LEXICOGRAPHIC OPTIMIZATION BASED SENSOR NETWORK DESIGN FOR ROBUST FAULT DIAGNOSIS

Mani Bhushan * **Raghunathan Rengaswamy** **

* *Chemical and Materials Engineering Department, University of Alberta, Edmonton, AB, Canada*
** *Department of Chemical Engineering, Clarkson University, Potsdam, NY, USA*

Abstract: An appropriately designed sensor network is crucial for the success of any fault diagnostic strategy. In previous works (Bhushan and Rengaswamy, 2002a,b) strategies for optimally locating sensors based on criteria of reliability maximization and cost minimization have been devised. These opposing objectives were treated in a lexicographic (a special type of multiobjective optimization) manner. Signed digraph (SDG) based process models were used to generate the cause-effect information and fault occurrence and sensor failure probabilities were used to calculate a measure of reliability of a sensor network. While reliability and cost were considered in selecting the optimal sensor network, the robustness of the selected network with respect to uncertainties/errors in the underlying signed directed graph models and the available probability data was not considered explicitly. In this article, lexicographic formulations which incorporate some robustness enhancing criteria while designing cost-optimal sensor network for reliable fault diagnosis are presented. Some robustness to modeling errors in the SDG can be incorporated by choosing a distributed sensor network. Robustness to available probability data can be incorporated by maximizing reliability of the faults involving uncertain probability data. Integer linear programming (ILP) formulations incorporating these criteria in a lexicographic manner along with overall reliability-maximization and cost-minimization objectives are presented. The utility of the proposed approach is demonstrated through application to the Tennessee Eastman case study.

Keywords: Sensor, Fault Diagnosis, Multiobjective Optimization

1. INTRODUCTION

For safe and optimal operation of a chemical plant, it is essential to quickly detect and identify faults when they occur. Hence an efficient fault diagnosis methodology is very useful for modern day chemical plants. The efficiency of any diagnostic system depends critically on the location of the sensors monitoring important process variables. With hundreds of process variables available for measurement in a typical chemical plant, selection of crucial and optimum sensor positions is an important problem with possibly significant implications for various fault diagnosis strategies.

While the area of sensor location in general has been popular among researchers for quite sometime now, the area of sensor location from a fault diagnosis perspective has not received adequate attention. Even though related, there are significant differences in these two areas. The aim of sensor location problem as posed and solved in literature has been to identify variables to be measured such that various requirements on observability (ability to estimate values of all/important variables), estimation accuracy for key variables/parameters, and reliability (the probability of being able to estimate a variable) amongst others, are met, while the aim of sensor location from fault diagnostic perspective is to identify variables to be measured such that various faults when they occur can be detected, and diagnosed. Criteria such as resolution (ability to differentiate between different faults) achieved, cost and reliability of the sensor network, speed of detection and diagnosis, and false alarm rate are some of the criteria that can be incorporated in this approach.

In previous work, optimization formulations for sensor network design for reliable fault diagnosis have been presented (Bhushan and Rengaswamy, 2002*a*). These formulations incorporate various constraints on the sensor network design problem, as well as utilize quantitative information such as fault occurrence and sensor failure probabilities, and cost of sensors. In this article, formulations for incorporating robustness to the data (fault occurrence probability and sensor failure probability) used in the problem, are presented. Robustness to modeling errors/uncertainties is also considered. These formulations are presented in a lexicographic optimization framework. The application of the proposed formulations is demonstrated on the Tennessee Eastman case study.

2. RELATED PREVIOUS WORK

Bhushan and Rengaswamy (2002a) posed sensor location problem as a constrained lexicographic optimization problem based on the concept of unobservability minimization. The unobservability of a fault is defined as the probability of the fault occurring and remaining undetected and can be calculated as:

$$U_i = f_i \prod_{j=1}^{n} (s_j)^{(d_{ij} x_j)} \qquad (1)$$

Here index j is for the variables which can be potentially measured, x_j is the number of sensors placed to measure variable j with probability of failure of each sensor being s_j, index i is for the faults with f_i being the probability of occurrence of fault i, U_i is the unobservability of fault i, constant $d_{ij} = 1$ if fault i affects variable j and is zero otherwise. Construction of matrix D (with elements d_{ij}) is based on fault simulation either in the process DG, the process SDG, or the process SDG with gains (Bhushan and Rengaswamy, 2002*b*). For the case study presented later in this article, SDG with gains is used to generate the D matrix.

