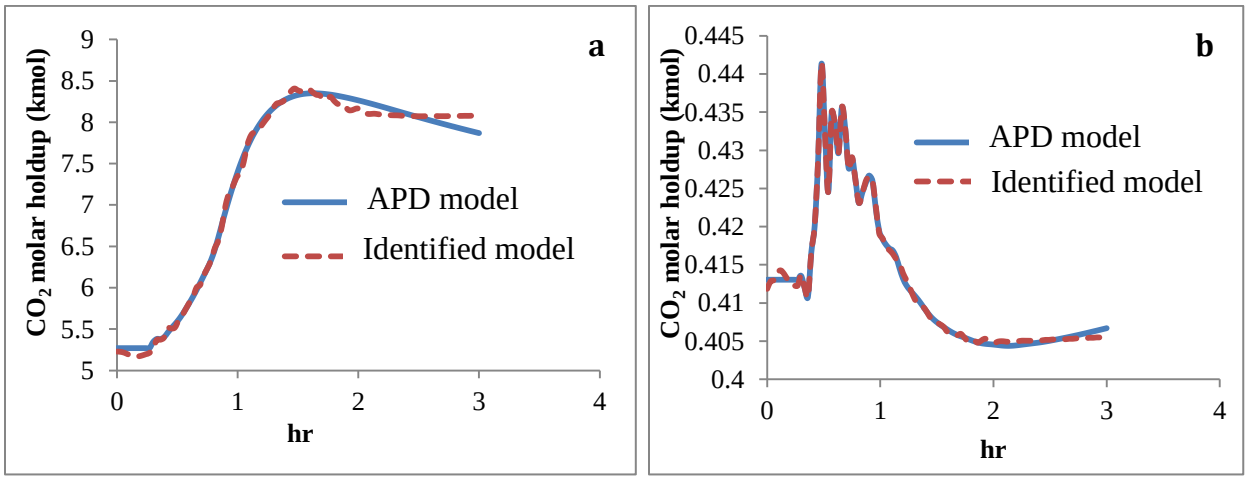

Figure 6.7: Response in (a) CO_2 molar holdup on first tray of CO_2 absorber and (b) CO_2 molar holdup on fifth tray of SELEXOL stripper.

6.5.2 NDMSND Results

Table 6.2 shows the results of the case studies for different budgets while maximizing CO_2 capture efficiency. The case studies are performed subject to the disturbance of 0.3 bar step increase in inlet syngas pressure. The disturbance rejection period is 6 hr. The case studies in Table 6.2 show that as the budget increases, the number, location and type of sensors change, which improve the estimation accuracy of the controlled variables. This, in turn, increases

efficiency (by decreasing deviation $\Delta\eta_{est}$ up to the goal factor, k' =1.12. Table 6.3 shows the results while minimizing budget.

Table 6.2: Number of Sensors, Value of Objective Functions and Integral Deviation from Optimal Efficiency for Different Budget While Maximizing Efficiency

Case Study	Total no.	Budget	Fitness (Objective)	$\sum \Delta\eta_{est}$ (mol CO_2/MWh)
1a	24	$321,600	92.84	8.17
2a	21	$65,400	230.62	14.79
3a	22	$47,200	441.61	21.05

Table 6.3: Number of Sensors, Value of Objective Function and Integral Deviation from Optimal Efficiency for Different Budget While Minimizing Budget

Case Study	Total no.	Budget	Fitness (Objective)	$\sum \Delta\eta_{est}$ (mol CO_2/MWh)
1b	24	$151,200	95.99	8.44
2b	22	$63,400	246.79	15.12
3b	20	$40,400	547.63	23.07

It should be noted that the GA terminates optimization once the maximum efficiency is found below a pre-specified budget (linear inequality constraints). Thus the maximum efficiency obtained at that budget does not guarantee the minimal cost of the sensors. Different sets of sensors with different cost might result in similar efficiency. The results of different case studies presented in Table 6.3 show that almost similar efficiency can be achieved even with lower budget.

Table 6.4 shows the analysis of sensor sets obtained from multi-objective optimization. It is observed that while maximizing efficiency, GA selects more expensive sensors for achieving desired estimation accuracy in comparison to the sensors that are selected while minimizing the cost. At higher budget of $321,600, four H_2S analyzers are selected which account for more than half of the budget but the corresponding Step 2 solution (cost minimization) only selects one H_2S analyzer. In this case, the sensor network cost is reduced to $151,200 while the value of the fitness function is almost same for both steps (efficiency and budget optimization). In the 2nd case study in Tables 6.2 and 6.3, the budget reduction is not significant, but the efficiency values are similar. Interestingly, instead of selecting more CO_2 analyzers, the Step 2 sensor placement

algorithm selects more flow sensors for a budget of $63,400 in comparison to Step 1 sensor placement that selects more CO_2 analyzers for a budget of $65,400.

Table 6.4: Analysis of Sensor Sets in Table 6.2 and 6.3

Sensors	Cost	Solution Sets from Table 6.2 (Maximizing Efficiency)			Solution Sets from Table 6.3 (Minimizing Budget)		
		$321,600	$65,400	$47,200	$151,200	$63,400	$40,400
Temperature	$1000	8	8	10	8	8	9
Pressure	$2200	8	7	6	6	7	7
Flow	$4000	4	3	6	5	5	4
H_2S Analyzer	$70,000	4	---	---	1	---	---
CO_2 Analyzer	$10,000	---	3	---	4	2	---

6.5.3 Transient Performance of the SSND vs. NDMSND

Figure 6.8 shows the performance comparison of the sensor network obtained by the SSND algorithm (Paul et al., 2015) and the NDMSND algorithm. The efficiency obtained using the sensor sets from the SSND algorithm approaches the optimal efficiency at steady state as would be expected, but the transient profile is significantly inferior to that obtained using the NDMSND algorithm.

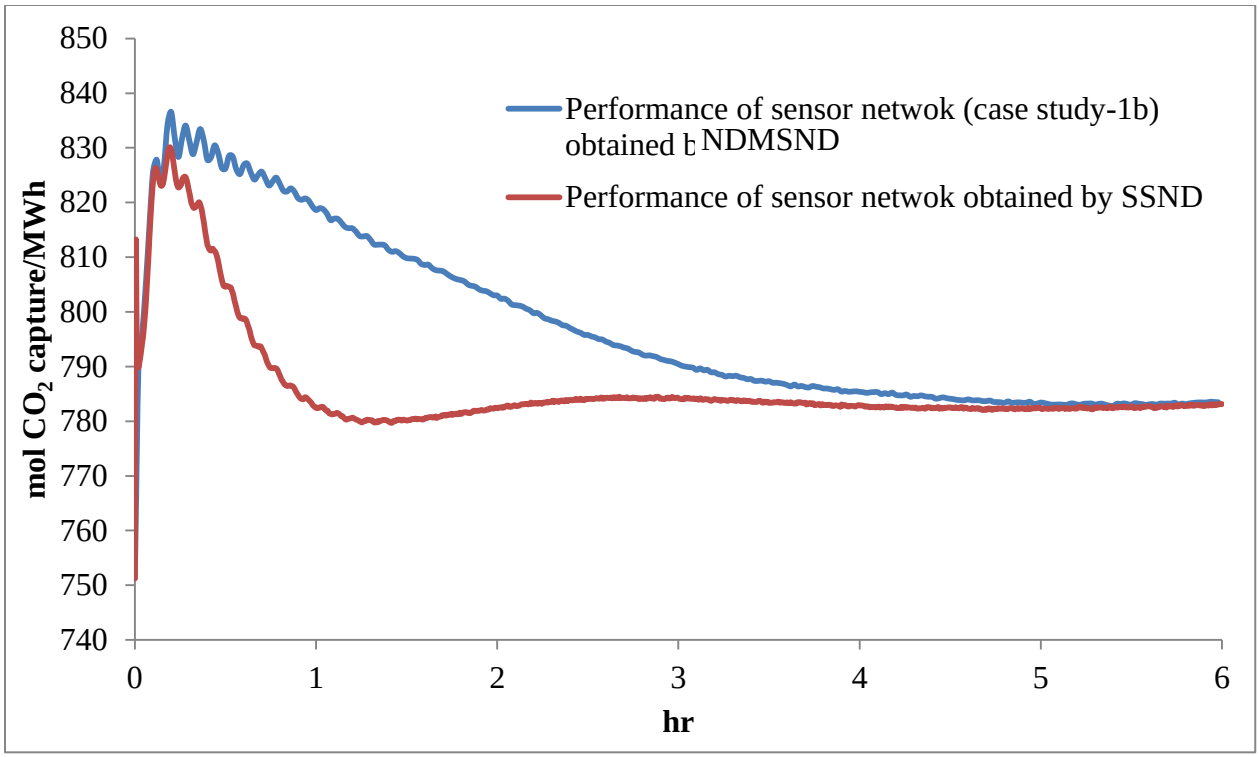

Figure 6.8: Comparison of the efficiency profile using the sensor sets obtained by the SSND and NDMSND algorithms.

6.5.4 Transient Performance of the DMSND vs. NDMSND

Figure 6.9 shows the performance comparison of the sensor network obtained by DMSND algorithm and NDMSND algorithm. The efficiency obtained using the sensor sets from the NDMSND algorithm and DMSND algorithm matches at steady state and compares well during transient response. However, due to use of the nonlinear model and UKF, the desired estimation accuracy is obtained even at lower budget in NDMSND algorithm.

