

# A Novel Adaptive Nonlinear Dynamic Data Reconciliation and Gross Error Detection Method

James H. Taylor and Mazyar B. Laylabadi

**Abstract**—Data reconciliation is a well-known method in on-line process control engineering aimed at estimating the true values of corrupted measurements under constraints. Most nonlinear dynamic data reconciliation methods have studied cases where the input variables are constant over relatively long periods of time separated by simple step changes (e.g., set-point changes). While this scenario is not uncommon in process control, it imposes strong limitations on a method's applicability. In this paper a novel adaptive nonlinear dynamic data reconciliation algorithm is presented that extends the method presented by Laylabadi and Taylor [1] to the cases where the input variables are ramps or slow sinusoidal functions or, for that matter, any slow, smooth variation.

## I. INTRODUCTION

Data reconciliation (DR) is a well-known method in on-line process control engineering aimed at estimating the true values of corrupted measurements under constraints [2], [3], [4], [5]. Most nonlinear dynamic data reconciliation (NDDR) techniques today are based on two major assumptions: 1) availability of known dynamic and statistical models, and 2) gross-error-free measurements. These two assumptions limit the application of these NDDR algorithms, as they are often not true in reality. In this paper first a new gross error detection (GED) and statistical model identification approach is developed and combined with the original NDDR algorithm to remove these barriers.

Next, the dynamic behavior of existing NDDR algorithms is studied. Most NDDR methods have been applied to cases where the input variables are comprised of simple step set-point changes. In this paper, however, a novel adaptive nonlinear dynamic data reconciliation (ANDDR) algorithm is presented that extends the applicability of the method presented by Laylabadi and Taylor [1] to the cases where the input variables are ramps or slow sinusoids. The proposed package has been successfully applied to the continuously stirred tank reactor (CSTR) model cited commonly in the literature.

The combined ANDDR and GED together with the enhanced dynamic tracking feature produce a package that is suitable for most chemical process applications. This package is suggested for use in chemical process control, in order to improve process monitoring and lessen operator load work.

James H. Taylor is a Professor in the Department of Electrical & Computer Engineering, University of New Brunswick, PO Box 4400, Fredericton, NB CANADA E3B 5A3 [jtaylor@unb.ca](mailto:jtaylor@unb.ca)

Mazyar Laylabadi is a MSc. candidate with the Department of Electrical & Computer Engineering, University of New Brunswick, PO Box 4400, Fredericton, NB CANADA E3B 5A3 [mazyar.L@unb.ca](mailto:mazyar.L@unb.ca)

A brief overview of DR methods and algorithms is outlined in section II. In section III the original NDDR problem formulation and the solution strategy are outlined. In section IV the basic theory of ANDDR, which is proposed as an enhancement to the existing NDDR approach, is presented. Then, this ANDDR algorithm is combined with a novel GED and identification method, and an enhanced dynamic tracking feature is described which addresses cases where the input variable is a ramp or sinusoid. In section V the results of the implementation and application of the proposed package on a simulated CSTR model are depicted. Finally, in section VI conclusions and future work are discussed.

## II. DATA RECONCILIATION BACKGROUND

The DR problem was first introduced by Kuehn and Davidson [6] for linear steady-state models. As far as engineering processes are concerned they often operate dynamically in highly nonlinear regions where traditional methods such as the Kalman filter or extended Kalman filter may be ineffective [2].

The necessity of developing NDDR methods was proposed by Liebman and Edgar [3], and the advantages of using nonlinear programming (NLP) over traditional steady-state DR methods were demonstrated. In the next step Liebman et al. [2] developed their main NDDR algorithm. Their approach was based on simultaneous optimization and solution techniques where efficient state estimation was performed. There was no GED or identification included in the NDDR approach of Liebman et al. [2]; since then there have been some extensions which are capable of handling gross errors (more specifically, of detecting, identifying and editing of gross errors) as well. For instance, Soderstrom et al. [4] proposed an approach to simultaneously tackle the problem of GED and identification together with DR.