A good sensor network should lead to low unobservability values for the faults. The measure of process unobservability chosen (Bhushan and Rengaswamy, 2002*a*) is the maximum unobservability among all faults. Apart from unobservability minimization, cost minimization was also considered, and these two objectives were combined in a lexicographic manner. The sensor location design problem was posed as an integer linear programming (ILP) problem as follows:

Formulation I: One-Step Optimization considering unobservability and cost

$$\min_{(x_j)} [U - \alpha\ x_s] \qquad (2)$$

subject to,

$$U \geq \log U_i, \quad i = 1, ..., m \qquad (3)$$

$$\sum_{j=1}^{n} c_j x_j + x_s = C^* \qquad (4)$$

$$x_j \in \mathbb{Z}^+, \quad j = 1, ..., n \qquad (5)$$

$$x_s \in \mathbb{R}^+ \qquad (6)$$

In the above problem, the first term in the objective function U is the maximum unobservability among all faults (this is captured by constraints 3), C^* is the available cost (resources) for performing sensor location, and c_j is the cost of installing one sensor on variable j . The problem is linear in the decision variables (x_j) since instead of unobservability U_i (as given by equation 1) for a fault, the log of unobservability is being used. Hardware redundancy is incorporated in the above formulation by not restricting the decision variables to be binary, but rather allowing them to take any non-negative integer value. The variable x_s is the slack in the cost constraint, which takes non-zero real values. The higher the value of the variable x_s, lower is the cost used for sensor location. α is a positive constant which has to be chosen such that the primary objective (minimizing unobservability) still attains its earlier optimal value. For such a case, negative contribution of the second term ($\alpha\ x_s$) in the objective function will ensure that among all solutions which yield minimum unobservability, the one which has highest x_s will be chosen. Thus, if the constant α is appropriately chosen, the solution to Formulation I will give a sensor network, which will have the least cost among all the networks which yield the minimum unobservability. A sufficient range for the value of α was given as: $0 < \alpha < 1/(aC^*)$, where a is a constant that can be calculated based on the given fault and sensor probabilities. The optimization problem as presented above is known as lexicographic optimization , where the idea is to arrange multiple objectives in a lexicographic order. This ordering means that the more important objective is infinitely more important than a less important objective (Miettinen, 1998). Sherali (1982) has presented an algorithm for selecting suitable weights for combining multiple objectives in a lexicographic manner. The selection of constant α as presented above by Bhushan and Rengaswamy (2002a) is similar to the approach used by Sherali (1982). Bhushan and Rengaswamy (2002a) have also pointed out that the sensor network design problem as presented above can be solved not only for the basic fault observability case, but also for other cases, such as single or multiple fault resolution. The basic idea is to convert any given scenario into a suitable fault-observability case by generating appropriate pseudo-faults. The same holds true for formulations to be presented next in this article. Also, similar to formulation I, all the formulations presented in the next section are ILP problems.

3. ROBUST SENSOR NETWORK DESIGN

The formulation I for sensor location requires fault occurrence and sensor failure probability data, and assumes that this data is accurately known. In practice, this may not always be the case. Also, it is possible that the underlying fault-models (SDG with gains for this article) are not accurate and have errors/uncertainties in them. In this section, some criteria to incorporate robustness to these factors will be considered.

3.1 Robustness to available probability data

For a given process, it is quite possible that some fault occurrence probabilities are accurately known (based on experience or past plant data), and for others only approximate values are available. Sensor failure probability data would be comparatively easier to obtain as there are several possible sources of such data, such as vendors, instrumentation handbooks, or lab/pilot scale experiments. However, similar to fault occurrence probability data, it is quite possible that depending on the source/reliability of information, values for some sensor failure probabilities are only approximately known (for example, the values calculated using lab experiments may not be exactly same as that obtained during the actual process operation). Hence it is desirable to incorporate some robustness to such data mismatch in the sensor location formulations. In this section, two scenarios are considered: (i) inaccuracies in occurrence probabilities of some faults (all sensor probabilities accurately known) and (ii) inaccuracies in failure probabilities of some sensors (all fault occurrence probabilities accurately known). In the rest of the article, to simplify the notation, a fault whose unobservability calculation involves inaccurately known data, will be referred to as "inaccurate" fault. A fault can be inaccurate for either of the two scenarios.