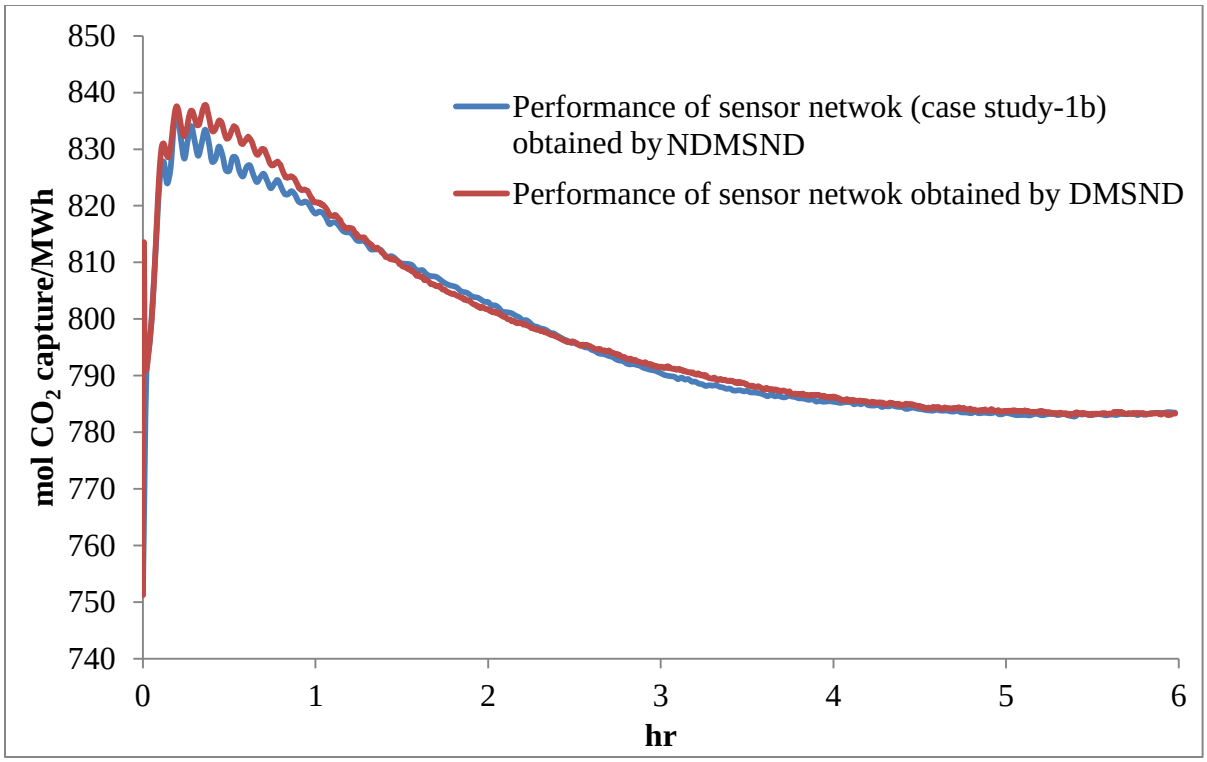

Figure 6.9: Comparison of the efficiency profile using the sensor sets obtained by the DMSND and NDMSND algorithms.

6.5.5 Performance Comparison of Case Study-1a with -1b

Figure 6.10 shows that the efficiency profile obtained for Case Study-1a is very similar to that obtained for Case Study-1b. The fitness (objective) values for Case Study-1a and -1b have been compared in Tables 6.2 and 6.3, respectively. Again the difference is very small.

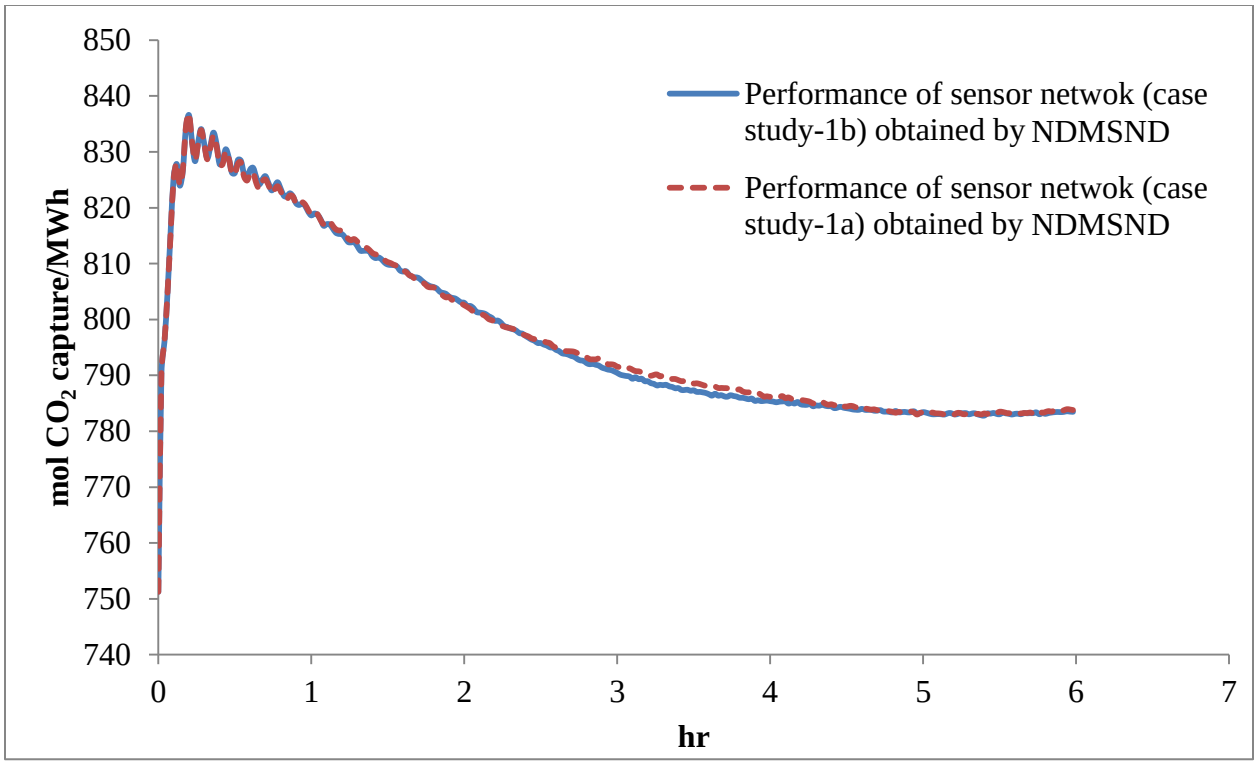

Figure 6.10 Comparison of efficiency profile for the optimal sensor set using case study-1b and -1a.

6.5.6 Net Present Value (NPV) Analysis

The (partial) net present value (denoted pNPV) for the AGR system can be calculated by the following equation by just considering the sensors cost and operating cost due to cost of electricity due to CO_2 capture:

$$pNPV = -budget(\$) - P\left(\frac{(1+i)^n-1}{i(1+i)^n}\right)\left(\frac{1}{1+i}\right) \quad (41)$$

$$P = \frac{MWh\times 1000}{mol\ CO_2\ capture} \times \frac{mol\ CO_2\ capture}{year} \times \frac{price\ (\$)\ of\ electricity}{KWh} \quad (42)$$

Following assumptions are made: plant life=10 yr, period of operation (n)=9 yr, i=0.1

Obviously, pNPV will be negative, and therefore, a lower value is preferred.