Although several authors have stated the need for covariance estimation for DR, none of them has proved the effectiveness of using this matrix in DR except for Alici [5] who demonstrated its necessity and briefly discussed the effects that covariance matrix estimation has on DR. She also addressed the combination of dynamic model identification with NDDR. Finally, Laylabadi and Taylor [1] presented the theory of ANDDR where they enhanced the original NDDR method for the cases where the statistical model is not known. They also presented a novel smart tracking system which tackles the problem of delay seen in the results of NDDR estimations.

### III. PROBLEM FORMULATION AND SOLUTION STRATEGY

In this section first the original NDDR problem is outlined. Next, the solution strategy is presented.

#### A. NDDR General Formulation

1) *Problem Statement:* The general NDDR problem developed by Liebman et al. [2] may be re-stated as follows:

$$\min_{\hat{y}} \phi(\tilde{y}, \hat{y}; \sigma) \quad (1)$$

subject to:

$$f\left(\frac{d\hat{y}(t)}{dt}, \hat{y}(t)\right) = 0 \quad (2)$$

$$h(\hat{y}(t)) = 0 \quad (3)$$

$$g(\hat{y}(t)) \geq 0 \quad (4)$$

Here the corrupted measurements  $\tilde{y}$  and reconciled estimates  $\hat{y}$  include both state variables and input variables. The first constraint (2) represents the process dynamics (often formulated as  $\frac{d\hat{y}}{dt} = f(\hat{y})$ ), the second constraint (3) may describe energy and/or material balance, and the third (4) may impose process variable limits. For more details one can refer to the paper by Liebman et al. [2].

For most applications the objective function is weighted least-square error (WLSE):

$$\phi(\tilde{y}, \hat{y}; \sigma) = \sum_{j=0}^c \frac{1}{2} (\hat{y}(t_j) - \tilde{y}_j)^T V^{-1} (\hat{y}(t_j) - \tilde{y}_j), \quad (5)$$

where  $c$  is the current time,  $\tilde{y}_j$  measured (corrupted) values and  $\hat{y}_j$  the reconciled estimates at discrete time  $t_j$ , and  $V$  is the variance-covariance matrix where each diagonal element  $V_{ii}$  is  $\sigma_i^2$ .

2) *Moving Horizon Window:* The solution adopted by Liebman et al. [2] is a moving horizon window (MHW) approach. Figure 1 shows the basic idea of the MHW approach for NDDR. In this method, after collecting the process measurements up to  $t_c$ ,  $\phi$  is optimized over the horizon from  $t_c - H$  to  $t_c$ , the current time. Then  $\hat{y}(t_c)$  is saved and the procedure is repeated at the next time step [2].

The MHW approach has the advantage of reduced optimization problem size together with giving the user only one tuning parameter, the window horizon  $H$ , compared to other nonlinear approaches such as the EKF where more tuning parameters need to be adjusted. Another advantage is its capability of handling constraints such as equalities and inequalities, whereas other approaches such as the EKF cannot handle the constraints [2].

In order to solve the NLP problem of equations (1) to (4), we need to discretize the nonlinear model presented as the first constraint (equation (2)). We chose the fourth-order Runge-Kutta method to simulate and discretize the model. In other words, for our application  $\frac{d\hat{y}}{dt} = f(\hat{y})$

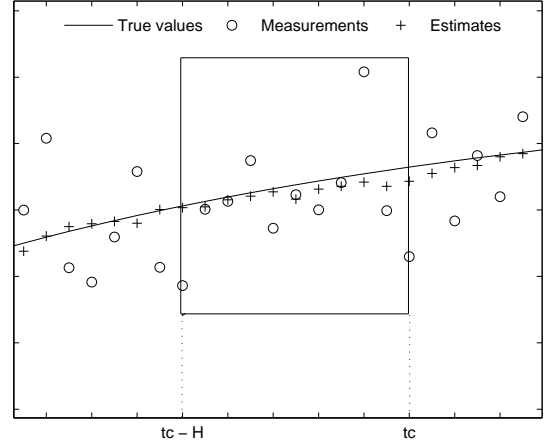


Fig. 1. History horizon for NDDR

is solved numerically over the window horizon and  $\hat{y}_j$  obtained by sampling this solution. Once the discretization is implemented, equations (2) to (5) can be rewritten as the following NLP problem:

$$\min_{\hat{y}} \sum_{i=0}^{n_i+n_s} \eta_i \sum_{j=c-H}^c \left( \frac{\hat{y}_{ij} - \tilde{y}_{ij}}{\sigma_i} \right)^2, \quad (6)$$

subject to:

$$\frac{d\hat{y}}{dt} - f(\hat{y}) = 0 \quad (7)$$

$$h(\hat{y}) = 0 \quad (8)$$

$$g(\hat{y}) \geq 0 \quad (9)$$

where  $f(\hat{y})$ ,  $h(\hat{y})$  and  $g(\hat{y})$  now represent the constraints obtained through discretization,  $\eta$  is a vector of weights and  $n_i$  and  $n_s$  are the numbers of inputs and states (outputs), respectively.

In the original NDDR algorithm, inputs over a moving window were assumed to be constant [2], which caused a significant delay in the estimation of responses to step inputs; this also interferes with tracking in other cases where inputs vary continuously, as discussed here. The step tracking problem was solved in [1] where Laylabadi and Taylor presented their smart tracking system with two input levels over the window instead of one. If there are no gross errors, then a change in a system input that exceeds three times the standard deviation of the noise is declared to be a step input change, and two input levels are adjusted in the optimization procedure; the first level is used up to the time of the step change and the second level thereafter. When there are gross errors, then a modified procedure incorporates logic to distinguish between isolated gross errors and step changes. Results in both cases are presented in [1].

#### IV. ANDDR AND GED METHOD: BASIC THEORY AND FORMULATION

Here the method presented by Laylabadi and Taylor [1] is extended in order to include applications with ramp and sinusoidal inputs. The main purpose of the ANDDR + GED approach is aimed for cases where the statistical model for noise is not given, or, in other words, the standard deviation  $\sigma$  or covariance matrix  $V$  is unknown, and where outliers may occur.

In this study GED is implemented first and gross error effects are removed, then a covariance matrix estimation method is applied. In this methodology the same idea of a moving window approach is used and  $\sigma$  is estimated as each measurement variable is processed. This novel ANDDR + GED can be described as follows: The moving window provides us with  $H$  measurements at each time step. Assume that  $H \geq 10$  in this discussion; if this is not true, then a longer window should be used for estimating  $\sigma$ . Since the measurement error is Gaussian and white (uncorrelated from sample-to-sample), and the true process variables change slowly over a data window, we can assume we have a Gaussian process and use the sample variance to estimate  $\sigma$  for each variable. It is known that the standard deviation of the sample variance for a Gaussian process is [7]:

$$\sigma_{\hat{V}} = \sqrt{2\frac{V}{H}} \quad (10)$$

For  $H \geq 10$  the estimate is adequate for a threshold test that is usually conservatively chosen, e.g., threshold =  $3\hat{\sigma}$ . This is the basis of the  $\sigma$  estimation method which justifies using  $\hat{\sigma} = \sqrt{\hat{V}}$  in solving the ANDDR problem.

The noise standard deviation,  $\hat{\sigma}$ , is up-dated at each time step, and can vary depending on the measurements statistical behavior. To perform GED, the difference  $d_{c,i}$  is derived for each element of  $\tilde{y}_c$  (11), using the previous value of the mean, (13), for each time step, and it is compared with the previous  $\hat{\sigma}$ ,  $\hat{\sigma}_{c-1,i}$ , (14). If  $|d_{c,i}|$  exceeds the threshold, as defined in (12), then the algorithm detects the existence of a gross error and removes it by replacing it with the previous estimate,  $\hat{y}_{c-1}$ .

$$d_{c,i} = \tilde{y}_{c,i} - \hat{m}_{c-1,i} \quad (11)$$

$$\text{IF } |d_{c,i}| > 3\hat{\sigma}_{c-1,i} \text{ THEN } \tilde{y}_{c,i} \text{ is an outlier} \quad (12)$$

$$\hat{m}_{c,i} = \sum_{j=c-H}^c \left( \frac{\tilde{y}_{i,j}}{H} \right) \quad (13)$$

$$\hat{\sigma}_{c,i} = \sqrt{\sum_{j=c-H}^c \left( \frac{(\tilde{y}_{i,j} - \hat{m}_{c,i})^2}{H-1} \right)} \quad (14)$$

Since in this study the input to the model is assumed to be a ramp, sinusoid or other smooth function, the difficulty of set-point change detection does not exist. One can refer to [1]

where the problem of the set-point change is tackled while GED is performed.