The central idea is again based on the concept of lexicographic optimization. The primary objective still is to minimize system unobservability (maximum unobservability amongst all faults) calculated using nominal values for fault occurrence and sensor failure probabilities. The secondary objective is to try to maximize the slack in the unobservability constraints (3) of inaccurate faults. The idea here is that since the system unobservability characterizes only the maximum unobservability amongst all faults, a process with two different sensor networks can have the same system unobservability with both the networks, and yet have different unobservabilities for some faults. Amongst these two networks, one would like to choose the network which yields lower unobservability for the inaccurate fault. For a given overall system unobservability, the higher the slack in the unobservability constraint of a fault, the lower is its individual unobservability. In general, there can be more than one inaccurate fault. The secondary objective is then to maximize the minimum slack amongst all the slack variables in the unobservability constraints of these faults. With optimization of these two objective functions (overall system unobservability minimization and slack maximization for inaccurate faults), it is still possible to obtain multiple optimal solutions with possibly different cost required for each solution. The third objective is then to minimize the cost used. The three objective functions are combined in a single-weighted objective function using appropriately computed weighting constants.

The formulations for the two scenarios are presented next. Even though the basic idea for the two scenarios is the same, there are some differences.

Formulation II: Robustness to Inaccurate Fault Occurrence Probability Data

$$\min_{(x_j)} [\lambda_1 U - \lambda_2 \phi - x_s] \tag{7}$$

subject to,

$$\sum_{j=1}^{n} c_j x_j + x_s = C^* \tag{8}$$

$$U \geq \log(U_i), \;\; i = 1, ..., m_1 \tag{9}$$

$$U = \log(U_i) + \phi_i, \;\; i = m_1 + 1, ..., m \tag{10}$$

$$\phi \leq \phi_i, \;\; i = m_1 + 1, ..., m \tag{11}$$

$$x_j \in \mathbb{Z}^+, \;\; (x_s, \phi_i, \phi) \in \mathbb{R}^+ \tag{12}$$

In the above formulation, ϕ_i is the (positive) slack variable of the unobservability constraint for the i^{th} fault. This slack variable is added to the unobservability constraints of those faults (faults $m_1+1, ..., m$) whose occurrence probabilities are not accurately known. The secondary objective is to then maximize the minimum slack (ϕ) amongst all the faults with inaccurately known probabilities. Constraints 11 and maximization of ϕ in the objective function ensures that ϕ is equal to the minimum value amongst all individual ϕ_i values. Positive constants λ_1 and λ_2 ensure that the solution to Problem II solves a lexicographic optimization problem with unobservability minimization, slack maximization and cost minimization as the objectives in decreasing order of preference. The values for λ_1 and λ_2 can be derived using the idea presented by Sherali (Sherali, 1982) and are:

$$\lambda_1 = (1+M)(1+C^*) \tag{13}$$

$$\lambda_2 = (1+C^*) \tag{14}$$

where the constant

$$M = \max_{(i=m_1+1,...,m)} (\phi_i^*) \tag{15}$$

The constants ϕ_i^* are defined in the next paragraph. For the sake of brevity, the procedure followed in deriving the above values is not discussed in this article.

Formulation II as presented has one small problem. It can lead to solutions where extra cost is spent in increasing ϕ to meaningless high values. To understand this, consider a process with one inaccurate fault with nominal fault occurrence probability of 10^{-2}. In the worst case, the actual occurrence probability of this fault will tend to $10^0 = 1$. Considering constraint 10 for this fault, it can be seen that if ϕ_i for this fault is equal to 2, then even in the worst case, the system unobservability will not be higher than the value calculated using nominal fault occurrence probability data. Hence, spending extra cost to increase ϕ above 2 for this case is not required. This is achieved by replacing constraints 11 by the following set of constraints