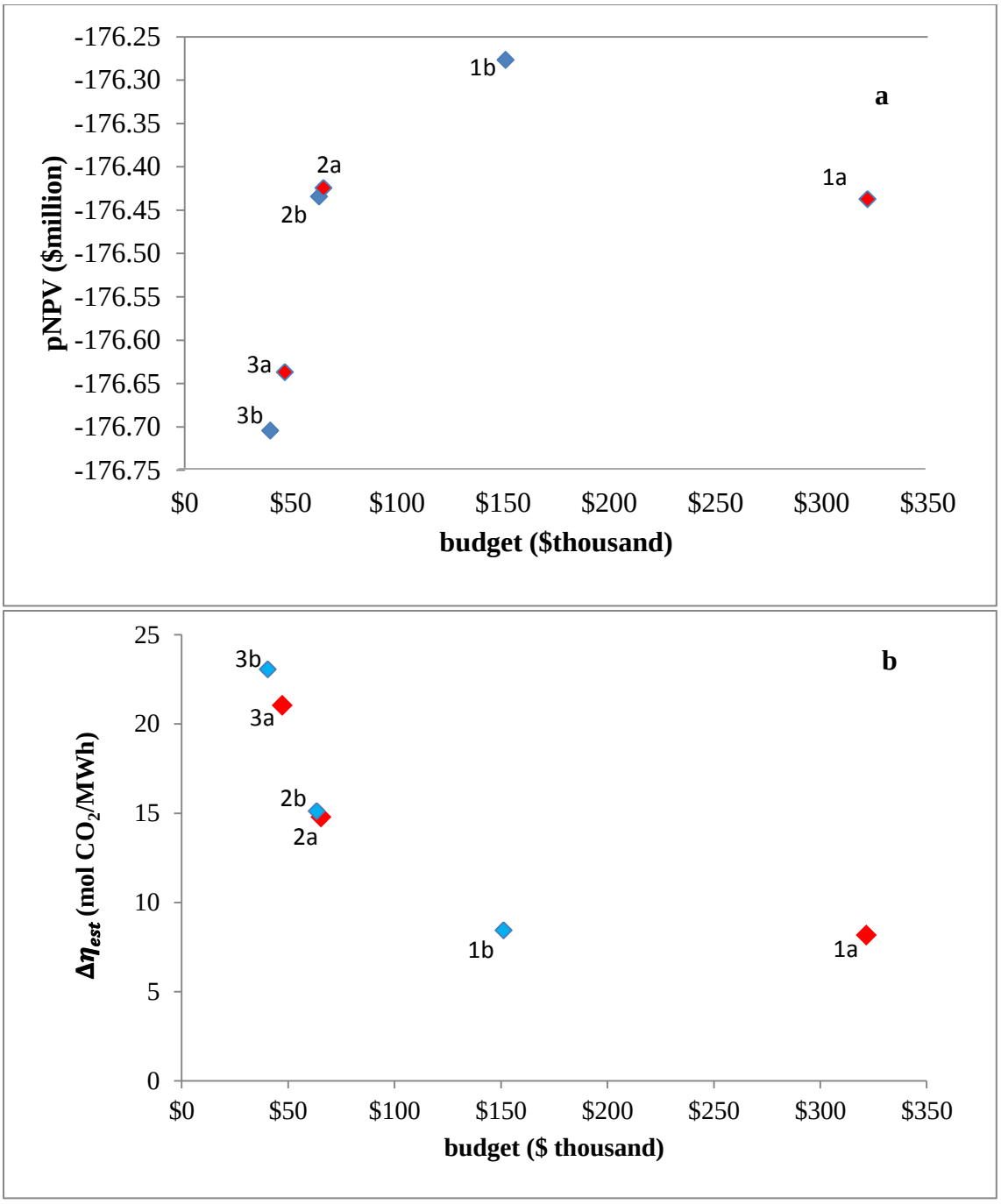

Figure 6.11: (a) pNPV vs. budget, (b) deviation in efficiency vs. budget.

Figure 6.11(a) shows the change in pNPV ($ million) with the increase in budget ($ thousand). Figure 6.11(b) shows the deviation in efficiency with increase in budget. It is observed that the pNPV has the least negative value at a sensor budget of $150,200 even though the CO_2 capture efficiency is slightly lesser than that achieved at a sensor budget of $321,600. Thus the final

optimal senor set is the one that is obtained at a sensor budget of $150,200. The pNPV analysis is therefore a nice tool to evaluate the trade-off between the cost of sensors and process efficiency.

Table 6.5 : Optimal Set of Sensors at $151,200

Temperature
1. Syngas Cooler Inlet 2. Clean Syngas at the Top of CO_2 Absorber 3. Stripped Gas Cooler Outlet 4. Rich Solvent at H_2S Absorber Bottom 5. Rich Solvent at Inlet to H_2S Concentrator 6. N_2 Gas Flow to H_2S Concentrator 7. H_2S Concentrator Vapor Outlet 8. Stripped Solvent at the Bottom of Selexol Stripper
Pressure
9. Top Outlet of H_2O K.O. Drum 10. Loaded Solvent at the Bottom of CO_2 Absorber 11. Rich Solvent at Inlet to H_2S Concentrator 12. Stripped Solvent at the Bottom of Selexol Stripper 13. Valve Outlet at the Bottom of Acid Gas K.O. Drum 14. Selexol Stripper Top Outlet
Flow
15. Inlet to H_2O K.O. Drum 16. Clean Syngas at the Top of CO_2 Absorber 17. Off Gas from Top of H_2S Absorber 18. H_2S Concentrator Vapor Outlet 19. Stripped Solvent at the Bottom of Selexol Stripper
H_2S Analyzer
20. H_2S Concentrator Vapor Outlet
CO_2 Analyzer
21. H_2O K.O. Drum Bottom Outlet 22. Off Gas from Top of H_2S Absorber 23. H_2S Concentrator Vapor Outlet 24. Top Outlet of Acid Gas K.O

It should be noted that the solution to the NDMSND problem is not necessarily unique. Furthermore, the optimal set of sensors can change depending on the disturbances and set point changes and their magnitudes and characteristics. The developed algorithm is generic and can be readily used to study the impact of such changes. Table 4.4 shows that list of optimal set of sensors at $151,200.

6.6 Conclusion

In this work, a NDMSND algorithm has been developed for efficiency maximization of an estimator-based control system. UKF is used to estimate primary controlled variables in the presence of a nonlinear process model and noisy measurements. The nonlinear model is obtained by the LIP method. The NDMSND algorithm solves a lexicographic optimization where CO_2 capture efficiency is maximized followed by budget minimization. The algorithm is applied to a large-scale acid gas removal unit as part of an IGCC power plant with CO_2 capture. It is observed that lexicographic optimization helps to achieve almost the same efficiency even at lower budgets for each case study. The pNPV analysis helps to identify the final optimal set of sensors by evaluating the trade-off between the cost of sensors and process efficiency of all sets of sensors obtained by multi-objective optimization.

Chapter 7

Suggestions for Future Work

The SND algorithm can be developed to optimize different performance criteria to achieve efficient plant operation. A systematic design of the SND algorithm is crucial for any process plant. But the challenges usually encountered while developing SND algorithm are the computational expense and the limitation of available resources.

Computational expense has always been an issue while solving large scale problems especially by using GA. Alternative approaches such as tree search method should be investigated for developing SND algorithm for efficiency maximization. A systematic approach is needed for determining the variable step size. Variable step size for solving dynamic model-based SND algorithm for reducing computational expense should be investigated and included in the SND algorithm.

Limitations of the available resources further increase computational expense. The GA available in MATLAB has some restrictions in solving integer problems (IP/MINLP). Development of a GA code that can handle linear and nonlinear equality constraints is encouraged. This would result in more efficient implementation of the SND algorithm in the GA framework. In this work, the developed algorithm has been parallelized by using a distributed computing server and implemented in a remote cluster computer. It is expected that the computation time will decrease with the increase in number of workers in parallelization. But an anomaly is observed with the use of the GA available in MATLAB. This is an area of further investigation for efficient implementation of SND algorithm while using GA to find optimal set of sensors.

The SSND algorithm developed in this work, considers P-only controller for the primary control variables. However, estimator-based control action can be improved by using PI/PID control laws for those variables. The approaches used for reducing computational expense in the

DMSND algorithm, including reduced order model, and using incumbent solution can also be utilized in the SSND algorithm to achieve the advantage of computational efficiency.

The DMSND algorithm in this work considers traditional KF for estimating key variables for process control and monitoring. In traditional KF, the covariance matrices Q and R remains fixed. Thus good guesses for Q and R are required to obtain satisfactory filter performance. However, in real-life applications, these matrices are unknown and it is difficult to generate good guesses for them. To address this issue, the use of an adaptive KF can be investigated where Q and R are updated at every time instant based on the estimation error in the previous time instants.

The NDMSND algorithm could be further extended for very large scale problems involving thousands of nonlinear equations. The computational expense could be decreased by using an incumbent solution approach. Variable step size could also be considered for efficient implementation of the algorithm. The NDMSND algorithm is developed assuming each candidate sensor set being evaluated from one steady-state to new steady-state following any disturbance/setpoint change. It results in an inefficient search for the optimal sensor sets.

References

Ali Y. Sensor network design for maximizing reliability of chemical processes. *Ph.D. Thesis, Indian Institute of Technology Kanpur*, India, 1993.

Ali Y, Narasimhan S. Sensor network design for maximizing reliability of linear processes. *AIChE Journal*. 1993; 39:820–828.

Ali Y, Narasimhan S. Redundant sensor network design for linear processes. *AIChE Journal*. 1995; 41(10):2237-2249.

Ali Y, Narasimhan S. Sensor network design for maximizing reliability of bilinear processes, *AIChE Journal*. 1996;42(9):2563-2575.

Alonso AA, Kevrekidis IG, Banga JR, Frouzakis CE. Optimal sensor location and reduced order observer design for distributed process system. *Computer & Chemical Engineering*. 2004.

Anderson BDO, Moore JB, Optimal Filtering, *Prentice-Hall, Englewood Cliffs*, New Jersy, USA. 1979.

Antoulas AC. Approximation of linear dynamical systems, *Wiley Encyclopedia of Electrical and Electronics Engineering*. Edited by J.G. Webster. 1999;11:403-422.

Antoulas AC, Sorensen DC. Approximation of large-scale dynamical systems: An overview. *International Journal of Applied Mathematics and Computer Science.* 2001;11(5):1093–1121.

Bagajewicz M. Design and retrofit of sensor networks in process plants. *AIChE Journal* 1997; 43:2300.

Bagajewicz M. Process Plant Instrumentation: Design and Upgrade. *Boca Raton, FL: CRC press*, 2000.

Bagajewicz M. On a new definition of a stochastic-based accuracy concept of data reconciliation-based estimators. *Proceedings of the 15th European symposium on computer-aided process engineering*. 2005a.

Bagajewicz M. On the definition of software accuracy in redundant measurement systems. *AIChE Journa.l* 2005b;51(4):1201–1206.

Bagajewicz M. Value of accuracy in linear systems. *AIChE Journa.l* 2006; 52(2):638–650.

Bagajewicz M, Cabrera E. New MILP formulation for instrumentation network design and upgrade. *AIChE Journal.* 2002;48:2271-2228.

Bagajewicz M, Markowski M. Instrumentation design and upgrade using an unconstrained method with pure economical objectives. *Foundations of Computer Aided Process Operations*. Florida: Coral Springs; 2003.

Bagajewicz M, Markowski M, Budek A. Economic value of precision in the monitoring of linear systems. *AIChE Journal.* 2005;51(4):1304–1309.