The key modification that we made for continuous inputs is the way that input variables are estimated for each moving window. In this paper the inputs are assumed to be ramps or slow sinusoids. Therefore, the extended ANDDR solution is obtained by assuming that the input over each moving window is a ramp. In this way, the algorithm can be significantly faster and smoother in tracking the dynamic behavior of a system with continuous inputs.

#### V. CASE STUDY

In this section the performance of the proposed algorithm is demonstrated. The case studied in this paper is a simulated CSTR model cited commonly in the literature. First, the situation where the inputs to the model are ramps is considered. Then, in the next study the response of the system to sinusoidal inputs is considered and the tracking capability for general slow and smoothly varying inputs is illustrated.

##### A. Ramp Function Tracking

We consider the CSTR model used by Liebman et al. [2] with the same assumptions and parameters values, except the input is a ramp function. The normalized model can be presented as follows:

$$\frac{dA}{dt} = \frac{q}{v}(A_0 - A) - \alpha_d k A \quad (15)$$

$$\frac{dT}{dt} = \frac{q}{v}(T_0 - T) + \alpha_d \frac{\Delta H_r A_r}{\rho C_p T_p} k A - \frac{UA}{\rho C_p V}(T - T_c) \quad (16)$$

$$k = k_0 \exp\left(\frac{-E_A}{T T_r}\right) \quad (17)$$

where the input stream feed concentration,  $A_0$ , and feed temperature,  $T_0$ , are input variables and the concentration,  $A$ , and temperature,  $T$ , are output variables. There are two simple constraints on both input and output variables as follows:

$$0 \leq A, A_0 \leq 20.0 \quad (18)$$

$$0 \leq T, T_0 \leq 10.0 \quad (19)$$

The values of other constants are provided in [2].

1) *Gross-Error-Free Ramp Tracking*: In this case study both the two inputs and two states (outputs) are estimated, assuming that no gross errors exist. Measurements were simulated by creating measurement noise which is assumed to be Gaussian with  $\sigma$  equal to 0.05 and zero mean [2]. The time step is assumed to be 2 seconds and the simulation is run for 100 samples with window width of  $H = 10$ . Obviously, the first estimates are obtained at time step 10 where the first window of measurements is available. The first input,  $A_0$ , is assumed to be a ramp starting from 6.5 and increasing to 8.5,

as shown in Fig. 2, and the second input,  $T_0$ , is assumed to be constant at 3.5.

Figures 2 to 5 demonstrate the successful implementation of the ANDDR approach for the ramp function. The solid lines in these figures show the true values, circles show the corrupted measurements, and plus signs mark the proposed ANDDR estimation results. There is a significant noise reductions in both input and output estimation.

To demonstrate the successful  $\sigma$  estimation feature of the ANDDR algorithm, the values of  $\hat{\sigma}_i$  for one input variable ( $T_0$ ) and one output variable ( $T$ ) are depicted in figures 6 and 7 knowing that the true  $\sigma$  is 0.05. Observe that the variation in  $\hat{\sigma}$  for each variable is in the order of  $\pm 40\%$  in each case, which is much less than the threshold defined in this methodology ( $3\hat{\sigma}$ ).