$$\phi \le M \tag{16}$$

$$\phi \le \phi_i + My_i, \;\; i = m_1+1, ..., m \tag{17}$$

$$Py_i \ge \phi_i - \phi_i^*, \;\; i = m_1+1, ..., m \tag{18}$$

$$P(y_i - 1) \le \phi_i - \phi_i^*, \;\; i = m_1+1, ..., m \tag{19}$$

$$y_i \in \{0, 1\}, \;\; i = m_1+1, ..., m \tag{20}$$

In the above constraints, P is a large positive constant (such as 100), y_i is a binary variable, and ϕ_i^* is the maximum meaningful value of the slack variable for the i^{th} inaccurate fault (2 in the above example). The basic idea behind introducing variable y_i is that if for a fault i, the slack variable $\phi_i > \phi_i^*$, then the value ϕ is not restricted by value of ϕ_i, and increasing ϕ_i further will not increase ϕ. Constraints 18 and 19 ensure that $y_i = 1$ for this case, and this ensures that ϕ_i (constraint 17) does not limit the upper value of ϕ. For the case when $\phi_i < \phi_i^*$, constraints 18 and 19 force y_i to be 0, and this reduces constraint 17 for fault i to $\phi \le \phi_i$ as it should be. For the case when $\phi_i = \phi_i^*$, constraints 18 and 19 allow y_i to be either 0 or 1, but maximization of ϕ in the objective function ensures that $y_i = 1$, if the maximum value of ϕ is being restricted by ϕ_i (constraint 17). Constraint 16 ensures that the maximum value for ϕ is not more than the maximum meaningful value among all the slack variables.

To summarize, the formulation that is proposed for incorporating robustness to fault occurrence probability data is: Formulation II with constraints 16-20 used in place of constraints 11. Formulation to incorporate robustness to sensor failure probability data is considered next.

Formulation III: Robustness to Inaccurate Sensor Failure Probability Data

$$\min_{(x_j)} [\lambda_1 U - \lambda_2 \phi - x_s] \tag{21}$$

subject to,

$$\sum_{j=1}^{n} c_j x_j + x_s = C^* \tag{22}$$

$$U = \log(U_i) + \phi_i, \;\; i = 1, ..., m \tag{23}$$

$$\phi_i^* = -\sum_{j \in S_u} d_{ij} (\log s_j) x_j, \;\; i = 1, ..., m \tag{24}$$

$$Py_i \ge \phi_i - \phi_i^*, \;\; i = 1, ..., m \tag{25}$$

$$P(y_i - 1) \le \phi_i - \phi_i^*, \;\; i = 1, ..., m \tag{26}$$

$$\phi \le \phi_i + My_i, \;\; i = 1, ..., m \tag{27}$$

$$\phi \le M \tag{28}$$

$$x_j \in \mathbb{Z}^+, \;\; (x_s, \phi_i, \phi) \in \mathbb{R}^+, y_i \in \{0, 1\} \tag{29}$$

In the above formulation, S_u is the set of sensor with uncertain failure probabilities, constant M is the maximum meaningful value for ϕ for a given problem, and P is a large positive constant (for example, 100). The constant M is now given as

$$M = \max_i \sum_{j \in S_u} d_{ij} (\log s_j) x_j^* \tag{30}$$

where x_j^* is an upper bound on the maximum number of j^{th} sensors that can be selected, and can be calculated as quotient of C^*/c_j. Constraint 24 is written for each fault, and it calculates the maximum meaningful value of slack required in a fault unobservability constraint, based on the chosen sensors. For example, suppose that the unobservability calculation of a fault does not depend on any sensor with inaccurate failure probability. Then no slack is required for this fault constraint. As can be seen from equation 24, ϕ_i^* for this fault will be 0. Constraints 25-28 are same as the constraints used in Formulation II.

Formulation II incorporated robustness to inaccurate fault occurrence probabilities (assuming all sensor failure probabilities were exactly known), and formulation III considered robustness to inaccurate sensor failure probabilities (assuming all fault occurrence probabilities were exactly known). A formulation where both types of uncertainties are considered can also be formulated as a combination of the two formulations. This formulation is not presented in this article.

