New MILP Formulation for Instrumentation Network Design and Upgrade

Miguel Bagajewicz and Enmanuel Cabrera
School of Chemical Engineering and Materials Science, University of Oklahoma, Norman, OK 73019

The problem of instrumentation network design and upgrade consists of determining the optimal set of instruments to obtain sufficiently accurate and reliable estimates of variables. An existing formulation of the cost optimal sensor network design and upgrade problem (Bagajewicz, 1997) is in the form of a mathematical programming problem, but it has the disadvantage that the constraints are not stated explicitly in analytical form, forcing therefore a special tree search solution procedure. An alternative MILP formulation is presented, which is useful for small to medium size problems, where all constraints are explicit.

Introduction

Instrumentation is needed in process plants to obtain data that are essential to perform several activities: control, quality assurance, production or yield accounting and fault detection. In addition, parameter estimation like heat exchanger fouling or column efficiencies, among others, are becoming increasingly important, especially for new techniques such as online optimization, where the construction of reliable computer models is essential.

The problem of instrumentation network design and upgrade consists of determining the optimal set of instruments such that sufficiently accurate and reliable estimates of variables of interest is obtained, while, at the same time, bad data due to possible instrument malfunction are filtered.

There are several articles and book chapters studying this problem. Different objective functions were used (precision, cost, reliability, and so on) and different techniques were employed (graph theory, mathematical programming, genetic algorithms, and heuristic searching) to solve the problem. Among the book chapters that one can quote are the design of sensor networks for variable observability and precision using graph theory in Madron (1992) and the chapter dedicated to the issue in Narasimhan and Jordache's (2000) book. Additionally, many articles are reviewed in the book by Bagajewicz (2000). These articles use different objective functions: maximum precision, minimum cost, maximum reliability, and so on. However, there is a connection between all these models. Indeed, a special theorem due to Tuy (1987) can be used to show that, for example, a maximum precision model constrained by cost is equivalent to a minimum cost problem constrained by precision to the extent that the solution of one is one solution of the other (Bagajewicz and Sánchez, 1999). This idea was extended to the design of reliable systems by Bagajewicz and Sánchez (2000a) showing that the connection between a maximum reliability problem constrained by cost (or the number of sensors) is equivalent to a minimum cost (or number of sensors) problem constrained by reliability in its various definitions. Bagajewicz (2000) discusses other extensions. We therefore concentrate on the minimum cost model.

Bagajewicz (1997) also incorporated a few gross error robustness criteria related to the ability of the network to handle the presence of gross errors (biases, leaks) effectively. Finally, upgrade issues were discussed by Bagajewicz and Sánchez (2000b) and the role of maintenance was analyzed by Sánchez and Bagajewicz (2000).

To solve the cost optimal design problem, Bagajewicz (1997) introduced binary variables to define whether a sensor is measuring a variable or not. In attempting to define a standard mathematical programming model, he showed that the constraints were not amenable to be written explicitly in terms of these binary variables. Thus, he used a special branch and bound procedure, which had some tree pruning capabilities. Recently, Chmielewski et al. (1999) offered an alternative formulation based on linear matrix inequalities (LMI), which is convex and allows the constraints to be written explicitly in terms of the binary variables. Based on this alternative formulation, a user can apply the standard branch and bound search algorithm to find all globally optimal solutions. Chmielewski (2001) also presents an extension to the dynamic data reconciliation case. In their model, integers are,

Correspondence concerning this article should be addressed to M. Bagajewicz.

however, still needed. Chmielewski (2001) presents further results of this approach.

In this article an alternative MILP formulation of the problem is presented. Examples are given to illustrate the effectiveness of the model.

Problem Statement

The cost-optimal sensor network of a system with precision constraints is obtained solving the following optimization problem (Bagajewicz, 1997)

$$\left.\begin{array}{l}\text{Min} \sum_i c_i \mu_i \\ \text{s.t.} \\ \hat{s}_i(\mu) \le s_i^* \forall i \in M_p\end{array}\right\} \tag{1}$$

where μ is the vector of binary variables indicating that a sensor is located in variable i, c_i is the cost of such a sensor, and M_p is the set of variables of interest. The precision constraint states that the variance of the estimates $\hat{s}_i$ (obtained through data reconciliation when appropriate) has to be smaller or equal to certain variance threshold s_i^* for the variables of interest (M_p). In addition, one can easily restrict the set of streams where sensors can be located by setting the corresponding binary variables to zero.

When all variables are measured, the variance of the estimates obtained from reconciliation is given by

$$\hat{\boldsymbol{S}} = \boldsymbol{S} - \boldsymbol{S}\cdot\boldsymbol{A}^T(\boldsymbol{A}\cdot\boldsymbol{S}\cdot\boldsymbol{A}^T)^{-1}\boldsymbol{A}\cdot\boldsymbol{S} \tag{2}$$

where $\boldsymbol{A}$ is the material balance matrix of the system, $\boldsymbol{S}$ is the variance matrix. Therefore, $\hat{s}_i = \hat{S}_{ii}$. When only a subset of all variables is measured, then an observability analysis is needed, but Eq. 2 applies using a matrix of redundant measured variables $\boldsymbol{A_R}$ instead of $\boldsymbol{A}$. In such a case, other equations are then needed for the unmeasured and observable variables (Bagajewicz, 1997). This use of different equations for different sets of sensors (different instances of $\boldsymbol{\mu}$) makes writing explicit constraints difficult (Bagajewicz, 1997).

An extension of the model to use many instruments as candidates to measure a certain variable (Bagajewicz, 1997, 2000) was also made. Let l_i^m be the number of different alternative candidates of measurement devices and let the cost of each of these candidates be given by $c_{i,k}$ ($k=1, \ldots, l_i^m$). Then, for each variable, binary variables $\mu_{i,k}$ ($k=1, \ldots, l_i^m$) are introduced to determine which candidate instrument is used, that is, $\mu_{i,k}$ is equal to one if device k is used to measure variable i, and zero otherwise. The model becomes

$$\left.\begin{array}{ll}\text{Min} \sum_{i=1}^{m}\sum_{j=1}^{l_i^m} c_{ij}\mu_{ij} & \\ \text{s.t.} & \\ \hat{s}_i(\mu) < s_i^* & \forall i \in M_p \\ 0 \le \sum_{j=1}^{l_i^m} \mu_{ij} \le d_i & \forall i \in M \\ \mu_{ij} \in \{0,1\} & \forall i \in M, \forall j \in l_i^m\end{array}\right\} \tag{3}$$

The last constraint guarantees that, at most, d_i devices are assigned to each variable. When $d_i = 1$, not more than one measurement per variable is allowed, that is, only systems with spatial redundancy and no hardware redundancy are considered.

MILP Model

Although one will almost always want to consider hardware redundancy in the formulation, we start by developing a model with no hardware redundancy for simplicity of presentation.

By introducing a change of variables, Eq. 2 can be rewritten as follows

$$\left.\begin{array}{l}\hat{\boldsymbol{S}} = \boldsymbol{S} - \boldsymbol{S}\cdot\boldsymbol{A}^T\cdot\boldsymbol{P}\cdot\boldsymbol{A}\cdot\boldsymbol{S} \\ \boldsymbol{P}(\boldsymbol{A}\cdot\boldsymbol{S}\cdot\boldsymbol{A}^T) = \boldsymbol{I}\end{array}\right\} \tag{4}$$

When only spatial redundancy is allowed, the elements of the diagonal of the matrix $\boldsymbol{S}$ are: $S_{ii} = \sum_{t=1}^{l_i^m} \mu_{it}\sigma_{it}^2$, with $S_{ij} = 0$, $\forall i \ne j$. Thus, Eq. 4 becomes

$$\left.\begin{array}{l}\hat{S}_{ii}(\mu) = \sum_{t=1}^{l_i^m} \mu_{it}\sigma_{it}^2 - \sum_{t=1}^{l_i^m}\sum_{k=1}^{n}\sum_{d=1}^{n} \mu_{it}\sigma_{it}^4 a_{ki} p_{kd} a_{di} \\ \sum_{d=1}^{n}\sum_{i=1}^{m}\sum_{t=1}^{l_i^m} p_{kd} a_{di} \sigma_{it}^2 \mu_{it} a_{ji} = \delta_{kj}\end{array}\right\} \tag{5}$$

The set of Eqs. 5 can only be used to calculate the variance of the estimates when all variables are measured. To extend it to systems with no measurements, we assume that a fake sensor of very large variance and no cost is used for the locations where no sensor is installed. This was also assumed by Chmielewski et al. (1999) and Chmielewski (2001). The validity of this substitution is proven in the Appendix.

Finally, the absence of hardware redundancy and the fact that at least one sensor will be installed (fake or real) requires replacing the last constraint in Eq. 3 by

$$\sum_{t=1}^{l_i^m} \mu_{it} = 1 \qquad \forall i \in M \tag{6}$$

Equation 5 includes the product of a binary variable and a continuous variable ($p_{kd}\mu_{it}$). We now recall that $z = p\mu$ is equivalent to $G(z, p, \mu, \Omega) \ge 0$ (Glover, 1975) with

$$G(z, p, \mu, \Omega) = \begin{cases}\Omega\cdot\mu - z \\ z \\ (1-\mu)\cdot\Omega - (p - z) \\ p - z\end{cases} \tag{7}$$

where Ω is an upper bound of p. One can verify that when $\mu = 0$, variable z is forced to be also equal to zero by the first two constraints, while the rest are trivial. Conversely, when $\mu = 1$, then $z = p$, which is forced by the second two constraints with the first two being trivial. This transformation linearizes products of continuous and binary variables by substituting them with a linear set of inequalities. The price of this linearization technique is an increase in the problem size by the addition of new continuous variables and model equations. However, the obvious advantage is that more efficient tools are available to solve integer linear problems.