Bagajewicz M, Nguyen D. Stochastic-based accuracy of data reconciliation estimators for linear systems. *Computer & Chemical Engineering.* 2008; 32(6):1257-1269.

Bagajewicz M, Sanchez M. Cost-optimal design of reliable sensor networks. *Computers & Chemical Engineering.* 2000;23(11-12):1757-1762.

Behringer FA. Lexicographic quasiconcave multiobjective programming. *Zeitschrift fier operation research.*1977; 21:103-116.

Bhattacharyya D, Turton R, Zitney SE. Steady-state simulation and optimization of an integrated gasification combined cycle (IGCC) power plant with CO_2 capture. *Industrial & Engineering Chemistry Research*. 2011; 50:1674-1690.

Bhushan M, Rengaswamy R. Design of sensor network based on the signed directed graph of the process for efficient fault diagnosis. *Ind Eng Chem Res*. 2000;39:999–1019.

Bhushan M, Rengaswamy R. Comprehensive design of a sensor network for chemical plants based on various diagnosability and reliability criteria. 1. Framework. *Ind Eng Chem Res*. 2002a;41(7):1826-1839.

Bhushan M, Rengaswamy R. Comprehensive design of a sensor network for chemical plants based on various diagnosability and reliability criteria. 2. Applications. *Ind Eng Chem Res.* 2002b;41:1840−1860.

Bhushan M, Narasimhan S, Rengaswamy R. Robust sensor network design for fault diagnosis. *Comput Chem Eng*. 2008;32:1067–1084.

Carnero M, Hernandez J, Sanchez M, Bandoni A. An evolutionary approach for the design of nonredundant sensor networks *Industrial & Engineering Chemistry Research.* 2001; 40:5578–5584.

Carnero M, Hernandez J, Sanchez M, Bandoni A. On the solution of the instrumentation selection problem. *Industrial & Engineering Chemistry Research.* 2005; 44:358–367.

Chmielewski DJ, Palmer T, Manousiouthakis V. On the theory of optimal sensor placement. *AIChE Journal* 2002; 48:1001–1012.

Cohen GH, Coon GA. Theoretical considerations of retarded control. *Trans ASME* 1953; 75:827-834.

Ding W, Wang J and Rizos C. Improving Adaptive Kalman Estimation in GPS/INS Integration. *The Journal of Navigation*. 2007;60:517-529.

Dunik M, Simandl J. Sigma point Gaussian Sum Filter design using square root unscented filters, *Proceedings of the 16th IFAC World Congress*. 2005;16:1000–1005.

Figueroa JD, Fout T, Plasynski S, McIlvried H, Srivastava R. Advances in CO_2 capture technology-The U.S Department of Energy's carbon sequestration program. *International J Greenhouse Gas Control.* 2008;2:9–20.

Gelb A. Applied Optimal Estimation, *MIT Press*, Cambridge, MA. 1974.

Georges D. The use of observability and controllability gramians or functions for optimal sensor and actuator location in finite-dimensional systems. *Proceedings of the 34th Conference on Decision and Control*. New Orleans, LA, 1995:3319.

Ghosh K, Ramteke M, Srinivasan R. Optimal variables selection or effective statistical process monitoring. *Computer & Chemical Engineering.* 2014; 60:260-276.

Glover K. All optimal $Hankel\ \infty$-norm approximations of linear multivariable systems and their L^{∞}- error bounds. *International Journal of Control*. 1984; 39:1115–1193.

Grewal MS, Andrews AP. Kalman Filtering Theory and Practice. *Upper Saddle River*, *Prentice Hall.* New Jersey, USA, 1993.

Haupt RL, Haupt SE. Practical genetic algorithms. 2nd ed. *Wiley-Interscience,* New Jersey. 2004.

Hermann R, Kerner AJ, Nonlinear controllability and observability. *IEEE Trans. Autom. Control* 1977;22:728.

Holsapple R, Iyer R, Doman D. Variable step-size selection methods for implicit integration schemes for ODEs. *International Journal of Numerical Analysis and Modeling*. 2007;4(2):212-242.

Isidori A. Nonlinear Control Systems, 3rd ed.; *Springer-Verlag*: New York,USA. 1995.

Jacobs OLR. Introduction to Control Theory, 2nd Edition. *Oxford University Press*. 1993.

Jones D, Bhattacharyya D, Turton R, and Zitney SE. Plant-Wide Control System Design: Primary Controlled Variable Selection. *Computer & Chemical Engineering.* 2014:71;220-234.

Julier S, Uhlmann J, A New Extension of the Kalman Filter to Nonlinear System. *Proceedings of SPIE: The International Society for Optical Engineering*. 1997; 3068:182–193.

Julier S, Uhlmann J, Unscented filtering and nonlinear estimation, *Proc. IEEE.* 2004;92:401–422.

Kadu SC, Bhushan M, Gudi R. Optimal sensor network design for multirate systems. *Journal of Process Control*. 2008;18(6):594-604.

Kalman RE, Bucy RS. New results in linear filtering and prediction theory. *Journal of Basic Engineering.* 1961;83(3):95-108.

Karim S, Reza BM, Mohsen M. Design of Instrumentation Sensor Networks for Non-Linear Dynamic Processes Using Extended Kalman Filter, J. Chem. Chem. Eng, 2008; 27(3):11-23.

Kelly JD, Zyngier D. A new and improved MILP formulation to optimize observability, redundancy and precision for sensor network problems. *AIChE Journal.* 2008; 54:1282–1291.

Kottakki KK, Bhartiya S, Bhushan M. State estimation of nonlinear dynamical systems using nonlinear update based Unscented Gaussian Sum Filter. *Journal of Process Control*. 2014,24:1425-1443.

Kretsovalis A, Mah RS. Effect of redundancy on estimation accuracy in process data reconciliation. *Chemical Engineering Science*. 1987;42:2115−2121.

Lee AJ, Diwekar UM. Optimal sensor placement in integrated gasification combined cycle power systems. *Applied Energy.* 2012;99:255–64.

Lewis R. Optimal Estimation with an Introduction to Stochastic Control Theory. *John Wiley & Sons, Inc*. 1986.

Liptak BG. Process measurement and Analysis. 4th ed. Boca Raton, Florida: CRC press; 2003.

Lopez T, Alvarez J. On the effect of the estimation structure in the functioning of a nonlinear copolymer reactor estimator. *Journal of Process Control.* 2004;14:99.

Luong M, Maquin D, Huynh C, Ragot J. Observability, redundancy, reliability and integrated design of measurement systems. *Proceeding of 2nd IFAC Symposium on Intelligent Components and Instrument Control Applications*, Budapest, Hungary, 1994.

Madron F, Veverka V. Optimal selection of measuring points in complex plants by linear models. *AIChE Journal.* 1992;38:227–236.

Maybeck PS. Stochastic Models, Estimation, and Control, Volume 1. *Academic Press, Inc*. 1979.

Mehra R. On the Identification of Variance and Adaptive Kalman Filtering. *IEEE Trans Automat Contr*, 1970;15(2):175-184.

Mehra R. 1971. On-Line Identification of Linear Dynamic Systems with Applications to Kalman Filtering. *IEEE. Trans Automat Contr*. 1971;16(1): 12-21.

Mellefont DJ, Sargent RWH. Selection of measurements for optimal feedback control. *Industrial Engineering Chemistry*. *Process Design and Development*.1978; 17(4):549–552.

Meyer D. Fractional balanced reduction: Model reduction via a fractional representation. *IEEE Trans Autom Control.* 1990; 35(12):1341–1345.

Meyer M, Le Lann J, Koehret B, Enjalbert M. Optimal selection of sensor location on a complex plant using a graph oriented approach. *Computer & Chemical Engineering*. 1994;18:535–540.

Miettinen KM. Nonlinear multiobjective optimization. *Kluwer Academic Publishers*. Massachusetts, USA.2002.

Mohamed A. and Schwarz K. Adaptive Kalman Filtering for INS/GPS. *Journal of Geodesy*. 1999;73:193-203.

Musulin E, Bagajewicz M, Nougues JM, Puigjaner L. Instrumentation design and upgrade for principal components analysis monitoring. Ind Eng Chem Res. 2004;43:2150–2159.

Nabil M, Narasimhan S. Sensor network design for optimal process operation based on data reconciliation. *Industrial & Engineering Chemistry Research.* 2012; 51:6789-6797.

Narasimhan S, Rengaswamy R. Quantification of performance of sensor networks for fault diagnosis. AIChE J. 2007;53:902–917.

Narendra KS. Adaptive Control Using Neural Networks and Approximate Models. IEEE Transaction on Neural Networks. 1997;8(3):475–485.

Nguyen D, Bagajewicz M. Design of nonlinear sensor networks for process plants. . *Industrial & Engineering Chemistry Research.* 2008; 47:5529-5542.

Nguyen D, Bagajewicz M. New sensor network design and retrofit method based on value of information. *AIChE Journal.* 2011;57:2136–2148.

Nguyen D, Bagajewicz M. Efficient approximate methods for the design and upgrade of sensor networks. *Industrial & Engineering Chemistry Research.* 2013;52:83-90.

Ogunnaike BA, Ray WH. Process dynamics, modeling and control. *Oxford University press.* New York, USA; 1994.

Paul P, Bhattacharyya D, Turton R, Zitney SE. Sensor network design for maximizing process efficiency: an algorithm and its application. *AIChE Journal*. 2015; 61(2):464-476.