3.2 Robustness to Modeling Errors

Formulations to incorporate robustness to inaccurate probability data were presented above. It can also be the case that the effects of faults on the measurable variables (matrix D with elements d_{ij} in our formulations) as used in the formulation are not accurate. This may happen since the process may not be exactly known and/or the simulation procedure to obtain the matrix D may make some restrictive assumptions (such as steady state gains for SDG with gain model). Also, since the effects of process/sensor noise, fault magnitude, and sampling times for sensors have not been included in the formulations, the effects of various faults predicted on the variables may not match exactly with those measured in the plant. The efficacy of a selected sensor network will then be affected by these unknown factors. In order to incorporate some robustness to such modeling errors, it is proposed that a distributed sensor network should be preferred, since for a given fault, even if effects on some variables are wrongly modeled, the chances of correctly diagnosing that fault would be higher since some other variables being effected by that fault are also being measured. Once again, this is done in a lexicographic framework, and the formulation that is proposed to achieve this is as follows:

Formulation IV: Lexicographic formulation for achieving a distributed sensor network when some fault occurrence probabilities are inaccurately known

$$\min_{(x_j)} [\alpha_1 U - \alpha_2 \phi - \alpha_3 x_s - N] \tag{31}$$

subject to

$$\sum_{j=1}^{n} c_j x_j + x_s = C^* \tag{32}$$

$$U = \log(U_i) + \phi_i, \;\; i = 1, ..., m \tag{33}$$

$$\phi \leq M \tag{34}$$

$$\phi \leq \phi_i + M y_i, \;\; i = m_1 + 1, ..., m \tag{35}$$

$$P y_i \geq \phi_i - \phi_i^*, \;\; i = m_1 + 1, ..., m \tag{36}$$

$$P(y_i - 1) \leq \phi_i - \phi_i^*, \;\; i = m_1 + 1, ..., m \tag{37}$$

$$n_j \leq x_j, \quad j = 1, ..., n \tag{38}$$

$$N = \sum_{j=1}^{n} n_j \tag{39}$$

$$N, x_j \in \mathbb{Z}^+, \quad j = 1, ..., n \tag{40}$$

$$x_s, \phi_i, \phi \in \mathbb{R}^+ \tag{41}$$

$$n_j \in \{0, 1\}, \quad j = 1, ..., n \tag{42}$$

$$y_i \in \{0, 1\}, \;\; i = m_1 + 1, ..., m \tag{43}$$

In the above formulation, N is the total number of variables measured in the process (different from total number of sensors used since a variable may have more than one sensor), and characterizes the sensor distribution in the process. Variable n_j is a binary variable, which is 1 if variable j is measured (irrespective of the number of sensors used to measure variable j), and is 0 if variable j is not measured. This is ensured by constraints 38, 39 and maximization of N in the objective function. Constraints 33-37 are the same as those used in Formulation II. α_1,α_2 and α_3 are positive constants which ensure that solution to Problem IV is optimal in the lexicographic sense with unobservability minimization, slack maximization, cost minimization, and network distribution maximization being the four objectives in decreasing order of priority. These constants can be calculated using the algorithm by Sherali (1982) as:

$$\alpha_1 = (1 + N^*)(1 + C^*)(1 + M) \tag{44}$$

$$\alpha_2 = (1 + N^*)(1 + C^*) \tag{45}$$

$$\alpha_3 = (1 + N^*) \tag{46}$$

where N^* is an upper bound on the maximum number of different sensors that can be selected in the process for a given available cost C^*, and can be calculated based on the given data. Even though theoretically sound, depending on the values of constants N^*, C^*, M, in the case study section, for some cases scaling problems were encountered, which were handled by appropriately scaling some variables (such as N). Alternatively, for this case, one can obtain the lexicographic solution by solving more than one problem in sequence (Sherali, 1982).

Formulation IV is posed assuming that the some fault probability data is inaccurately known. Incase, some sensor failure probabilities are not accurately known, a formulation similar to Formulation IV can be posed, with constraints borrowed from Formulation III. This formulation is referred to as formulation V in the case study section.

4. CASE STUDY: TENNESSEE EASTMAN PROCESS

To illustrate the utility of the ideas presented in this article, the Tennessee Eastman (TE) process (Downs and Vogel, 1993) is used as a case study.