However, this transformation is valid only if p is positive. Thus, we redefine p_{dk} as $p_{dk} = p_{dk,1} - p_{dk,2}$, where both $p_{dk,1}$ and $p_{dk,2}$ are positive. Then, the transformation $z_{dkit} = p_{dk}\,\mu_{it}$ becomes

$$\left.\begin{array}{l} z_{dkit} = p_{dk,1}\,\mu_{it} - p_{dk,2}\,\mu_{it} = z_{dkit,1} - z_{dkit,2} \\ G(z_{dkit,1},\, p_{dk,1},\, \mu_{it},\, \Omega_1) \ge 0 \\ G(z_{dkit,2},\, p_{dk,2},\, \mu_{it},\, \Omega_2) \ge 0 \end{array}\right\} \tag{8}$$

Finally, when Eq. 6 is used, one can reduce the number of binary variables by writing

$$\mu_{il_i^m} = 1 - \sum_{j=1}^{l_i^m - 1} \mu_{ij} \tag{9}$$

With the introduction of the above transformations the model is MILP.

Let n_I be the number of balance equations, n_J the number of streams in the system and n_T the number of sensor options. The nonlinear model originally has $(n_I^2 + 1)$ continuous variables, and the linearization process adds $(2n_J n_T + 1)n_I^2$ new continuous variables. Examples are presented below to illustrate the application of this model.

Hardware Redundancy

The variance of a variable being measured by several sensors can be expressed as a function of the variances of each sensor. One alternative for hardware redundancy is to define new sensors having a variance and cost corresponding to the installation of these sensors. Another alternative is to develop a representation specific to the use of multiple sensors. For this purpose, the variance of a variable being measured by several sensors can be expressed as a function of the variances of each sensor as follows (Chmielewski, 2001)

$$\frac{1}{S_{ii}} = \sum_{t=1}^{l_i^m} \frac{\mu_{it}}{\sigma_{it}^2} \tag{10}$$

which in turn can be rewritten as follows

$$S_{ii} = \frac{\lambda_i}{\sum_{t=1}^{l_i^m} \mu_{it}\gamma_{it}} \tag{11}$$

where

$$\lambda_i = \prod_{t=1}^{l_i^m} \sigma_{it}^2,\ \gamma_{it} = \prod_{\substack{r=1 \\ r \ne t}}^{l_i^m} \sigma_{ir}^2.$$

One can easily verify that Eq. 11 reduces to

$$S_{ii} = \sum_{t=1}^{l_i^m} \mu_{it}\sigma_{it}^2$$

when only one sensor is allowed. Substituting S_{ii} from Eqs. 11 in Eq. 5, one obtains

$$\left.\begin{array}{l} \dfrac{\lambda_i}{\sum_{t=1}^{l_i^m} \mu_{it}\gamma_{it}} - \sum_{k=1}^{n} \sum_{d=1}^{n} \dfrac{a_{ki}\,p_{kd}\,a_{di}\,\lambda_i^2}{\sum_{j=1}^{l_i^m} \sum_{t=1}^{l_i^m} \mu_{ij}\,\mu_{it}\gamma_{ij}\gamma_{it}} < s_i^* \\ \sum_{h=1}^{n} \sum_{k=1}^{m} \dfrac{p_{ih}a_{hk}a_{jk}\lambda_k}{\sum_{t=1}^{l_i^m} \mu_{kt}\gamma_{kt}} = \delta_{ij} \end{array}\right\} \tag{12}$$

The equations above are nonlinear. In order to linearize them, we create new variables β_{ihk}, ρ_i and θ_{ikd} defined as follows

$$\left.\begin{array}{l} \beta_{ihk} \sum_{t=1}^{l_i^m} \mu_{kt}\gamma_{kt} = p_{ih} \\ \theta_{ikd} \sum_{j=1}^{l_i^m} \sum_{t=1}^{l_i^m} \mu_{ij}\,\mu_{it}\gamma_{ij}\gamma_{it} = p_{kd} \\ \rho_i \sum_{t=1}^{l_i^m} \mu_{it}\gamma_{it} = 1 \end{array}\right\} \tag{13}$$

Introducing these new variables, the set of equations in Eq. 12 becomes

$$\left.\begin{array}{l} \rho_i\lambda_i - \sum_{k=1}^{n} \sum_{d=1}^{n} a_{ki}a_{di}\lambda_i^2\theta_{ikd} < s_{ii}^* \\ \sum_{h=1}^{n} \sum_{k=1}^{m} a_{hk}a_{jk}\lambda_k\,\beta_{ihk} = \delta_{ij} \end{array}\right\} \tag{14}$$

The equations are still nonlinear, but with the nonlinearities now consisting of products of one continuous variable and one or two binary variables. To linearize the product $\beta_{ihk}\,\mu_{kt}$, we define $x_{ihkt} = \beta_{ihk}\,\mu_{kt}$. The product $\beta_{ihk}\,\mu_{kt}$ is replaced by the following linear system

$$\left\{\begin{array}{l} x_{ihkt} = \beta_{ihk,1}\,\mu_{kt} - \beta_{ihk,2}\,\mu_{kt} = x_{ihkt,1} - x_{ihkt,2} \\ G(x_{ihkt,1},\, \beta_{ihk,1},\, \mu_{kt},\, \Omega_1) \ge 0 \\ G(x_{ihkt,2},\, \beta_{ihk,2},\, \mu_{kt}, \Omega_2) \ge 0 \end{array}\right. \tag{15}$$

Likewise, we define $y_{it} = \rho_i\,\mu_{it}$, which, since ρ_i is always positive, can be replaced by the system of inequalities $G(y_{it},\, \rho_i,\, \mu_{it},\, \Xi_i) \ge 0$, where Ξ_i is an upper bound for ρ_i. The last term to linearize is $\theta_{ikd}\,\mu_{ij}\,\mu_{it}$. We first make the following definitions

$$\left\{\begin{array}{l} z_{ikdjt} = \theta_{ikd}\alpha_{ijt} \\ \alpha_{ijt} = \mu_{ij}\,\mu_{it} \end{array}\right. \tag{16}$$

Since θ_{ikd} can be either positive or negative, then the following holds

$$\begin{cases} z_{ikdjt} = \theta_{ikd,1}\alpha_{ijt} - \theta_{ikd,2}\alpha_{ijt} = z_{ikdjt,1} - z_{ikdjt,2} \\ G\left(z_{ikdjt,1}, \theta_{ikd,1}, \alpha_{ijt}, \Lambda_1 \right) \geq 0 \\ G\left(z_{ikdjt,2}, \theta_{ikd,2}, \alpha_{ijt}, \Lambda_2 \right) \geq 0 \end{cases} \tag{17}$$

Finally, the product $\alpha_{ijt} = \mu_{ij}\mu_{it}$ can be replaced by $H(\alpha_{ijt}, \mu_{it}, \mu_{ij}) \geq 0$ (Glover, 1975), where

$$\boldsymbol{H}(\alpha_{ijt}, \mu_{it}, \mu_{ij}) = \begin{cases} \mu_{it} - \alpha_{ijt} \\ (1 - \mu_{it}) - (\mu_{ij} - \alpha_{ijt}) \\ \mu_{ij} - \alpha_{ijt} \\ \alpha_{ijt} \end{cases} \tag{18}$$

Clearly, when $\mu_{ij} = 0$, the last two equations set $\alpha_{ijt} = 0$ and the first two are satisfied. When $\mu_{ij} = 1$ and $\mu_{it} = 0$, the first two equations set $\alpha_{ijt} = 0$ and the other two are satisfied. Finally, when $\mu_{ij} = 1$ and $\mu_{it} = 1$, the first two equations set $\alpha_{ijt} = 0$ and the other two are satisfied. Finally, when $\mu_{ij} = 1$ and $\mu_{it} = 1$, the first two equations set $\alpha_{ijt} = 1$. This linearization technique also increases the size of the model by adding new continuous variables and model equations; nevertheless, it allows the problem to be linear, which in almost all situations is a more desirable quality than having fewer continuous variables.

At last, when all terms are linearized and the substitutions explained above are made, this extension of the original model becomes MILP. Finally, we need to add

$$\left.\begin{array}{l} 1 \leq \sum\limits_{t=1}^{l_i^m} \mu_{it} \leq d_i \\ \sum\limits_{t=1}^{l_i^m - 1} \mu_{it} \geq 1 \Rightarrow \mu_{il_i^m} = 0 \end{array}\right\} \tag{19}$$

In this case of hardware redundancy, the nonlinear model has the same number of continuous variables as the model for no hardware redundancy. Nevertheless, the linearization process adds $(2n_I^2 n_J(n_T + 2) + n_B\lfloor(2 + n_T) + 2n_I^2 + (1 + n_T)^2(1 + 2n_I^2)\rfloor)$ new continuous variables (n_B is the number of key variables for precision).