Paul P, Bhattacharyya D, Turton R, Zitney S E. Adaptive Kalman filter for estimation of environmental performance variables in an acid gas removal process. *Proceedings of the 2013 American Control Conference.* 2013;2717-2721.

Peng JK, Chmielewski DJ. Covariance based hardware selection part 2: simultaneous sensor and actuator selection. *IEEE Transactions on Control Systems Technology.* 2006;14(2):362–368.

Peng JK, Chmielewski DJ. Optimal sensor network design using the minimally backed-off operating point notion of profit. *Proceeding of American Control Conference*. 2005; 220−224.

Raghuraj R, Bhushan M, Rengaswamy R. Locating sensors in complex chemical plants based on fault diagnostic observability criteria. AIChE J. 1999;45:310–322.

Safonov MG, Chiang RY, *Limebeer* DJN. Optimal *Hankel* model reduction for non-minimal system. *IEEE*. 1990; 35(4):496-502.

Sahraei MH, McCalden D, Hughes R, Ricardez-Sandoval LA, A survey on current advanced IGCC power plant technologies, sensors and control system. *Fuel*. 2014;137:245-259.

Scherpen JMA. Balancing of nonlinear systems. *Syst. Control Lett*. 1993;21:143.

Sen S, Narasimhan S, Deb K. Sensor network design of linear processes using genetic algorithms. *Computer & Chemical Engineering*. 1998;22:385–390.

Sentoni GB, Biegler LT, Guiver JB, and Zhao H. State-Space Nonlinear Process Modeling: Identification and Universality. AIChE Journal. 1998;44 (10).

Serpas MR, Soft Sensors for Process Monitoring of Complex Processes. *Ph.D. Thesis, Texas A&M*, Texas, 2012.

Singh AK, Hahn J. Determining optimal sensor locations for state and parameter estimation for stable nonlinear system. *Industrial & Engineering Chemistry Research.* 2005; 44:5645-5659.

Singh AK, Hahn J. Sensor location for stable nonlinear dynamic system: Multiple sensors case. *Industrial & Engineering Chemistry Research.* 2006;45:3615-3623.

Skogestad S. Plantwide control: The search for the self-optimizing control structure. *Journal of Process Control.* 2000;10(5):487-507.

Skogestad S. Control structure design for complete chemical plants. *Computer & Chemical Engineering.* 2004;28:219-234.

Soderstrom T, Discrete Time Stochastic Systems, *Springer*, New York, 2002.

Sorenson HW. Least-Squares estimation: from Gauss to Kalman. *IEEE Spectrum*, 1970;7:63-68.

Sorenson H, Alspach D. Recursive Bayesian estimation using Gaussian sums, *Automatica.* 1971;7:465–479.

Straka O, Dunik J, Simandl M, Gaussian sum unscented Kalman Filter with adaptive scaling parameters. *Proceedings of the 14th International Conference on Information Fusion*. 2011:1–8.

Vaclavek V, Loucka M. Selection of Measurements Necessary to Achieve Multicomponent Mass Balances in Chemical Plant. *Chemical Engineering Science*. 1976;31: 1199.

Wan E, Merwe VD. The Unscented Kalman Filter, Chap.7 in Kalman Filtering and Neural Networks. *Wiley*. New York. 2002:221–282.

Wang J, Stewart M, Tsakiri M. Online Stochastic Modeling for INS/GPS Integration. *ION GPS-99 Proceedings*. Nashville, TN. 1999:1887–1895.

Wang J. Stochastic Modeling for Real-Time Kinematic GPS/GLONASS Position. *Navigation*. 2000;46(4):297-305.

Willcox K, Peraire J. Balanced model Reduction via the proper orthogonal decomposition. *Journal of AIAA.* 2002;40:2323-2330.

Wouwer AV, Point N, Porteman, S, Remy M. An approach to the selection of optimal sensor locations in distributed parameter systems. *Journal of Process Control.* 2000;10:291.

Zarchan P, Musoff H. Fundamentals of Kalman Filtering: A Practical Approach. *American Institute of Aeronautics and Astronautics, Inc*. Virginia, USA. 2009.

Ziegler JG, Nichols NB. Optimum setting for automatic controllers. *Trans ASME*. 1942;64:759-768.

Zumoffen D, Basualdo M. A systematic approach for the design of optimal monitoring systems for large scale processes. *Industrial & Engineering Chemistry Research.* 2010;49:1749-1761.

Appendix A

A.1 Integrated Gasification Combined Cycle (IGCC) Plant with CO_2 Capture

The IGCC power plant with CO_2 capture that is considered in this work is based upon the work of Bhattacharyya et al, 2011. A simplified block flow diagram of the process is shown in Figure A.1.

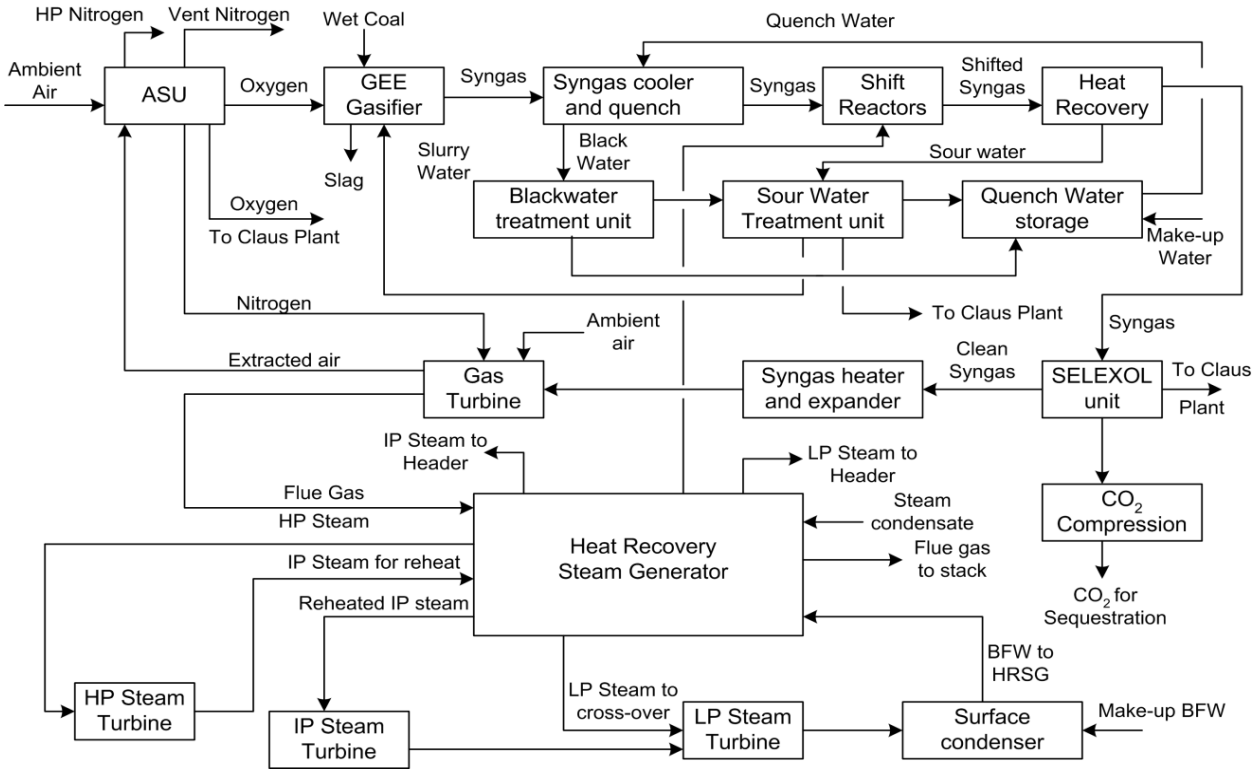

Figure A.1: Block Flow Diagram of IGCC with Carbon Capture (Bhattacharyya et al, 2011).

The coal is gasified to generate the raw syngas. This syngas mainly consists of carbon monoxide, carbon dioxide, hydrogen, water, hydrogen sulfide, and nitrogen. This raw syngas is then sent to a series of water-gas shift reactors. The shifted syngas is then sent to the acid gas

removal unit (AGR) where carbon dioxide and hydrogen sulfide are selectively removed from the syngas. The carbon dioxide is sent to a series of compressors where it is pressurized and sent for sequestration. The hydrogen sulfide-rich stream is sent to the Claus unit. The cleaned syngas is then sent to the gas turbine (GT) for power production. The hot tail gas from the GT is then sent to a heat recovery steam generation unit (HRSG) where it is used to raise three pressures of steam for additional power production.

A.2 Optimal Sensor Sets from SSND

Tables A1-A6 present the list of optimal sensors at different budgets obtained from SSND algorithm.