2) *Gross Errors and Ramp Tracking*: With the same assumptions made as for the previous case, random gross errors are added to the measurements of each input and output variable in this second study. There are four random gross errors for each variable with different amplitudes. Figures 8 and 9 demonstrate the successful GED algorithm results for one input ( $A_0$  and one output ( $T$ ). These figures show the comparison of the situation where GED was implemented along with ANDDR, and where ANDDR is individually implemented. The solid lines in these figures show the true values, circles show the corrupted measurements, stars present the proposed ANDDR + GED estimation results and plus signs mark the ANDDR data without GED. As the figures show, the gross errors have been detected and successfully removed, and the estimation has not been corrupted. Observe that the outliers cause significant corruption of the estimates from ANDDR alone; the use of the algorithm with GED eliminates this problem. Note that the method is applicable where many gross errors may exist in the measurements, whether they are successive or isolated, since the algorithm can detect and identify them without the need to distinguish outliers from step changes.

### B. Sinusoidal Input Tracking

In this section the performance of the algorithm for sinusoidal inputs is studied. The previous CSTR model with the same parameters and assumptions is used. The feed concentration,  $A_0$ , is a slow sinusoidal function as follows:

$$A_0 = \sin\left(\frac{\pi}{100}t\right) + 6.5, \quad (20)$$

and the feed temperature,  $T_0$ , is constant with the same value of 3.5.

1) *Gross-Error-Free Sinusoidal Input Tracking*: First we consider the measurements are not corrupted by gross errors and only contain the same zero-mean Gaussian noise defined in section V-A.1. As Figures 10 and 11 demonstrate, the ANDDR algorithm has successfully tracked the sinusoidal behavior of the model, for both input and output estimates. A slight delay is seen in the estimation of the feed concentration,  $A_0$ . One suggestion to reduce this delay would be to assume two different slopes for the input estimation in

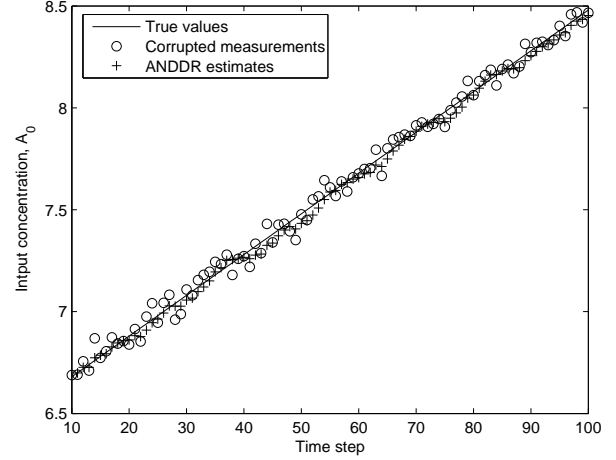


Fig. 2. First input,  $A_0$ , estimation (gross-error-free).

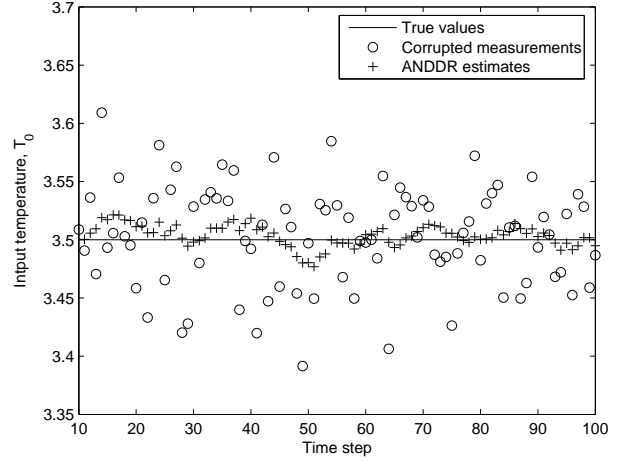


Fig. 3. Second input,  $T_0$ , estimation (gross-error-free).

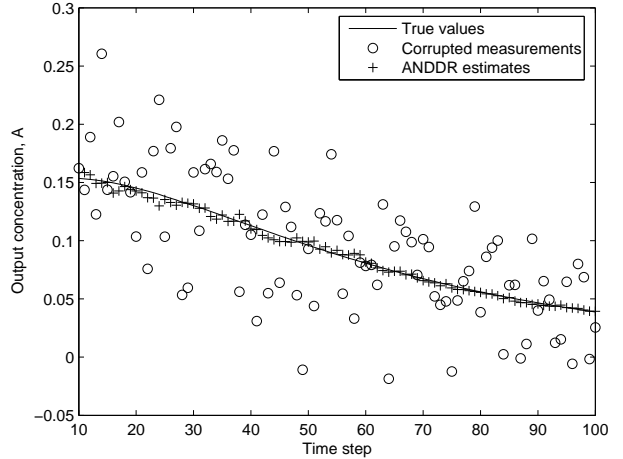


Fig. 4. First output,  $A$ , estimation (gross-error-free).