Numerical Aspects and Scaling

The use of sensors with large variance to model nonmeasured variables has numerical problems. Indeed, the use of such large numbers makes some elements of the matrix $\boldsymbol{ASA^T}$ very large, making it almost singular. Thus, scaling is required to deal with the inverse matrix numerically. We chose to scale the matrix $\boldsymbol{A}$ by dividing all its elements by a large number. Indeed, defining $\boldsymbol{B} = \boldsymbol{A}/\Phi$, where Φ is large scalar and substituting on Eq. 2, one obtains

$$\hat{\boldsymbol{S}} = \boldsymbol{S} - \boldsymbol{S} \cdot \Phi \boldsymbol{B}^T \left(\Phi \boldsymbol{B} \cdot \boldsymbol{S} \cdot \Phi \boldsymbol{B}^T \right)^{-1} \Phi \boldsymbol{B}^T \cdot \boldsymbol{S}$$
$$\Rightarrow \hat{\boldsymbol{S}} = \boldsymbol{S} - \boldsymbol{S} \cdot \boldsymbol{B}^T \left(\boldsymbol{B} \cdot \boldsymbol{S} \cdot \boldsymbol{B}^T \right)^{-1} \boldsymbol{B}^T \cdot \boldsymbol{S} \tag{20}$$

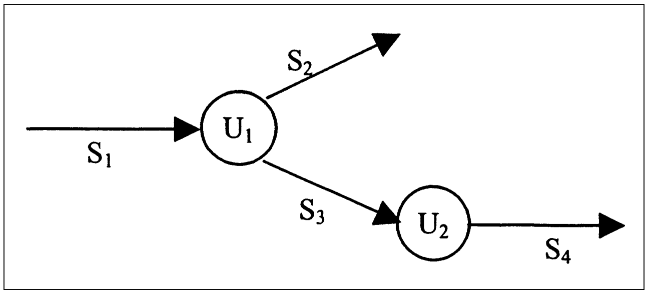

Figure 1. Example 1.

Thus, matrix $\boldsymbol{P}$ (inverse of $\boldsymbol{ASA^T}$) becomes $\boldsymbol{P} = (\boldsymbol{BSB^T})^{-1} = \Phi^2(\boldsymbol{ASA^T})^{-1}$. From Eq. 20, it follows that the rest of the model is not affected by this scaling. The model was implemented in GAMS and solved using CPLEX.

Besides the scaling, one has to choose a value for the variance of the nonmeasured variables to be large enough for the model to account for it as being infinite, while not making the matrix $\boldsymbol{ASA^T}$ singular. Values of the order of 10^3 are large enough for almost all design problems; and the elements of the matrix $\boldsymbol{P}$ are manageable when using the scaling approach explained above.

The upper bounds (Ω) used for the linearization of Eqs. 7 also need to be carefully chosen. Because the elements of the matrix $\boldsymbol{P}$ can take a wide range of values, these bounds could be very different for each element of the matrix. In this work, the maximum value for p_{ij}, $\forall i,j$ was obtained by a straightforward iterative process performed on MS Excel. A more accurate approach would be having an upper bound for each element of the matrix $\boldsymbol{P}$.

Example 1

Consider the process flow diagram of Figure 1 taken from Bagajewicz (1997).

The flow rates for the different streams (S_1, S_2, S_3 and S_4) are given by $F = (151.1{,}52.3{,}97.8{,}97.8)$. Flowmeters of precision 3%, 2% and 1% are available at costs 800, 1,500 and 2,500, respectively, regardless of size. Precision is only required for variables F_1 and F_4 with $\sigma_1^* = 1.5\%$ and $\sigma_4^* = 2.0\%$. No hardware redundancy is allowed. The results are the same as those obtained by Bagajewicz (1997) and are shown in Table 1. To get the alternative optimal solution, one optimal solution was forbidden. CPLEX explored 13 nodes of a total of 2^{12} possible and performed a total of 109 LP iterations to find each solution. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is less than 1 s.

In this first example the nonlinear model has five continuous variables, while the linearized problem has 105 continuous variables.

Table 1. Results of Example 1

	F_1	F_2	F_3	F_4	Cost $
Sol. A	—	2%	2%	—	3,000
Sol. B	—	2%	—	2%	3,000

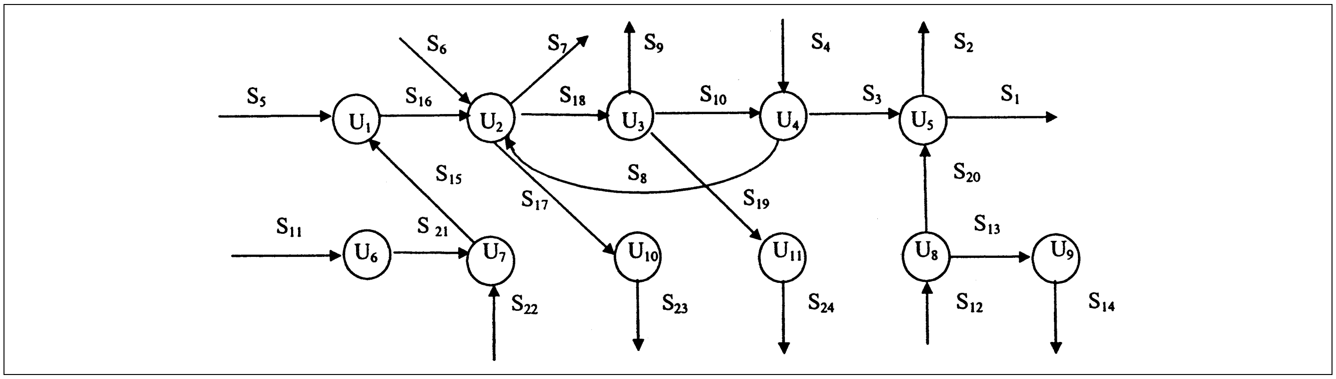

Figure 2. Example 2 (taken from Madron and Veverka, 1992).

Example 2: retrofit problem

The network of Figure 2 was proposed by Madron and Veverka (1992). It contains eleven nodes and 24 streams, nine of which are unmeasured (streams S_1 through S_9) and 15 streams are already measured (S_{10} through S_{24}). The candidates to be measured are streams S_1 through S_9.

Originally, the problem requested selecting what measurements should be added to make streams S_1 through S_5 observable. Meyer et al. (1994) added the costs of Table 2 and the flow rates of Table 3 were considered. It is assumed that all the candidate and existing sensors have a precision of 2.5%.

Madron and Veverka as well as Meyer et al. (1994) found the solution to be $x_m = \{F_2, F_4, F_8\}$ with a total cost of $C_T = 35$. By using the MILP model proposed in this article, the same solution was found. CPLEX explored 12 nodes out of a total of 2^9 and performed 2,927 LP iterations before finding the optimal solution. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is 8 s.

The nonlinear model has 122 continuous variables, and the linearization process adds 5,929 new continuous variables. However, the solving time does not increase in the same proportion.

Example 3: effect of hardware redundancy

Consider the network of Example 1 with the same flow rates and precision constraints. If hardware redundancy of order two is allowed, the optimal solutions are the same as those shown in Table 1. If the costs of the sensors of 3% precision are modified to be 700, instead of 800, four different optimal solutions exist, two with hardware redundancy and two without it (see Table 4).

Although all four options are equivalent from the point of view of cost, it is more desirable to choose one of the solutions with hardware redundancy because it will generate a more robust sensor network. For instance, if one of the sensors is found to have a gross error and is eliminated, a network with hardware redundancy would still have a better level of precision than a network without hardware redundancy. CPLEX explored 108 nodes out of a total of 2^{20} and performed 10,545 LP iterations before finding the optimal solution. The original nonlinear model has five continuous variables (as in example 1), and the linearization process adds 1,188 new ones. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is 5 s.

Table 2. Cost of Flowmeters (Meyer et al., 1994)

Stream	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9
Cost	19	17	13	12	25	10	7	6	5

Example 4: hardware redundancy in retrofit problem

Consider again the flowsheet in Figure 2 with the same flow rates and costs as in example 2. Precision is required to be at least of 2% for the flow rates of streams 1–5, and as much as two devices can be measuring a single variable ($d_i = 2$). The optimal solution has a total cost of $C_T = 168$ with devices located in all streams and an additional device located in streams S_2, S_4 and S_5. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is 17 min. CPLEX explored 92 nodes out of a total of 2^{27} and performed 77,801 LP iterations before finding the optimal solution. In this case, the linearization process adds 35,397 continuous variables to the original 122. Clearly, the limitations of this approach for large systems are starting to show.

Residual Precision

In many cases, measurements contain biases and, therefore, they are eliminated. Once a redundant measurement is found to have a bias and is eliminated, the overall degree of redundancy is lowered and the precision of all variables decreases. It is desired that the precision of certain key variables remain above the specified threshold upon any deletion of gross errors. The *residual precision* is defined as the ability

Table 3. Flow Rates for Example 2

Stream	Flow	Stream	Flow	Stream	Flow
S_1	140	S_9	10	S_{17}	5
S_2	20	S_{10}	100	S_{18}	135
S_3	130	S_{11}	80	S_{19}	45
S_4	40	S_{12}	40	S_{20}	30
S_5	10	S_{13}	10	S_{21}	80
S_6	45	S_{14}	10	S_{22}	10
S_7	15	S_{15}	90	S_{23}	5
S_8	10	S_{16}	100	S_{24}	45

Table 4. Results of Example No. 3

	F_1	F_2	F_3	F_4	Cost \$
Sol. A	3%	3%	2%	—	2,900
Sol. B	3%	3%	—	2%	2,900
Sol. C	—	3%,3%	2%	—	2,900
Sol. D	—	3%,3%	—	2%	2,900

of the network to guarantee a certain level of precision in key selected variables when gross errors are detected and the measurements are eliminated (Bagajewicz, 1997). If the residual precision requested is equal to the precision $\psi_i^*(k) = \sigma_i^*$, the constraint on precision can be dropped, as the former contains the latter.