Table A.1: Optimal Set of Sensors at $322,600

Temperature Sensor
1. [a]H_2S $Absorber_8$
2. [a]H_2S $Absorber_{14}$
3. [a]H_2S $Absorber_{26}$
4. [a]CO_2 $Absorber_2$
5. [a]Selexol $Stripper_1$
6. Inlet to H_2O K.O. Drum
7. Off Gas from Top of H_2S Absorber
8. Loaded Solvent at the Bottom of CO_2 Absorber
9. Rich Solvent at H_2S Absorber Bottom
10. H_2 Recovery Flash Vessel Outlet
11. Semi-lean Solvent Cooler Inlet
12. Semi-lean Solvent Cooler Outlet
13. H_2S Concentrator Vapor Outlet
14. Stripped Gas Compressor Outlet
15. Acid Gas K.O. Drum Liquid Outlet
16. H_2 Recovery Compressor Outlet

17. 1st LP CO_2 Compressor Inlet
18. 2nd LP CO_2 Compressor Outlet
19. 3rd LP CO_2 Compressor Outlet
20. 4th LP CO_2 Compressor Inlet
21. 5th LP CO_2 Compressor Inlet
22. 1st HP CO_2 Compressor Outlet
23. Vapor Outlet of MP CO_2 Flash for H_2O Removal
24. 3rd MP CO_2 Compressor Inlet
25. 3rd MP CO_2 Compressor Outlet
26. Liquid of CO_2 Flash Vessel
27. Tail Gas to H_2S Absorber

Pressure Sensor

28. [a]Selexol $Stripper_1$
29. [a]Selexol $Stripper_3$
30. Syngas Cooler Inlet
31. Clean Syngas at the Top of CO_2 Absorber
32. Semi-lean Solvent Cooler Inlet
33. Rich Solvent at Selexol Stripper Inlet
34. Stripper Solvent at Lean/Rich H.E. Inlet
35. Inlet to H_2 Recovery Flash Vessel
36. MP Flash Vessel Inlet
37. LP Flash Vessel Inlet
38. Recycle Pump Outlet
39. Recycle Stream at CO_2 Absorber Inlet
40. Stripped Gas Compressor Outlet
41. Acid Gas K.O. Drum Liquid Outlet
42. HP Flash Vessel Outlet
43. Valve Outlet at the Top of MP Flash Vessel
44. Outlet of 1st LP CO_2 Compressor
45. Outlet of 2nd MP CO_2 Compressor

46. 1st HP CO_2 Compressor Outlet
47. 2st HP CO_2 Compressor Intlet
Flow Sensor
48. Syngas Inlet
49. Rich Solvent at H_2S Absorber Bottom
50. Stripped Solvent at the Bottom of Selexol Stripper
51. HP Flash Vessel Intlet
52. H_2 Recovery Flash Vessel Outlet
53. HP Flash Vessel Outlet
54. LP Flash Vessel Bottom
55. Steam to the Selexol Stripper
56. H_2O K.O. Drum Bottom Outlet
CO_2 Analyzer
57. [a]Liquid Phase in H_2S $Absorber_5$
58. [a]Liquid Phase in H_2S $Concentrator_5$
59. [a] Liquid Phase in CO_2 $Absorber_8$
60. [a] Liquid Phase in CO_2 $Absorber_{14}$
61. H_2S Absorber Bottom
62. H_2S Concentrator Liquid Outlet
63. Stripped Solvent at the Bottom of Selexol Stripper
64. Valve Outlet at the Bottom of Acid Gas K.O. Drum
65. H_2 Recovery Top Outlet
H_2S Analyzer
66. [a]H_2S $Absorber_{16}$

Table A.2: Optimal Set of Sensors at $229,400

Temperature Sensor
1. [a]H_2S $Absorber_2$
2. [a]H_2S $Concentrator_3$
3. [a]CO_2 $Absorber_8$
4. [a]$Selexol\ Stripper_3$
5. Off Gas at the Inlet to CO_2 Absorber
6. Clean Syngas at the Top of CO_2 Absorber
7. Loaded Solvent Cooler Outlet
8. Rich Solvent at H_2S Absorber Bottom
9. Stripped Solvent at the Bottom of Selexol Stripper
10. Solvent Cooler Outlet
11. H_2 Recovery Flash Vessel Outlet
12. HP Flash Vessel Outlet
13. MP Flash Vessel Outlet
14. Stripped Gas Cooler Outlet
15. Steam to the Selexol Stripper
16. H_2 Recovery Compressor Outlet
17. H_2 Recovery Cooler Outlet
18. HP Flash Vessel Vapor K.O. Drum Outlet
19. 1^{st} LP CO_2 Compressor Outlet
20. 2^{nd} LP CO_2 Compressor Outlet
21. 4^{th} LP CO_2 Compressor Inlet
22. 5^{th} LP CO_2 Compressor Outlet
23. 1^{st} MP CO_2 Compressor Inlet
24. 2^{nd} MP CO_2 Compressor Inlet
25. 2^{nd} MP CO_2 Compressor Outlet
26. 1^{st} HP CO_2 Compressor Outlet
27. Liquid of CO_2 Flash Vessel

Pressure Sensor
28. [a]H_2S Absorber$_{16}$
29. Acid Gas K.O. Drum Vapor Outlet
30. [a]CO_2 Absorber$_3$
31. [a]CO_2 Absorber$_{15}$
32. [a]Selexol Stripper$_3$
33. Semi-lean Solvent Cooler Inlet
34. Lean Solvent at the inlet to CO_2 Absorber
35. Inlet to H_2 Recovery Flash Vessel
36. H_2 Recovery Flash Vessel
37. MP Flash Vessel Inlet
38. Stripped Gas Compressor Outlet
39. Valve Outlet at the Bottom of Acid Gas K.O. Drum
40. H_2 Recovery Top Outlet
41. H_2 Recovery Compressor Outlet
42. Valve Outlet at the Top of HP Flash Vessel
43. Valve Outlet at the Top of LP Flash Vessel
44. Outlet of 2nd MP CO_2 Compressor
45. Glycol Absorber Inlet
46. 2st HP CO_2 Compressor Intlet
47. Tail Gas to H_2S Absorber
Flow Sensor
48. Rich Solvent at H_2S Absorber Bottom
49. HP Flash Vessel Intlet
50. LP Flash Vessel Bottom
51. H_2S Concentrator Vapor Outlet
52. Selexol Stripper Top Outlet
53. Steam to the Selexol Stripper
54. MP Flash Vessel Top Outlet
55. H_2O K.O. Drum Bottom Outlet

56. [a]Liquid Phase in H_2S Absorber$_5$
CO_2 Analyzer
57. [a] Liquid Phase in CO_2 Absorber$_8$
58. Loaded Solvent at the Bottom of CO_2 Absorber
59. H_2S Absorber Bottom
60. HP Flash Vessel Bottom Outlet
61. LP Flash Vessel Top
62. H_2S Concentrator Vapor Outlet
63. Selexol Stripper Top Outlet
64. Glycol Absorber Top Outlet

Table A.3: Optimal Set of Sensors at $187,900

Temperature Sensor
1. [a]H_2S Absorber$_{14}$
2. [a]H_2S Absorber$_{20}$
3. [a]H_2S Absorber$_{26}$
4. Acid Gas K.O. Drum Vapor Outlet
5. [a]CO_2 Absorber$_2$
6. [a]CO_2 Absorber$_{14}$
7. [a]Selexol Stripper$_3$
8. [a]Selexol Stripper$_{10}$
9. Off Gas Cooler Outlet Temperature
10. Clean Syngas at the Top of CO_2 Absorber
11. Loaded Solvent Cooler Outlet
12. Loaded Solvent Chiller Outlet
13. Rich Solvent at H_2S Absorber Bottom
14. Rich Solvent Heater Inlet
15. Solvent Cooler Outlet
16. MP Flash Vessel Outlet

17. Semi-lean Solvent Cooler Outlet
18. Stripped Gas Cooler Outlet
19. H_2 Recovery Top Outlet
20. H_2 Recovery Cooler Outlet
21. HP Flash Vessel Vapor K.O. Drum Outlet
22. 1st LP CO_2 Compressor Inlet
23. 2nd LP CO_2 Compressor Outlet
24. 3rd LP CO_2 Compressor Outlet
25. 5th LP CO_2 Compressor Inlet
26. 5th LP CO_2 Compressor Outlet
27. 1st MP CO_2 Compressor Inlet
28. 2nd MP CO_2 Compressor Inlet
29. Vapor Outlet of MP CO_2 Flash for H_2O Removal
30. 3rd MP CO_2 Compressor Inlet
31. 1st HP CO_2 Compressor Outlet
32. 2nd HP CO_2 Compressor Inlet
33. Tail Gas to H_2S Absorber

Pressure Sensor

34. [a]H_2S Concentrator$_4$
35. Acid Gas K.O. Drum Vapor Outlet
36. [a]CO_2 Absorber$_9$
37. [a]Selexol Stripper$_3$
38. Semi-lean Solvent Cooler Inlet
39. H_2 Recovery Flash Vessel
40. LP Flash Vessel Inlet
41. Valve Outlet at the Bottom of Acid Gas K.O. Drum
42. H_2 Recovery Top Outlet
43. H_2 Recovery Compressor Outlet
44. Valve Outlet at the Top of MP Flash Vessel
45. Valve Outlet at the Top of LP Flash Vessel
46. Outlet of 2nd MP CO_2 Compressor

47. Glycol Absorber Top Outlet
48. 1st HP CO_2 Compressor Outlet
Flow Sensor
49. Off Gas from Top of H_2S Absorber
50. H_2S Concentrator Liquid Outlet
51. MP Flash Vessel Top Outlet
52. N_2 Gas Flow to H2S Concentrator
53. H_2O K.O. Drum Bottom Outlet
CO_2 Analyzer
54. [a]Liquid Phase in H_2S $Absorber_{16}$
55. Off Gas from Top of H_2S Absorber
56. [a] Liquid Phase in CO_2 $Absorber_{14}$
57. Stripped Solvent at the Bottom of Selexol Stripper
58. HP Flash Vessel Bottom Outlet
59. LP Flash Vessel Top
60. Valve Outlet at the Bottom of Acid Gas K.O. Drum
61. Selexol Stripper Top Outlet
62. Glycol Absorber Top Outlet