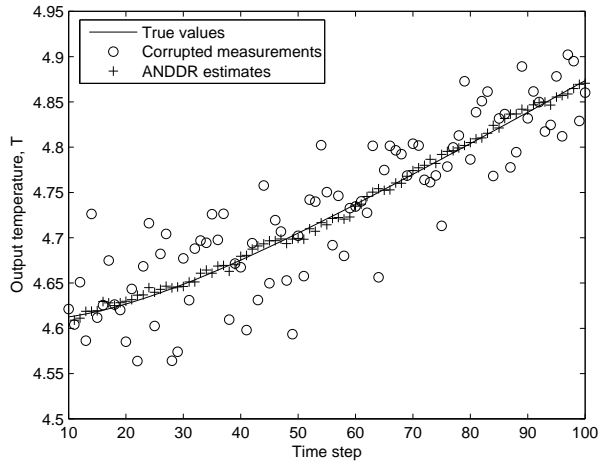


Fig. 5. Second output,  $T$ , estimation (gross-error-free).

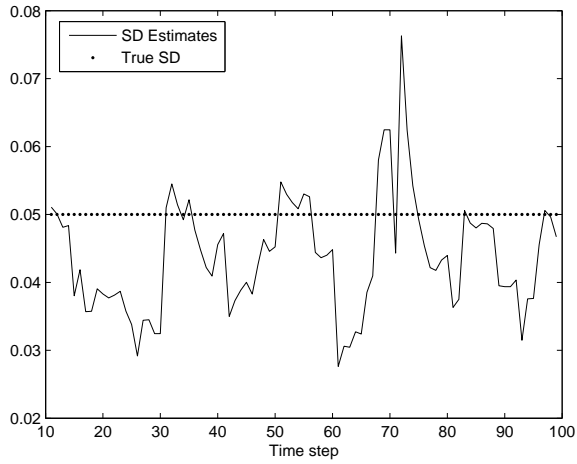


Fig. 6.  $\sigma$  estimates obtained for the second input,  $T_0$

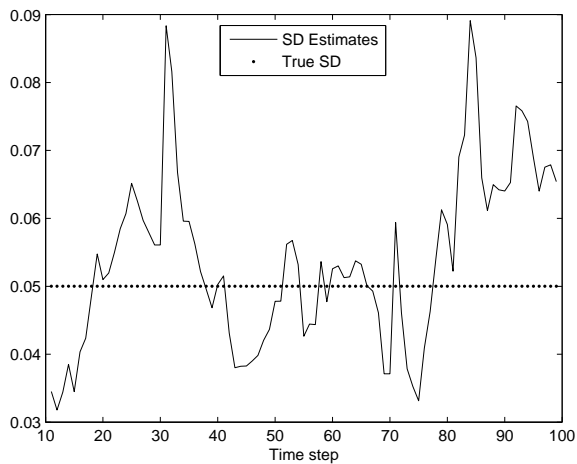


Fig. 7.  $\sigma$  estimates obtained for the second output,  $T$

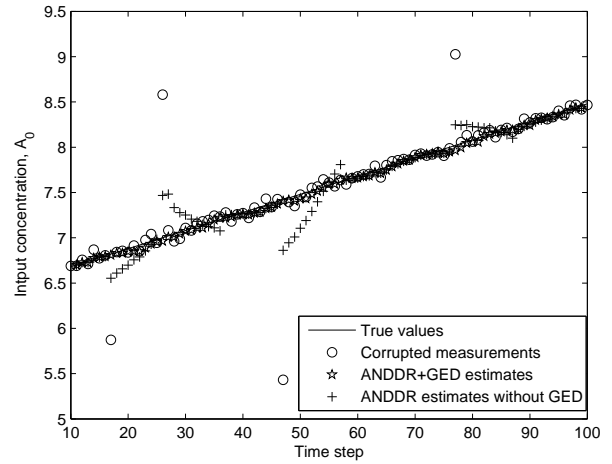


Fig. 8. First input,  $A_0$ , estimation (with gross errors).