We now proceed to develop linear constraints. Consider a fixed set of measurements, that is, fixed μ. Assume now that there are r different combinations of arranging k deletions of measurements. Let $\boldsymbol{t}^r$ be a binary vector of the same dimension as the number of variables (n). An element of $\boldsymbol{t}^r$ is defined as equal to one if variable i is eliminated and equal to zero otherwise. If $k=1$, then there are n such vectors, and, in general, there are as many as combinations of k elements taken from a pool of n elements. The superscript r refers to each one of those combinations. The vectors $\boldsymbol{t}^r$ are therefore parameters of the model and have to satisfy the following equations

$$\sum_{i=1}^{n} t_i^r = k \tag{21}$$

This condition indicates that only k elements of $\boldsymbol{t}^r$ are nonzero, something that is obtained by construction. We now proceed to provide a way of calculating $\psi_i(k)$. Consider now a new binary matrix $\boldsymbol{q}^r$ constructed by eliminating from μ all measurements contained in $\boldsymbol{t}^r$, that is

$$q_{ij}^r = \begin{cases} \mu_{ij} - t_i^r & \text{if } \mu_{ij} = 1 \\ \mu_{ij} & \text{if } \mu_{ij} = 0 \\ \mu_{il_i^m} & \text{if } j = l_i^m \end{cases} \tag{22}$$

which in the absence of hardware redundancy can be rewritten as follows

$$\left.\begin{aligned} q_{ij}^r &= \mu_{ij}(1 - t_i^r)\ \forall j \neq l_i^m \\ q_{il_i^m}^r &= 1 - \sum_{j=1}^{l_i^m - 1} q_{ij}^r \end{aligned}\right\} \tag{23}$$

When $t_i^r = 0$, then $q_{ij}^r = \mu_{ij}$. When $t_i^r = 1$ and $\mu_{ij} = 1$ for some $j \neq l_i^m$, then $q_{ij}^r = 0$, $\forall j \neq l_i^m$ and $q_{il_i^m}^r = 1$, which is what is desired. Finally, when $t_i^r = 1$ and $\mu_{ili}^m = 1$, $q_{il_i^m}^r = 1$. In this last case, the combination $\boldsymbol{t}^r$ eliminates strictly less than k measurements and it does not affect results because it has lower (better) residual precision than the threshold. Indeed, the more measurements are eliminated, the worse the residual precision would be. Therefore, when k measurements are deleted, the new measurement vector is $\boldsymbol{q}^r$, and, therefore, a new precision for each variable $\sigma_i^r(q^r)$ is obtained. To guarantee this residual precision, the largest value for all possible values of $\boldsymbol{t}^r$ needs to be picked. Therefore

$$\psi_i(k) = \operatorname*{Max}_{\forall i} \{ \sigma_i^r(q^r) \} \leq \psi_i^*(k) \tag{24}$$

Equation 24 can be rewritten as follows

$$\sigma_i^r(q^r) \leq \psi_i^*(k)\ \forall r \tag{25}$$

Finally, $\sigma_i(q)$ is calculated using equations of the same type as Eq. 5 or Eq. 14. The constraints on residual precision explained above add $\lfloor(2n_J n_T + 1)n_I^2\rfloor(1 + n_R)$ new continuous variables after linearization (n_R is the maximum number of measurement elimination combinations based on the degree of residual precision k).

In the case of hardware redundancy, we define a binary matrix $\boldsymbol{T}_j^r$ with as many rows as the number of variables (m) and as many columns as the number of sensor candidates for each variable. The matrix now has to satisfy

$$\sum_{i=1}^{m} \sum_{j=1}^{l_i^m - 1} T_{ij}^r = k \tag{26}$$

which states that the k sensors that are eliminated from consideration could include more than one sensor per variable. Other more restrictive constraints requesting, for example, that the sensors eliminated should correspond to different variables, can be imposed.

The new binary vector $\boldsymbol{q}^r$ is obtained as follows

$$q_{ij}^r = \mu_{ij}(1 - T_{ij}^r)\ \forall j \neq l_i^m \tag{27}$$

whereas $q_{il_i^m}^r$ is defined as follows

$$\left.\begin{aligned} \sum_{t=1}^{l_i^m - 1} q_{it}^r q_{il_i^m}^r &= 0 \\ \sum_{t=1}^{l_i^m - 1} q_{it}^r &\geq (1 - q_{il_i^m}^r) \end{aligned}\right\} \tag{28}$$

This last system of equations assures that when $\sum_{t=1}^{l_i^m - 1} q_{it} = 0$, then $q_{il_i^m}^r = 1$, but when $\sum_{t=1}^{l_i^m - 1} q_{it} \geq 1$, then $q_{il_i^m}^r = 0$. Indeed, when $q_{it}^r = 0\ \forall t = 1, \ldots, (l_i^m - 1)$, then the first equation is trivially satisfied and the second forces $q_{il_i^m}^r = 1$. When $\sum_{t=1}^{l_i^m - 1} q_{it} \geq 1$, then the term $\sum_{t=1}^{l_i^m - 1} q_{it} q_{il_i^m}$ is forced to be zero by the first equation. In other words, $q_{il_i^m}^r = 0$. Equation 28 can be expressed linearly by

$$\left.\begin{aligned} \sum_{t=1}^{l_i^m - 1} \eta_{it}^r &= 0 \\ \sum_{t=1}^{l_i^m - 1} q_{it}^r &\geq (1 - q_{il_i^m}^r) \\ H(\eta_{it}^r, q_{it}^r, q_{il_i^m}^r) &\geq 0 \end{aligned}\right\} \tag{29}$$

For the case of hardware redundancy allowed, residual precision alone adds a total of $n_J n_T n_R + n_R(2n_I^2 n_J(n_T+2) + n_B[(n_T+2)+2n_I^2+(1+n_T)^2(1+2n_I^2)])$ new continuous variables.

Example 5: effect of residual precision

In example 1, it was found that two configurations with a cost of $C_T = 3{,}000$ satisfy the precision requirements. Now consider two residual precision constraints of order one ($k = 1$) as follows: $\psi_1^* = 2\%$ and $\psi_4^* = 3\%$. In this case, the optimal solution is $x_m = \{2\%, 3\%, 3\%, 3\%\}$, $C_T = 3{,}900$. CPLEX explored 71 nodes out of a total of 2^{12}, and performed 8,325 LP iterations before finding the optimal solution. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM was 15 s. If the residual precision is now required to be the same as the precision constraints, two different solutions with cost $C_T = 5{,}500$ are found: $x_m = \{1\%, 2\%, 2\%, --\}$ and $x_m = \{1\%, 2\%, --, 2\%\}$. In this last case, CPLEX explored 88 nodes out of a total of 2^{12}, and performed 11,961 LP iterations in 6 s on the same PC. The nonlinear model has 69 continuous variables, and the linearization process adds 500 new ones.

Example 6: effect of residual precision in retrofit problems

Consider example 2 with the same parameters and requirements. It was found the solution to be $x_m = \{F_2, F_4, F_8\}$ with a total cost of $C_T = 35$. Requesting the flow of streams S_1 to S_5 to be observable, with residual precision of order one and threshold of 30% ($\psi_i^* \le 30\%$, $i = 1, \ldots, 5$), then the optimal solution is $x_m = \{F_1, F_2, F_4, F_5, F_8\}$ with a total cost of $C_T = 79$. Notice that the requirement of 30% is somewhat equivalent to have them being observable. CPLEX explored 8 nodes of a total of 2^9 possible and performed 27,279 LP iterations. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is approximately 8 min. In this case, 148,225 continuous variables are added in the linearization process to the 3,602 of the original nonlinear problem. Although the size of this problem is much larger than that of example 4, the execution time is less than half.

Example 7: effect of residual precision with hardware redundancy

For the flowsheet of Figure 1, consider flowmeters of precision 3% and 2% at costs of 700 and 1,500, respectively. Precision is only required for variables F_1 and F_4 with $\sigma_1^* = 1.5\%$ and $\sigma_4^* = 2.0\%$. Hardware redundancy of second degree is allowed ($d_i = 2$). Residual precision constraints of order one ($k = 1$) are as follows: $\psi_1^* = 2\%$ and $\psi_4^* = 3\%$. The optimal solution has a total cost $C_T = 3{,}500$, and consists of locating one sensor of 3% precision in S_1, two sensors of 3% precision in S_2 and S_4, and no sensors in S_3. CPLEX explored 3 nodes of a total of 2^{20} possible and performed 27,279 LP iterations. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is approximately 5 min. There are 389 continuous variables in the nonlinear model and 12,164 are added in the linearization process.