Table A.4: Optimal Set of Sensors at $118,700

Temperature Sensor
1. [a]H_2S $Concentrator_5$
2. Acid Gas K.O. Drum Vapor Outlet
3. [a]CO_2 $Absorber_8$
4. [a]CO_2 $Absorber_{14}$
5. [a]Selexol $Stripper_7$
6. Off Gas at the Inlet to CO_2 Absorber
7. Rich Solvent at H_2S Absorber Bottom

8. H_2S Concentrator Liquid Outlet
9. Stripper Solvent at Lean/Rich H.E. Inlet
10. Semi-lean Solvent Cooler Inlet
11. Semi-lean Solvent Cooler Outlet
12. H_2S Concentrator Vapor Outlet
13. Acid Gas K.O. Drum Liquid Outlet
14. H_2 Recovery Compressor Outlet
15. HP Flash Vessel Vapor K.O. Drum Outlet
16. 1^{st} LP CO_2 Compressor Inlet
17. 2^{nd} LP CO_2 Compressor Outlet
18. 3^{rd} LP CO_2 Compressor Outlet
19. 2^{nd} MP CO_2 Compressor Inlet
20. 2^{nd} MP CO_2 Compressor Outlet
21. Glycol Absorber Top Outlet
22. 3^{rd} MP CO_2 Compressor Outlet
23. Liquid of CO_2 Flash Vessel

Pressure Sensor

24. [a]H_2S $Absorber_{16}$
25. [a]H_2S $Absorber_{25}$
26. [a]$Selexol\ Stripper_1$
27. Off Gas from Top of H_2S Absorber
28. Clean Syngas at the Top of CO_2 Absorber
29. Stripper Solvent at Lean/Rich H.E. Inlet
30. Lean Selexol Pump Outlet
31. Lean Solvent at the inlet to CO_2 Absorber
32. H_2 Recovery Flash Vessel
33. MP Flash Vessel Inlet
34. LP Flash Vessel Inlet
35. Stripped Gas Compressor Outlet
36. H_2 Recovery Top Outlet

37. Valve Outlet at the Top of HP Flash Vessel
38. Valve Outlet at the Top of MP Flash Vessel
39. Outlet of 2nd MP CO_2 Compressor
40. Tail Gas to H_2S Absorber
Flow Sensor
41. Syngas Inlet
42. Rich Solvent at H_2S Absorber Bottom
CO_2 Analyzer
43. Off Gas from Top of H_2S Absorber
44. [a] Liquid Phase in CO_2 $Absorber_8$
45. H_2S Concentrator Liquid Outlet
46. Selexol Stripper Top Outlet

Table A.5: Optimal Set of Sensors at $60,200

Temperature Sensor
1. [a]H_2S $Absorber_2$
2. Off Gas at the Inlet to CO_2 Absorber
3. Stripped Solvent at the Bottom of Selexol Stripper
4. Solvent Cooler Outlet
5. Stripped Gas Compressor Outlet
6. HP Flash Vessel Vapor K.O. Drum Outlet
7. 2nd LP CO_2 Compressor Outlet
8. 5th LP CO_2 Compressor Outlet
9. 2nd MP CO_2 Compressor Inlet
10. Vapor Outlet of MP CO_2 Flash for H_2O Removal
11. 2nd HP CO_2 Compressor Inlet

Pressure Sensor
12. [a]H_2S Absorber$_{25}$
13. [a]CO_2 Absorber$_3$
14. [a]Selexol Stripper$_3$
15. Rich Solvent at Selexol Stripper Inlet
16. H_2 Recovery Flash Vessel
17. H_2 Recovery Compressor Outlet
18. MP Flash Vessel Top Outlet
19. Tail Gas to H_2S Absorber
Flow Sensor
20. H_2S Concentrator Liquid Outlet
21. Selexol Stripper Top Outlet
22. Steam to the Selexol Stripper
CO_2 Analyzer
23. [a]Liquid Phase in H_2S Concentrator$_1$

Table A.6: Optimal Set of Sensors at $42,500

Temperature Sensor
1. [a]H_2S Absorber$_8$
2. Stripper Solvent at Lean/Rich H.E. Inlet
3. H_2 Recovery Compressor Outlet
4. HP Flash Vessel Vapor K.O. Drum Outlet
5. 5th LP CO_2 Compressor Outlet
6. 2nd MP CO_2 Compressor Inlet
7. 3rd MP CO_2 Compressor Outlet
8. Tail Gas to H_2S Absorber

Pressure Sensor
9. Syngas Cooler Inlet 10. Lean Selexol Pump Outlet 11. H_2 Recovery Flash Vessel 12. Acid Gas K.O. Drum Liquid Outlet 13. H_2 Recovery Top Outlet 14. Valve Outlet at the Top of MP Flash Vessel 15. 2^{st} HP CO_2 Compressor Inlet
Flow Sensor
16. Steam to the Selexol Stripper
CO_2 Analyzer
17. Glycol Absorber Top Outlet

Appendix B

Figure B.1 shows co-simulation of AGR process model in APD® and the estimator in MATLAB® Simulink. It mimics the real-life scenario for estimator-based control system. The measurements from nonlinear model in APD® are sent to MATLAB® where KF estimates the controlled variables. Then control action is calculated in MATLAB® based on the error (deviation from set point) and sent to APD® as control input. The syngas pressure, 55.06 bar at the inlet to H_2S absorber, is used as disturbance to the process. A step increase is introduced to inlet syngas to perturb the AGR process.

The estimator based control performance is further illustrated in Figure B.2. The comparison of H_2S capture efficiency profile between optimal process (no measurement noise) and estimator based control system have been presented. Even though the transient behavior of the estimator-based system deviates from the optimal profile, at steady state almost the same efficiency is obtained.

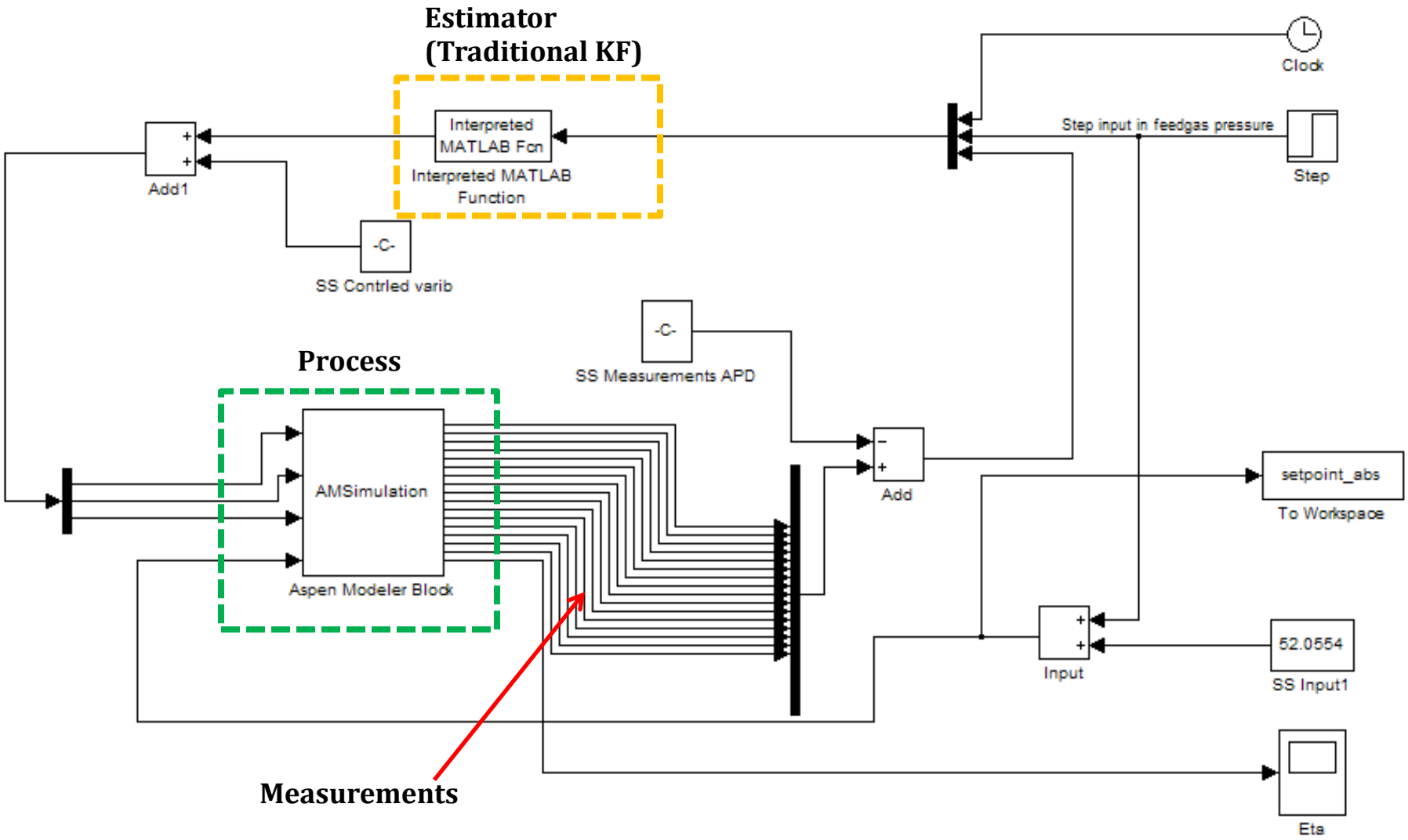

Figure B.1: Co-simulation of process model in APD® and Estimator in MATLAB®.