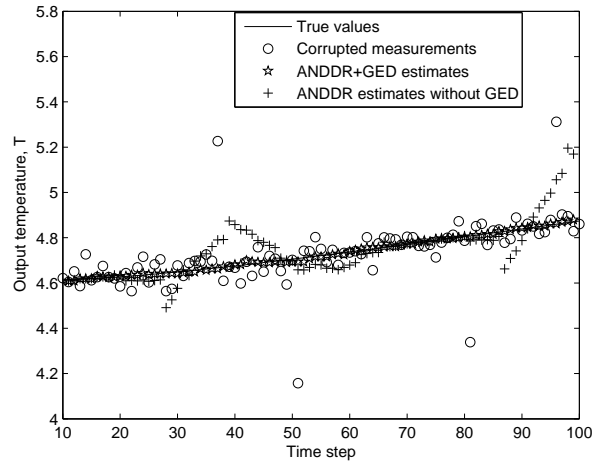


Fig. 9. Second output,  $T$ , estimation (with gross errors).

reduce the delay seen in the estimates of feed concentration, but increase complexity and solution time.

2) *Gross Errors and Sinusoidal Input Tracking*: To prove the performance of the proposed GED algorithm, we consider the existence of gross errors. There are four random outliers added to each input and output variable measurement, and the proposed ANDDR + GED is implemented. Figures 12 and 13 show the successful detection of outliers and make comparison studies of the two cases of ANDDR and ANDDR + GED implementations. It is clear that without GED the estimates become seriously corrupted.

## VI. CONCLUSIONS AND FUTURE WORKS

The ANDDR + GED algorithm was successfully implemented and tested on the simulated CSTR model of Liebman et al. [2] when the inputs are ramps or sinusoids. In both cases the estimates are satisfactory. A very slight delay exists in sinusoidal input tracking; however, we believe it can be reduced further by another simple extension (using two input slopes over the window, as mentioned).

each moving window. This modification would significantly

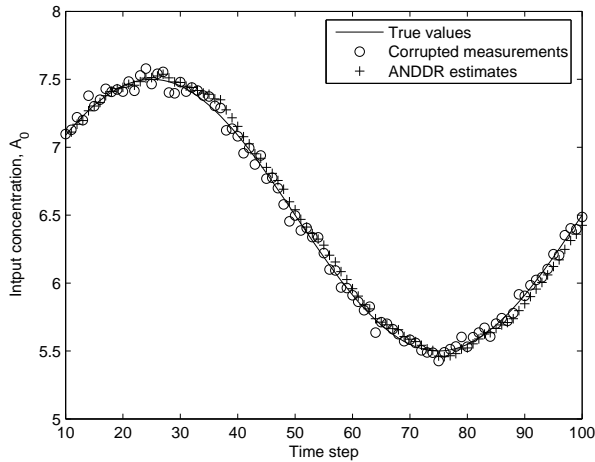


Fig. 10. First input,  $A_0$ , estimation (gross-error-free).

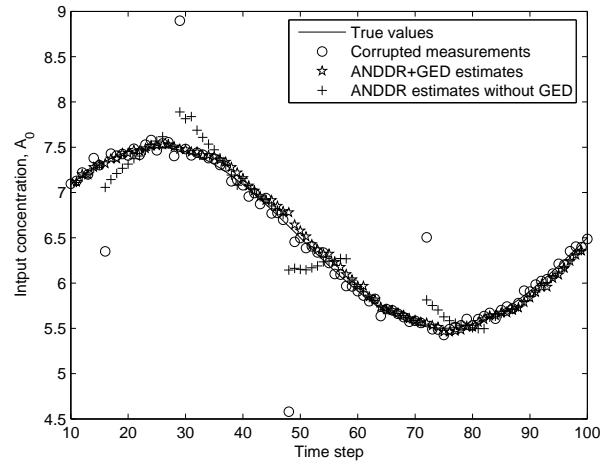


Fig. 12. First input,  $A_0$ , estimation (with gross errors).