Error Detectability

The ability of the network to detect k gross errors of a certain adimensional size κ_D or larger is called error detectability of order k (Bagajewicz, 1997). One can choose a threshold size of gross error for each variable δ_i^* and request those errors larger than this threshold to be detected by the network with a certain statistical confidence level. Let $\boldsymbol{r}$ be the vector of residuals of the system of equation. When the measurements follow a normal distribution with no gross errors, then the statistics $\boldsymbol{r}(\boldsymbol{A_R S_R A_R^T})^{-1}\boldsymbol{r}$ follow a central chi-square distribution with m degrees of freedom. However, in the presence of gross errors, it follows a noncentral chi-square distribution $\chi_m^2(\omega)$, where ω is the noncentrality parameter. When one gross error is present, ω is related to the gross error δ_i by the following expression (Madron, 1992)

$$\omega = \delta_i \frac{\left(s_{ii} - \hat{s}_{ii}\right)^{1/2}}{s_{ii}} \tag{30}$$

which is valid when no hardware redundancy is allowed. Concentrating on the error detectability of order $k = 1$, if one chooses one common dimensionless threshold value of gross error for all variables $\kappa_D = \delta_i^*/\sigma_i$, the error detectability criterion for variable i becomes

$$\omega \frac{\left(s_{ii}\right)^{1/2}}{\left(s_{ii} - \hat{s}_{ii}\right)^{1/2}} \le \kappa_D \tag{31}$$

Mathematical expressions capable of assessing the error detectability of larger order have not been developed yet. Since no hardware redundancy is allowed $\boldsymbol{s_i} = \sum_{t=1}^{l_i^m} \mu_{it}\sigma_{it}^2$ and Eq. 31 is expanded as follows

$$\left(\omega^2 - \kappa_D^2\right) \sum_{j=1}^{l_i^m} \mu_{ij}\sigma_{ij}^2 + \kappa_D^2 \hat{s}_i \le 0 \tag{32}$$

This last equation can only be applied to measured variables. The following equation is trivial for unmeasured variables

$$\left(\omega^2 - \kappa_D^2\right) \sum_{j=1}^{l_i^m - 1} \mu_{ij}\sigma_{ij}^2 + \kappa_D^2 \hat{s}_i \sum_{j=1}^{l_i^m - 1} \mu_{ij} \le 0 \tag{33}$$

By adding an auxiliary variable $\upsilon_{ij} = \mu_{ij}\hat{s}_i$, Eq. 33 can be converted into the following set of equations

$$\left.\begin{aligned} &\left(\omega^2 - \kappa_D^2\right) \sum_{j=1}^{l_i^m - 1} \mu_{ij}\sigma_{ij}^2 + \kappa_D^2 \sum_{j=1}^{l_i^m - 1} \upsilon_{ij} \le 0 \\ &G\left(\upsilon_{ij}, \hat{s}_i, \mu_{ij}, \Theta\right) \ge 0 \end{aligned}\right\} \tag{34}$$

where $\Theta > (\hat{s}_i)_{\max}$. The value of the noncentrality parameter ω is usually related to the level of confidence ϕ as follows: when ϕ increases, the noncentrality parameter decreases. This implies that higher gross errors will be predicted when the probabilities of finding them are larger.

The linearization of the constraints on error detectability adds only $(n_E n_T)$ continuous variables to the model (n_E is the subset of variables for which error detectability is required). When hardware redundancy is allowed, then Eq. 33 becomes

$$\left(\omega^2 - \kappa_D^2\right)\frac{\lambda_i \eta_i}{\sum\limits_{t=1}^{l_i^m} \mu_{it}\gamma_{it}} + \kappa_D^2 \hat{s}_i \eta_i \leq 0 \tag{35}$$

where η_i was introduced in order for the constraint to be trivially satisfied for unmeasured variables. Variable η_i is defined as follows

$$\left.\begin{array}{l} \sum\limits_{t=1}^{l_i^m-1} \mu_{it} = \eta_i \sum\limits_{t=1}^{l_i^m-1} \mu_{it} \\ \sum\limits_{t=1}^{l_i^m-1} \mu_{it} \geq \eta_i \\ \eta_i \geq 0 \end{array}\right\} \tag{36}$$

Substituting ρ_i from Eq. 14 in Eq. 35, one obtains

$$\left(\omega^2 - \kappa_D^2\right)\lambda_i\, \rho_i \eta_i + \kappa_D^2 \hat{s}_i \eta_i \leq 0 \tag{37}$$

Finally, the product $\eta_i \mu_{it}$ in Eq. 36, and the products $\rho_i \eta_i$ and $\hat{s}_i \eta_i$ in Eq. 37 are linearized by using the Glover (1975) equations. When hardware redundancy is allowed, error detectability constraints add $n_E(n_T+4)$ continuous variables.

Example 8: effect of error detectability

Consider adding error detectability constraints for all measurements of example 1 with $\kappa_D = 4.0$ and $\phi = 50\%$. Out of only four feasible solutions, the global optimum in this case is $x_m = \{1\%, 3\%, 2\%, 2\%\}$, with a total cost of $C_T = 6{,}300$. CPLEX explored 45 nodes out of 2^{12} possible and performed 1,989 LP iterations. The execution time on a Pentium III PC, 1.2 GHz, 1 G RAM is less than 1 s. For this particular example, the lowest value possible for κ_D is 3.386, that is, no feasible solution can be found for a lower value for κ_D. A total of 112 new continuous variables were added to the original five of the nonlinear formulation.

Example 9: effect of error detectability in retrofit problems

Consider example 2 with the same parameters and requirements. When error detectability for streams S_1 and S_3 (the larger flow rates) is required with $\kappa_D = 3.9$ and $\phi = 50\%$, the optimal solution is $x_m = \{F_1, F_2, F_3, F_4, F_5, F_8\}$ with a total cost of $C_T = 92$. CPLEX explored only one node out of 2^9 possible and performed 459 iterations. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is 8 s. This short execution time is because there are very few feasible solutions to the problem. The nonlinear model has 122 continuous variables, and the linearization process adds 5,931 new ones.

Example 10: error detectability with hardware redundancy

Consider again the flowsheet of Figure 2 with the same flow rates and costs as in example 2. As in example 3, precision is required to be at least of 2% for the flow rates of streams 1–5, and as many as two devices can be measuring a single variable ($d_i = 2$). Error detectability is requested for streams S_1 and S_3 (the larger flow rates) with $\kappa_D = 4$ and $\phi = 50\%$. The optimal solution has a total cost of $C_T = 168$ with devices located in all streams and an additional device located in streams S_2, S_4 and S_5. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is 4 min. CPLEX explored 47 nodes out of a total of 2^{27} and performed 77,801 LP iterations. The nonlinear model has 124 continuous variables, and the linearization process adds 35,407 new ones; of which only 12 come from the error detectability constraints. The optimal solution is the same as that of example 4, because the constraints on precision are stricter than the ones on error detectability. The execution time, nevertheless, decreases considerably in comparsion to example 4. This is due to the small number of feasible solutions of the problem.

Error Resilience

When a gross error of a certain magnitude in any variable occurs and is not detected, a smearing of this corrupted data takes place when reconciliation is performed. The ability of the network to limit the smearing effect of k undetected gross errors of a certain adimensional size κ_R or lower is called gross error resiliency or order k (Bagajewicz, 1997). Let m^r be a vector indicating one of the many possible positions of k gross errors, that is, $m_j^r = 1$ indicates a gross error is present in variable F_j. Let M_δ be the set containing all such vectors. Thus, for example, when $k = 1$, then $M_\delta = \{e_1, e_2, \ldots, e_m\}$. When $k = 2$, the $M_\delta = \{e_1, e_2, \ldots, e_m, e_1 + e_2, e_1 + e_3, \ldots, e_1 + e_m, e_2 + e_3, e_2 + e_4, \ldots, e_2 + e_m, \ldots, e_{m-1} - e_m\}$ and so on, much in the same fashion as in residual precision. Let δ^r be the vector of gross errors corresponding to m^r. Then, δ^r is given by

$$\delta_i^r = \kappa_R \sigma_i m_i^r \tag{38}$$

For convenience, we introduce the following notation: $m^r = \sum\limits_{j \in \pi_r} e_j$ where $\pi_r = \{j/m_j^r = 1\}$. Thus, for example, if $m^r = e_2 + e_3$, then $\pi_r = \{2{,}3\}$. With this notation δ^r is now given by

$$\delta^r = \kappa_R \sum_{k \in \pi_r} \sigma_k e_k \tag{39}$$

Let $\Delta \tilde{\boldsymbol{F}}^r$ be the change of the reconciled value of the vector of flows $\tilde{\boldsymbol{F}}$ when a set of gross errors δ^r of adimensional size κ_R are present in the network. Thus, (Bagajewicz, 1997)

$$\Delta \tilde{\boldsymbol{F}}^r = -\boldsymbol{S}\boldsymbol{A}^T\left(\boldsymbol{A}\boldsymbol{S}\boldsymbol{A}^T\right)^{-1}\boldsymbol{A}\boldsymbol{\delta}^r = \boldsymbol{H}\boldsymbol{\delta}^r \tag{40}$$

where the matrix $\boldsymbol{H} = -\boldsymbol{S}\boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{S}\boldsymbol{A}^T)^{-1}\boldsymbol{A}$ is introduced. Then, the impact of the gross error vector on each element of the estimation vector is given by the rows of the matrix $\boldsymbol{H}$, namely $\Delta \tilde{F}_i^r = h_i^T\, \delta^r$, where $h_i^T = e_i^T \boldsymbol{H}$ is the ith row of $\boldsymbol{H}$. Then, if a desired level of resiliency is fixed for variable F_i by requesting $\Delta \tilde{F}_i$ to be lower than a certain threshold R_i^*, the resilience criterion becomes

$$\underset{\forall r}{\mathrm{Max}} |\Delta \tilde{F}_i^r| = \underset{\forall r}{\mathrm{Max}} \left| e_i^T \boldsymbol{H} \sum_{k \in \pi_r} \sigma_k \boldsymbol{e}_k \right| \kappa_R \leq R_i^* \quad \forall i \in M_R \tag{41}$$

where M_R is the subset of M for which error resilience is required. The resilience criterion changes depending on the definition adopted for $\boldsymbol{\delta_R}$. For example, Chmielewski (2001) showed that if a normalized 2-norm is chosen for $\boldsymbol{\delta_R}$, then the resilience constraint would have a similar effect as bounding the error covariance.