Table 1. Results for inaccurate fault occurrence probability data

Form.	C^*	Cost Used	U	ϕ	N	Sensors Selected
I	2900	2100	-5	0	5	2,8,9,13,58
	5400	4200	-8	0	5	2(2),8(2),9(2) 13(2),55(2)
II	2900	2200	-5	2	6	1,2,8,9,13,58
	5400	4300	-8	2	8	1,2(2),5,7,8, 9,13(2),57(2)
IV	2900	2200	-5	2	6	1,2,8,9,13,54
	5400	4300	-8	2	9	1,2(2),5,7,8, 9,13(2),57,58

Based on a reduced model of this process proposed by (Ricker and Lee, 1995), Bhushan and Rengaswamy (2002b) developed a signed digraph with gains for this process, and used it to generate the cause-effect information required for sensor network design. This cause-effect information is used in this section to design sensor networks for problems posed in this article. In this process, 61 potential measurements and 15 faults are considered. These possible locations of sensors, sensor failure probabilities, sensor costs, faults considered and probabilities of their occurrences and the diagraph based cause-effect model used, are taken from (Bhushan and Rengaswamy, 2002*b*) and are not presented here. All the results presented in this section are only for the single fault resolution (Bhushan and Rengaswamy, 2002*a*) case and were obtained by solving the formulations in the integer linear programming package LINDO.

Case I Robust design in the presence of inaccuracies in fault occurrence probability data: It is now assumed that the occurrence probabilities for faults $F14$ and $F15$ (with nominal values of 10^{-2} and 10^{-1} respectively) are not accurately known. Sensor location is performed for different cases and the some typical results are summarized in Table 1. Comparing the results of formulation I and II for available cost $C^* = 2900$ units, it is seen that for formulation II (where slack maximization for inaccurate faults is considered), the selected sensor network ensures that even if the true probability of the inaccurately known faults ($F14, F15$) were to be higher by 2 orders of magnitude (than their nominal values used in the optimization), the overall system unobservability will not change. The same cannot be said about the sensor location obtained with Formulation I. Comparison of results for $C^* = 5400$ for formulations II and IV indicate that the network is more distributed for formulation IV than for II, thereby incorporating robustness to some amount of modeling error as compared to Formulation II.

Case II Robust design in the presence of inaccuracies in sensor failure probability data: Results for cases when failure probabilities of sensors $S13$ and $S60$ (with nominal values of 10^{-3} each) are inaccurately known are presented in Table 2. Conclusions regarding utility of various formulations similar to Table 1 can be derived based on this Table also.

Table 2. Results for inaccurate sensor failure probability data

Form.	C^*	Cost Used	U	ϕ	N	Sensors Selected
III	2900	2700	-5	6	5	2,8,9,13(3),55
	5400	5400	-8	12	5	2(2),8(2),9(2), 13(6),58(2)
V	2900	2700	-5	6	5	2,8,9,13(3),57
	5400	5400	-8	12	8	2(2),5,7,8, 9,13(6),55,57

5. CONCLUSIONS

A solution to the problem of designing robust sensor networks for fault diagnosis has been attempted in this article. Depending on the requirements of the fault monitoring system, appropriate optimization problems have been posed. The proposed formulations incorporate various criteria, such as reliability (in terms of unobservability), cost, and robustness to fault/sensor probability data and process-model mismatch, in a lexicographic optimization framework. In the case study section, the results of the application of the various formulations to the Tennessee Eastman process are compared. These demonstrate the utility of the various optimization objectives presented in this article.

REFERENCES

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

Bhushan, M. and R. Rengaswamy (2002*b*). Comprehensive design of a sensor network for chemical plants based on various diagnosability and reliability criteria: II. application. *Ind. Eng. Chem. Res.* **41**, 1840–1860.

Downs, J. J. and E. F. Vogel (1993). A plant-wide industrial process control problem. *Comput. Chem. Engng.* **17**(3), 245.

Miettinen, K. M. (1998). *Nonlinear Multiobjective Optimization.* Kluwer Academic Publishers: Boston.

Ricker, N. L. and J. H. Lee (1995). Nonlinear modeling and state estimation for the tennessee eastman challenge process. *Comput. Chem. Engng.* **19**(9), 983.

Sherali, H. D. (1982). Equivalent weights for lexicographic multi-objective programs: Characterizations and computations. *European Journal of Operational Research* **11**, 367.