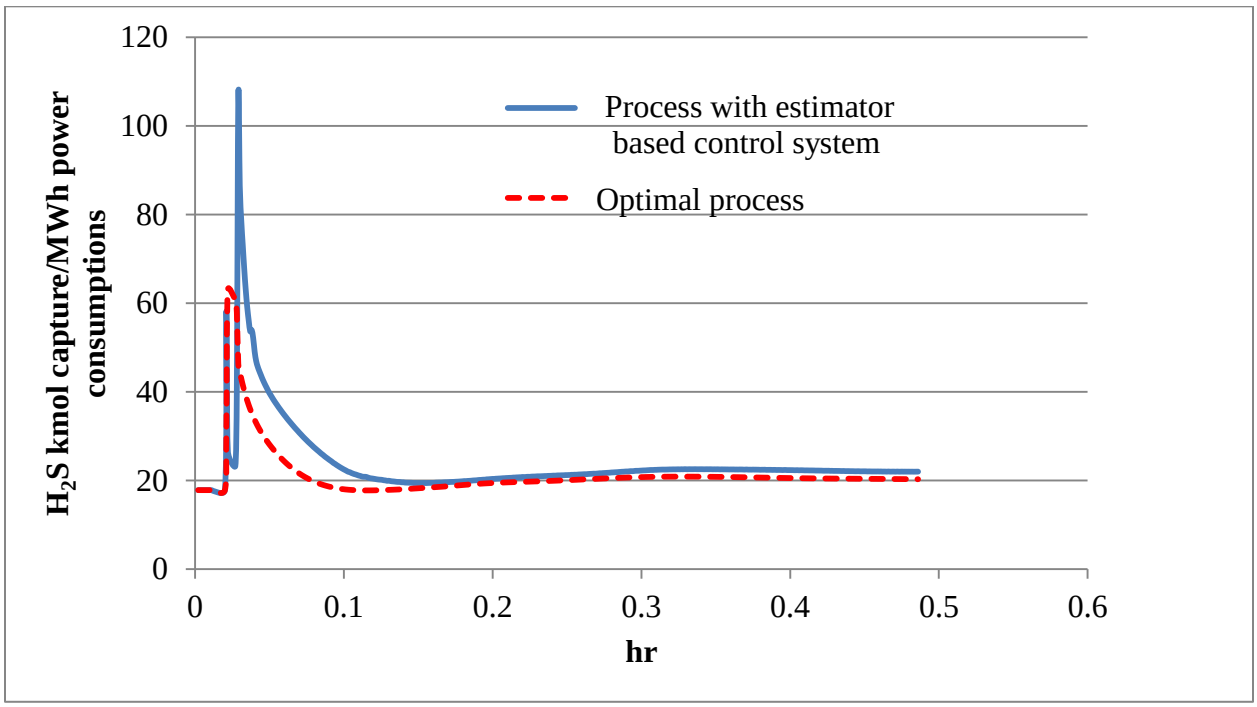

Figure B.2: Comparison of H_2S capture efficiency profile between optimal process (Nonlinear model and no measurement noise) and process with estimator based control system.

Appendix C

The optimal set of sensors at different budget obtained from NDMSND algorithm

Table C.1: Optimal Set of Sensors at $321,600

Temperature
1. Syngas Cooler Inlet 2. Inlet to H_2O K.O. Drum 3. Top Outlet of H_2O K.O. Drum 4. Clean Syngas at the Top of CO_2 Absorber 5. Off Gas from Top of H_2S Absorber 6. Rich Solvent at Inlet to H_2S Concentrator 7. Stripped Solvent at the Bottom of Selexol Stripper 8. Stripped Solvent at the Bottom of Selexol Stripper
Pressure
9. Inlet to H_2O K.O. Drum 10. Top Outlet of H_2O K.O. Drum 11. Clean Syngas at the Top of CO_2 Absorber 12. Stripped Solvent at the Bottom of Selexol Stripper 13. Top Outlet of Acid Gas K.O 14. Valve Outlet at the Bottom of Acid Gas K.O. Drum 15. Selexol Stripper Top Outlet 16. Stripped Solvent at the Bottom of Selexol Stripper
Flow
17. Inlet to H_2O K.O. Drum 18. Clean Syngas at the Top of CO_2 Absorber 19. H_2S Concentrator Vapor Outlet 20. Stripped Solvent at the Bottom of Selexol Stripper
H_2S Analyzer
21. H_2O K.O. Drum Bottom Outlet 22. Loaded Solvent at the Bottom of CO_2 Absorber 23. Rich Solvent at H_2S Absorber Bottom 24. Top Outlet of Acid Gas K.O

Table C.2: Optimal Set of Sensors at $65,400

Temperature
1. Syngas Cooler Inlet 2. Inlet to H_2O K.O. Drum 3. Clean Syngas at the Top of CO_2 Absorber 4. Stripped Gas Cooler Outlet 5. Off Gas from Top of H_2S Absorber 6. Rich Solvent at H_2S Absorber Bottom 7. Stripped Solvent at the Bottom of Selexol Stripper 8. Stripped Solvent at the Bottom of Selexol Stripper
Pressure
9. Inlet to H_2O K.O. Drum 10. Loaded Solvent at the Bottom of CO_2 Absorber 11. Rich Solvent at Inlet to H_2S Concentrator 12. Stripped Solvent at the Bottom of Selexol Stripper 13. Top Outlet of Acid Gas K.O 14. Valve Outlet at the Bottom of Acid Gas K.O. Drum 15. Selexol Stripper Top Outlet
Flow
16. Inlet to H2O K.O. Drum 17. Clean Syngas at the Top of CO_2 Absorber 18. Off Gas from Top of H_2S Absorber
CO_2 Analyzer
19. H_2O K.O. Drum Bottom Outlet 20. Off Gas from Top of H_2S Absorber 21. Selexol Stripper Top Outlet

Table C.3: Optimal Set of Sensors at $63,400

Temperature
1. Syngas Cooler Inlet
2. Top Outlet of H_2O K.O. Drum
3. Stripped Gas Cooler Outlet
4. Off Gas from Top of H_2S Absorber
5. Rich Solvent at H_2S Absorber Bottom
6. Rich Solvent at Inlet to H_2S Concentrator
7. N_2 Gas Flow to H_2S Concentrator
8. Stripped Solvent at the Bottom of Selexol Stripper
Pressure
9. Inlet to H_2O K.O. Drum
10. Top Outlet of H_2O K.O. Drum
11. Stripped Solvent at the Bottom of Selexol Stripper
12. Top Outlet of Acid Gas K.O
13. Valve Outlet at the Bottom of Acid Gas K.O. Drum
14. Selexol Stripper Top Outlet
15. Stripped Solvent at the Bottom of Selexol Stripper
Flow
16. Clean Syngas at the Top of CO_2 Absorber
17. Off Gas from Top of H_2S Absorber
18. H_2S Concentrator Vapor Outlet
19. Stripped Solvent at the Bottom of Selexol Stripper
20. Top Outlet of Acid Gas K.O
CO_2 Analyzer
21. H_2S Concentrator Vapor Outlet
22. Top Outlet of Acid Gas K.O

Table C.4: Optimal Set of Sensors at $47,200

Temperature
1. Syngas Cooler Inlet
2. Inlet to H_2O K.O. Drum
3. Top Outlet of H_2O K.O. Drum
4. Clean Syngas at the Top of CO_2 Absorber
5. Stripped Gas Cooler Outlet
6. Off Gas from Top of H_2S Absorber
7. N_2 Gas Flow to H_2S Concentrator
8. H_2S Concentrator Vapor Outlet
9. Stripped Solvent at the Bottom of Selexol Stripper
10. Stripped Solvent at the Bottom of Selexol Stripper
Pressure
11. Inlet to H_2O K.O. Drum
12. Stripped Solvent at the Bottom of Selexol Stripper
13. Top Outlet of Acid Gas K.O
14. Valve Outlet at the Bottom of Acid Gas K.O. Drum
15. Selexol Stripper Top Outlet
16. Stripped Solvent at the Bottom of Selexol Stripper
Flow
17. Inlet to H_2O K.O. Drum
18. Clean Syngas at the Top of CO_2 Absorber
19. Off Gas from Top of H_2S Absorber
20. H_2S Concentrator Vapor Outlet
21. Stripped Solvent at the Bottom of Selexol Stripper
22. Top Outlet of Acid Gas K.O

Table C.5: Optimal Set of Sensors at $40,400

Temperature
1. Syngas Cooler Inlet
2. Inlet to H_2O K.O. Drum
3. Clean Syngas at the Top of CO_2 Absorber
4. Off Gas from Top of H_2S Absorber
5. Rich Solvent at H_2S Absorber Bottom
6. Rich Solvent at Inlet to H_2S Concentrator
7. N_2 Gas Flow to H_2S Concentrator
8. H_2S Concentrator Vapor Outlet
9. Stripped Solvent at the Bottom of Selexol Stripper
Pressure
10. Inlet to H_2O K.O. Drum
11. Clean Syngas at the Top of CO_2 Absorber
12. Rich Solvent at Inlet to H_2S Concentrator
13. Stripped Solvent at the Bottom of Selexol Stripper
14. Top Outlet of Acid Gas K.O
15. Valve Outlet at the Bottom of Acid Gas K.O. Drum
16. Stripped Solvent at the Bottom of Selexol Stripper
Flow
17. Clean Syngas at the Top of CO_2 Absorber
18. Off Gas from Top of H_2S Absorber
19. H_2S Concentrator Vapor Outlet
20. Top Outlet of Acid Gas K.O