Expression 41 is nonlinear, both because the absolute value space and Max operators are nonlinear and also because of matrix $\boldsymbol{H}$, which is constructed using an inverse. Using the auxiliary matrix $\boldsymbol{P}$ (inverse of matrix $\mathbf{ASA}^T$), for which defining equations have been introduced before, the following system of equations results

$$\left.\begin{array}{l} |\Delta F_i^r| \leq R_i^* \\ \Delta F_i^r = -\kappa_R \sum\limits_{j\in\pi_r} \sum\limits_{t=1}^{l_i^m-1} H_{ij}\,\mu_{jt}\,\sigma_{jt} \end{array}\right\} \forall i \in M_R, \forall r \in M_\delta \quad (42)$$

Expanding H_{ij} one obtains the following expression

$$\Delta F_{ij}^r = -\kappa_R \sum_{j\in\pi_r} \sum_{t=1}^{l_i^m-1} \sum_{p=1}^{l_i^m} \sum_{d=1}^{n} \sum_{k=1}^{n} \mu_{jt}\sigma_{jt}\sigma_{ip}\mu_{ip}a_{di}p_{dk}a_{kj} \quad (43)$$

Therefore, one can substitute Eq. 41 by the following set of equations

$$\left.\begin{array}{l} R_i^* \geq \Delta F_i^r \\ R_i^* \geq -\Delta F_i^r \end{array}\right\} \quad (44)$$

In order to eliminate all the nonlinearities, the products $\mu_{jt}\,\mu_{ip}p_{dk}$ are substituted by systems of linear inequalities by using the Glover (1975) transformation. Thus, we let $x_{jtipdk} = y_{jtip}p_{dk}$ and $y_{jtip} = \mu_{jt}\,\mu_{ip}$, which can be expressed linearly by writing

$$\left.\begin{array}{l} x_{jtipdk} = p_{dk,1}y_{jtip} - p_{dk,2}y_{jtip} = x_{jtipdk,1} - x_{jtipdk,2} \\ G(x_{jtipdk,1}, p_{dk,1}, y_{jtip}, \Lambda_1) \geq 0 \\ G(x_{jtipdk,2}, p_{dk,2}, y_{jtip}, \Lambda_2) \geq 0 \\ H(y_{jtip}, \mu_{it}, \mu_{ip}) \geq 0 \end{array}\right\} \quad (45)$$

The linearization of the constraints on error resilience adds $[n_J n_T^2 n_E(n_I^2+1)]$ continuous variables to the model. When hardware redundancy is used, then we define a matrix Δ^r containing as many rows as variables and as many columns as sensor candidates. This matrix is given by

$$\Delta_{ij}^r = \kappa_R \sum_{(k_1,k_2)\in\Pi_r} \mu_{k_1k_2}\sigma_{k_1k_2}\left(E_{k_1k_2}\right)_{ij} \quad (46)$$

where Π_r is the set of matrix positions coordinates that say which matrices $E_{k_1k_2}$ contribute to Δ^r. The matrix $E_{k_1k_2}$ also contains as many rows as variables and as many columns as sensor candidates; and has only one element equal one and the rest equal zero. It is defined as follows

$$\boldsymbol{E}_{k_1k_2} = \left\{\left(E_{k_1k_2}\right)_{ij} = 1/k_1 = i, k_2 = j\right\}$$

$$\forall k_1 = 1, \ldots, m; \forall k_2 = 1, \ldots, l_i^m - 1 \quad (47)$$

Assuming that gross errors will be referred to no more than a single measurement or device per variable, the vectors of gross errors δ^r can be expressed as follows

$$\delta_i^r = \sum_{j=1}^{l_i^m-1} \Delta_{ij}^r = \kappa_R \sum_{j=1}^{l_i^m-1} \sum_{(k_1,k_2)\in\Pi_r} \mu_{k_1k_2}\sigma_{k_1k_2}\left(E_{k_1k_2}\right)_{ij} \quad (48)$$

Thus, when hardware redundancy is allowed, the resilience constraints are as follows

$$\left.\begin{array}{l} \Delta F_i^r = -\kappa_R \sum\limits_{k=1}^{n} \sum\limits_{d=1}^{n} \sum\limits_{h=1}^{m} \sum\limits_{j=1}^{l_i^m-1} \sum\limits_{(k_1,k_2)\in\Pi_r} \dfrac{a_{ki}a_{dh}\sigma_{k_1k_2}p_{kd}\,\mu_{k_1k_2}\left(E_{k_1k_2}\right)_{hj}\lambda_i}{\sum\limits_{t=1}^{l_i^m} \mu_{it}\gamma_{it}} \\ R_i^* \geq \Delta F_i^r \\ R_i^* \geq -\Delta F_i^r \end{array}\right\} \forall i \in M_R, r \in M_\delta \quad (49)$$

To linearize this last equation, the variable $\tau_{ikdk_1k_2}$ is introduced. That new variable is defined as follows

$$\tau_{ikdk_1k_2} \sum_{t=1}^{l_i^m-1} \mu_{it}\gamma_{it} = p_{kd}\,\mu_{k_1k_2} \quad (50)$$

The products $\tau_{ikdk_1k_2}\,\mu_{it}$ and $p_{kd}\,\mu_{k_1k_2}$ are replaced by linear systems of inequalities as in later sections of this article. Finally, the resilience constraint can be written in this way

$$\left.\begin{array}{l} \Delta F_i^r = -\kappa_R \sum\limits_{k=1}^{n} \sum\limits_{d=1}^{n} \sum\limits_{h=1}^{m} \sum\limits_{j=1}^{l_i^m-1} \sum\limits_{(k_1,k_2)\in\Pi_r} a_{ki}a_{dh}\sigma_{k_1k_2}\left(E_{k_1k_2}\right)_{hj}\lambda_i\tau_{ikdk_1k_2} \\ R_i^* \geq \Delta F_i^r \\ R_i^* \geq -\Delta F_i^r \end{array}\right\} \forall i \in M_R, r \in M_\delta \quad (51)$$

When hardware redundancy is allowed, error resilience constraints add $n_E n_I^2 n_J n_T(n_T+1)$ continuous variables.

Example 11: effect of error resilience

Assume that for example 1, error detectability is requested at a level of 4.0 times the standard deviation of the measurement for all measurements ($\kappa_D = 3.9$, $\phi = 50\%$), and resilience is requested at a level of 3 times the standard deviation for all measurements ($\kappa_R = 3$). In this case, the optimal solution is $x_m = \{1\%, 3\%, 2\%, 2\%\}$, with a total cost of $C_T =$ 6,300. CPLEX explored 72 out of 2^{12} possible nodes and performed 11,619 LP iterations. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is 12 s. A total of 1,408 new continuous variables were added to the original five of the nonlinear formulation. Of the new variables, 1,296 come from the error resilience constraints.

Example 12: effect of error resilience in retrofit problem

Consider the network of Figure 2. This last example includes all the constraints explained in the paper (precision, residual precision, error detectability and error resilience). Precision and residual precision are required as on example 6, $\sigma_1^* = 1.5\%$, $\sigma_4^* = 2.0\%$ and $\psi_i^* \leq 100\%$, $i = 1, \ldots, 5$; error detectability is required for streams one and three, larger flow rates, with $\kappa_D = 5.0$ ($\phi = 50\%$), and error resilience is requested at a level of 3 times the standard deviation for streams one and three (S_1 and S_3). The optimal solution is $x_m = \{F_1, F_2, F_3, F_4, F_5\}$ with a total cost of $C_T = 86$. CPLEX explored 10 out of 2^9 possible nodes and performed 38,557 LP iterations. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is about 15 min. In this case, the size of the problem is one of the main reasons for the long execution time. Indeed, the number of new continuous variables added for linearization is 159,891, compared to the 3,602 original variables from the nonlinear problem. Residual precision constraints alone add 145,776 continuous variables, while error resilience constraints add 11,664. It is quite clear that larger problems will start reaching the limits of existing solvers.

Example 13: effect of error resilience with hardware redundancy

Consider example 10 but now adding error resilience constraints. As in example 12, error resilience is requested at a level of 3 times the standard deviation for streams one and three (S_1 and S_3). In this case, the optimal solution has a total cost of $C_T = 200$ with devices located in all streams and an additional device located in streams S_1 through S_5. The execution time on a Pentium III PC, 1.2 GHz, 1 Gb RAM is slightly more than 2 h. CPLEX explored 270 nodes out of a total of 2^{27} and performed 1,004,488 LP iterations. Resilience constraints add 69,696 new continuous varibles to make a total of 105,103 variables added in the linearization process. The amount of equations involved in the optimization problem is one of the main reasons for the increase on the execution time. However, example 12 is larger than this one and its execution time is much shorter.

Conclusions

The MILP formulation presented in this article provides a useful tool to design and upgrade instrumentation networks. The first contribution of this work is the development of a model whose solution procedure does not depend on the particular configuration of the network. In addition, the model is easily implemented and works quite well for the design of small and medium size networks and for the upgrade of relatively large networks. Since all alternative models share the same limitations, the challenge remains for solving large size problems efficiently.

Acknowledgment

Financial support from NSF through grant CTS-0075814 is gratefully acknowledged.

Notation

a_{ij} = elements of matrix A
A = incidence matrix
A_R = incidence matrix for redundant systems
c_{ij} = cost of sensor j for stream i
C_T = total cost of the network
d_i = degree of hardware redundancy
E = number of streams for which error detectability is required
e_i = unit vector with $(e_i)_i = 1$ and $(e_i)_i = 0\ \forall j \neq i$
E_{ij} = unit matrix with $(E_{ij})_{ij} = 1$ and $(E_{ij})_{lm} = 0\ \forall(l,m) \neq (i,j)$
F_i = flow rate of stream i
$\boldsymbol{I}$ = unit matrix
l_i^m = number of candidates for sensor devices
M = set of all variables
M_p = subset of M for which precision or residual precision is required
M_r = subset of M for which error resilience is required
n_B = number of streams for which precision is required
n_E = number of variables for which error detectability is required
n_I = number of balance equations
n_J = number of streams in the system
n_R = maximum number of measurement elimination combinations for residual precision
n_T = number of sensor options
P = inverse of matrix ASA^T
p_{ij} = elements of matrix P
s_{ii} = variance of variable i before data reconciliation
$\hat{s}$ = variance of estimates from data reconciliation
$\tilde{s}$ = variance of estimates obtained from generalized Eq. 2
s_i^* = threshold for the variance of variable i
S = variance matrix
$\tilde{S}$ = variance of reconciled values
S_R = variance of redundant measured variables
S_O = variance of observable "unmeasured" variables
S_i = stream i
t_{ij}^k = binary variable indicating whether a measurement j is eliminated (1) or not (0) in variable i
T_{ij}^k = extension of t_{ij}^k for the case when hardware redundancy is allowed
x_m = measurements vector

Greek letters

δ_{kj} = Kroenecker function
δ_i = value for the size of gross error in variable i
κ_D = adimensional threshold value for the error detectability
μ_{ij} = binary variable stating whether a sensor j is located (1) or not (0) in variable i
σ_{ij} = standard deviation of sensor j for variable i
ϕ = probability of the existence of an unidentified gross error
ψ_i^* = threshold for the residual precision of variable i
ψ_i = residual precision of variable i
ω = noncentrality parameter

Literature Cited

Bagajewicz, M., *Design and Upgrade of Process Plant Instrumentation*, Technomic, PA (2000).

Bagajewicz, M., "Optimal Sensor Location in Process Plants," *AIChE J.*, **43**, 2300 (1997).

Bagajewicz, M., and M. Sánchez, "Sensor Network Design and Upgrade for Plant Parameter Estimation," *Comp. Chem. Eng.*, Supp., **23**, S593 (1999).

Bagajewicz, M., and M. Sánchez, "Cost-Optimal Design of Reliable Sensor Networks," *Comput. Chem. Eng.*, **23**, 11/12, 1757 (2000a).

Bagajewicz, M., and M. Sánchez, "Reallocation and Upgrade of Instrumentation in Process Plants," *Comput. Chem. Eng.*, **24** (8), 1961 (2000b).

Chmielewski, D., "Convex Methods in Sensor Placement," *Proc. of* 4*th IFAC Workshop on On-Line Fault Detection & Supervision in the Chemical Process Industries*, Seoul, Korea (Jun. 8–9, 2000).

Chmielewski, D., T. E. Palmer, and V. Manousiouthakis, "Cost Optimal Retrofit of Sensor Networks with Loss Estimation Accuracy," AIChE Meeting, Dallas (1999).

Glover, F., "Improved Linear Integer Programming Formulations of Nonlinear Integer Problems," *Manage. Sci.*, **22**, 455 (1975).

Madron, F., and V. Veverka, "Optimal Selection of Measuring Points in Complex Plants By Linear Models," *AIChE J.*, **38**, 227 (1992).

Madron, F., *Process Plant Performance. Measurement and Data Processing for Optimization and Retrofits*, Ellis Horwood, West Sussex, U.K. (1992).

Meyer, M. J., L. Le Lann, B. Koehret, and M. Enjalbert, "Optimal Selection of Sensor Location on a Complex Plant Using a Graph Oriented Approach," *Comput. & Chem. Eng.*, **18**, Suppl., S535 (1994).

Narasimhan, S., and C. Jordache, *Data Reconciliation & Gross Error Detection. An Intelligent Use of Process Data*, Gulf Publishing Company, Houston, TX (2000).

Sánchez, M., and M. Bagajewicz, "On the Impact of Corrective Maintenance in the Design of Sensor Networks," *Ind. and Eng. Chemistry Res.*, **39** (4), 977 (2000).

Tuy, H., "Convex Programs with an Additional Reverse Convex Constraint," *JOTA*, **52**, 463 (1987).

Appendix

In this section, we present mathematical proof of the claim that the equation of variance of the reconciled variables can be applied as the expression of variance for all the variables (regardless of their observability classification) when the standard deviation of the unmeasured variables is assumed to tend to infinity.

Claim

The variance of all variables, independently of their observability classification, can be expressed uniquely through the expression $\tilde{\boldsymbol{S}} = \boldsymbol{S} - \boldsymbol{S}\cdot\boldsymbol{A}^{\boldsymbol{T}}(\boldsymbol{A}\cdot\boldsymbol{S}\cdot\boldsymbol{A}^{\boldsymbol{T}})^{-1}\ \boldsymbol{A}\cdot\boldsymbol{S}$ when the instrument variance of the unmeasured variables tends to infinity. That is

$$\hat{\boldsymbol{S}} \overset{?}{=} \lim_{S_{UM}\to\infty} \tilde{\boldsymbol{S}} = \lim_{S_{UM}\to\infty} \left(\boldsymbol{S} - \boldsymbol{S}\cdot\boldsymbol{A}^{\boldsymbol{T}}\left(\boldsymbol{A}\cdot\boldsymbol{S}\cdot\boldsymbol{A}^{\boldsymbol{T}}\right)^{-1}\boldsymbol{A}\cdot\boldsymbol{S}\right) \quad \text{(A1)}$$

Proof

There are four different types of variables: measured redundant variables, measured nonredundant variables, unmeasured observable variables and unmeasured unobservable variables. Each one of the these cases will be treated separately.

Redundant Variables. Consider the following mass balance matrix for a system with observable and redundant variables only (without loss of generality, the canonical form is used)

$$\boldsymbol{A} = \begin{vmatrix} \boldsymbol{I} & \boldsymbol{A}_1 \\ 0 & \boldsymbol{A}_{\boldsymbol{R}} \end{vmatrix} \Rightarrow \boldsymbol{S} = \begin{vmatrix} \boldsymbol{S}_O & 0 \\ 0 & \boldsymbol{S}_{\boldsymbol{R}} \end{vmatrix} \quad \text{(A2)}$$

Let $(\boldsymbol{A}\cdot\boldsymbol{S}\cdot\boldsymbol{A}^{\boldsymbol{T}-1} = \boldsymbol{P} = \begin{vmatrix} \boldsymbol{P}_1 & \boldsymbol{P}_2 \\ \boldsymbol{P}_3 & \boldsymbol{P}_4 \end{vmatrix}$. Therefore, matrix $\tilde{\boldsymbol{S}}$ becomes: $\tilde{\boldsymbol{S}} = \boldsymbol{S} - \boldsymbol{S}\cdot\boldsymbol{A}^{\boldsymbol{T}}\cdot\boldsymbol{P}\cdot\boldsymbol{A}\cdot\boldsymbol{S}$. Then, including the matrices $\boldsymbol{A}$ and $\boldsymbol{S}$ in the expression for $\tilde{\boldsymbol{S}}$, we obtain

$$\left.\begin{aligned} &\tilde{\boldsymbol{S}}_O{=}\boldsymbol{S}_O - \boldsymbol{S}_O\boldsymbol{P}_1\boldsymbol{S}_O \\ &\tilde{\boldsymbol{S}}_{\boldsymbol{R}} = \boldsymbol{S}_{\boldsymbol{R}}{-}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_1^{\boldsymbol{T}}(\boldsymbol{P}_1\boldsymbol{A}_1{+}\boldsymbol{P}_2\boldsymbol{A}_{\boldsymbol{R}})\boldsymbol{S}_{\boldsymbol{R}}{-}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}(\boldsymbol{P}_3\boldsymbol{A}_1 + \boldsymbol{P}_4\boldsymbol{A}_{\boldsymbol{R}})\boldsymbol{S}_{\boldsymbol{R}} \end{aligned}\right\} \quad \text{(A3)}$$

Now expressions for $\boldsymbol{P}_1$, $\boldsymbol{P}_2$, $\boldsymbol{P}_3$ and $\boldsymbol{P}_4$ in terms of the elements of the matrices $\boldsymbol{A}$ and $\boldsymbol{S}$ are needed. Consider the Householder formula

If

$$\boldsymbol{A} = \begin{vmatrix} \boldsymbol{P} & \boldsymbol{Q} \\ \boldsymbol{R} & \boldsymbol{S} \end{vmatrix}$$

Then

$$\boldsymbol{A}^{-1} = \begin{vmatrix} \boldsymbol{E} & -\boldsymbol{EF} \\ -\boldsymbol{GE} & \boldsymbol{S}^{-1} + \boldsymbol{GEF} \end{vmatrix} \quad \text{(A4)}$$

where

$$\left.\begin{aligned} &\boldsymbol{F} = \boldsymbol{QS}^{-1} \\ &\boldsymbol{G} = \boldsymbol{S}^{-1}\boldsymbol{R} \\ &\boldsymbol{E} = (\boldsymbol{P} - \boldsymbol{QG})^{-1} \end{aligned}\right\} \quad \text{(A5)}$$

Therefore

$$P = \begin{vmatrix} P_1 & P_2 \\ P_3 & P_4 \end{vmatrix} = \begin{vmatrix} S_O + A_1S_RA_1^T & A_1S_RA_R^T \\ A_RS_RA_1^T & A_RS_RA_R^T \end{vmatrix}^{-1} \quad \text{(A6)}$$

Applying the Householder formula we can obtain expressions for the elements of the matrix $\boldsymbol{P}$

$$\begin{aligned} &\boldsymbol{F} = \left(\boldsymbol{A}_1\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}\right)\left(\boldsymbol{A}_{\boldsymbol{R}}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}\right)^{-1} \\ &\boldsymbol{G} = \left(\boldsymbol{A}_{\boldsymbol{R}}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}\right)^{-1}\left(\boldsymbol{A}_{\boldsymbol{R}}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_1^{\boldsymbol{T}}\right) \\ &\boldsymbol{E} = \left(\boldsymbol{S}_O + \boldsymbol{A}_1\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_1^{\boldsymbol{T}} - \boldsymbol{A}_1\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}\left(\boldsymbol{A}_{\boldsymbol{R}}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}\right)^{-1}\boldsymbol{A}_{\boldsymbol{R}}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_1^{\boldsymbol{T}}\right)^{-1} \end{aligned} \quad \text{(A7)}$$

and

$$\left.\begin{aligned} &\boldsymbol{P}_1{=}\boldsymbol{E}{=}\left(\boldsymbol{S}_O + \boldsymbol{A}_1\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_1^{\boldsymbol{T}}{-}\boldsymbol{A}_1\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}\left(\boldsymbol{A}_{\boldsymbol{R}}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_{\boldsymbol{R}}^{\boldsymbol{T}}\right)^{-1}\boldsymbol{A}_{\boldsymbol{R}}\boldsymbol{S}_{\boldsymbol{R}}\boldsymbol{A}_1^{\boldsymbol{T}}\right)^{-1} \\ &\boldsymbol{P}_2 = -\boldsymbol{EF} \\ &\boldsymbol{P}_3 = -\boldsymbol{GE} \\ &\boldsymbol{P}_4 = \boldsymbol{S}^{-1} + \boldsymbol{GEF} \end{aligned}\right\} \quad \text{(A8)}$$

One can see that

$$\lim_{S_O \to \infty} P = \begin{vmatrix} 0 & 0 \\ 0 & \left(A_R S_R A_R^T\right) \end{vmatrix} \qquad \text{(A9)}$$

This implies that

$$\lim_{S_{UM} \to \infty} \tilde{S}_R = S_R - S_R \cdot A_R^T \left(A_R \cdot S_R \cdot A_R^T\right)^{-1} A_R \cdot S_R = \hat{S}_R \qquad \text{(A10)}$$

which is what wanted to be proven.

Nonredundant Variables. The case of nonredundant variables is completely equivalent to having redundant variables with A_R equal zero. Therefore, evaluating $\tilde{S}_R$ from Eq. A3 when $A_R = 0$, one obtains

$$\tilde{S}_{NR} = \tilde{S}_R|_{A_R = 0} = S_{NR} - S_{NR} A_1^T P_1 A_1 S_{NR} \qquad \text{(A11)}$$

Now, taking the limit when S_O tends to infinity of the last expression, we can see that the second term which includes P_1 goes to zero, and, therefore, Eq. A13 reduces as follows

$$\lim_{S_O \to \infty} \tilde{S}_{NR} = S_{NR} = \hat{S}_{NR} \qquad \text{(A12)}$$

Observable Variables. Now, what is left to prove is that the equation also works for unmeasured variables. For observable variables, the variance is given by the following expression

$$\hat{S}_O = A_{RO} \hat{S}_R A_{RO}^T + A_{NRO} \hat{S}_{NR} A_{NRO}^T \qquad \text{(A13)}$$

Let us first consider the case when only redundant and observable variables are present. This means that in Eq. A13, A_{NRO} is equal zero and $A_{RO} = A_1$. Using P_1 from Eq. A8 in Eq. A3, one gets

$$\tilde{S}_O = S_O - S_O \Big(S_O + A_1 S_R A_1^T - A_1 S_R A_R^T \left(A_R S_R A_R^T\right)^{-1} \times A_R S_R A_1^T \Big)^{-1} S_O$$

$$\tilde{S}_O = S_O - \Big[S_O^{-1} + S_O^{-1} \Big(A_1 S_R A_1^T - A_1 S_R A_R^T \left(A_R S_R A_R^T\right)^{-1} A_R S_R A_1^T \Big) S_O^{-1} \Big]^{-1}$$

$$\Rightarrow \tilde{S}_O = S_O - \left(S_O^{-1} + S_O^{-1} A_1 \hat{S}_R A_1^T S_O^{-1} \right)^{-1} \qquad \text{(A14)}$$

Looking at Eq. A13 and comparing it with Eq. A14, one can see that in order to finish the proof, the following needs to be true

$$S_O - \left(S_O^{-1} + S_O^{-1} A_1 \hat{S}_R A_1^T S_O^{-1} \right)^{-1} \overset{?}{=} A_1 \hat{S}_R A_1^T \qquad \text{(A15)}$$

However, that is only true if and only if the following is true

$$\left(S_O - A_1 \hat{S}_R A_1^T \right) \left(S_O^{-1} + S_O^{-1} A_1 \hat{S}_R A_1^T S_O^{-1} \right) \overset{?}{=} I$$

$$I - A_1 \hat{S}_R A_1^T S_O^{-1} A_1 \hat{S}_R A_1^T S_O^{-1} \overset{?}{=} I \qquad \text{(A16)}$$

Now, taking the limit when S_O tends to infinity of the last expression, one can see that the second term goes to zero, and, therefore, the equality holds. This means that, when S_O tends to infinity, Eq. A15 also holds, and thus

$$\lim_{S_O \to \infty} \tilde{S}_O = A_1 \hat{S}_R A_1^T = \hat{S}_O \qquad \text{(A17)}$$

For the case when only nonredundant variables are present, one only has to evaluate Eq. A17 at $A_R = 0$, similar to what we did previously to obtain Eq. A12. The general case is when both nonredundant and redundant variables are present. For that, split the matrices A_1 and $\hat{S}_R$ assuming that some part will represent nonredundant variables, and another the redundant ones

$$\left. \begin{aligned} A_1 &= |A_{RO} \;\; A_{NRO}| \\ \hat{S}_R &= \begin{vmatrix} \hat{S}_R & 0 \\ 0 & \hat{S}_{NR} \end{vmatrix} \end{aligned} \right\} \qquad \text{(A18)}$$

Substituting these expressions in Eq. A17 we finally obtain

$$\lim_{S_O \to \infty} \tilde{S}_O = \hat{S}_O = A_{RO} \hat{S}_R A_{RO}^T + A_{NRO} \hat{S}_{NR} A_{NRO}^T \qquad \text{(A19)}$$

Unobservable Variables. In the case of unobservable variables, the reconciled variance must tend to infinity when the standard deviation of the instruments of unmeasured variables is assumed to tend to infinity, that is

$$\hat{S}_{UO} = \lim_{S_{UM} \to \infty} \Big(S_{UO} - S_{UO} \cdot A_{UO}^T \left(A_{UO} \cdot S_{UO} \cdot A_{UO}^T \right)^{-1} \times A_{UO} \cdot S_{UO} \Big) \overset{?}{=} (\to \infty) \qquad \text{(A20)}$$

where S_{UO} is equal to S_{UM}. By doing rearrangements to the equation of the reconciled variance $\tilde{S}_{UO} = S_{UO} - S_{UO} \cdot A_{UO}^T (A_{UO} \cdot S_{UO} \cdot A_{UO}^T)^{-1} A_{UO} \cdot S_{UO}$, one can arrive to

$$\left. \begin{aligned} \tilde{S}_{UO} &= z \cdot S_{UO} \cdot z^T \\ z &= \left[I - S_{UO} \cdot A_{UO}^T \left(A_{UO} \cdot S_{UO} \cdot A_{UO}^T \right)^{-1} A_{UO} \right] \end{aligned} \right\} \qquad \text{(A21)}$$

Let us analyze the order of each term of the equation above. Since S_{UO} tends to infinity, the terms $S_{UO} \cdot A_{UO}^T$ and $(A_{UO} \cdot S_{UO} \cdot A_{UO}^{\mathrm{T}})^{-1}$ tend to infinity and to zero, respectively. Thus, the overall term $S_{UO} \cdot A_{UO}^T (A_{UO} \cdot S_{UO} \cdot A_{UO}^T)^{-1} A_{UO}$ has an order of one, and z is also of order one. Finally, by looking at $\tilde{S}_{UO} = z \cdot S_{UO} \cdot z^T$ we can conclude that $\hat{S}_{UO} = \lim_{S_{UO} \to \infty} \tilde{S}_{UO}$ also tends to infinity

$$\hat{S}_{UO} \to \infty \qquad \text{(A22)}$$

This last result concludes the proof.

Manuscript received Sept. 20, 2001, and revision received Mar. 25, 2002.