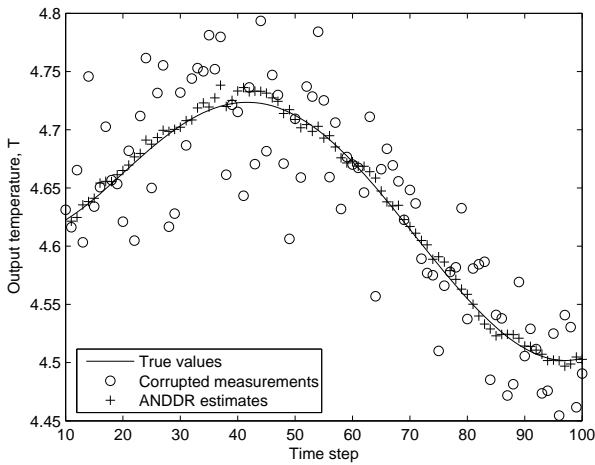


Fig. 11. Second output,  $T$ , estimation (gross-error-free).

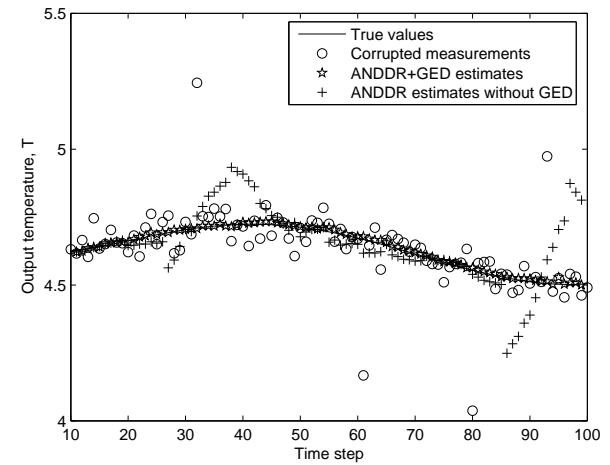


Fig. 13. Second output,  $T$ , estimation (with gross errors).

Note that in all estimation results presented here for ANDDR + GED the assumption was made that the statistical model,  $\sigma$ , was not available. All the results demonstrate that the use of the estimated  $\sigma$  was successful and no accuracy degradation was observed in the main NLP problem solution.

This package with its dynamic tracking features, statistical model identification and gross error elimination is suggested for use in industrial process control applications in order to improve process monitoring and lessen operator work load. Adding a dynamic model identification feature to the proposed approach is suggested as possible future work.

## VII. ACKNOWLEDGMENTS

This project is supported by Atlantic Canada Opportunities Agency (ACOA) under the Atlantic Innovation Fund (AIF) program. The authors gratefully acknowledge that support and the collaboration of the Cape Breton University (CBU), the National Research Council (NRC) of Canada, and the College of the North Atlantic (CNA).

## REFERENCES

- [1] M. B. Laylabadi and J. H. Taylor, "Anddr with novel gross error detection and smart tracking system," *Proc. 12th IFAC Symposium on Information Control Problems in Manufacturing (INCOM2006)*, Saint-Etienne, France, 2006.
- [2] M. J. Liebman, T. F. Edgar, and L. S. Lasdon, "Efficient data reconciliation and estimation for dynamic processes using nonlinear programming techniques," *Computers and Chemical Engineering*, vol. 16, no. 11, pp. 963–986, 1992.
- [3] M. J. Liebman and T. F. Edgar, "Data reconciliation for nonlinear process," *Proceedings of the AIChE Annual Meeting*, Washington, DC, 1998.
- [4] T. A. Soderstrom, D. M. Himmelblau, and T. F. Edgar, "A mixed integer optimization approach for simultaneous data reconciliation and identification of measurement bias," *Control Engineering Practice*, vol. 9, pp. 869–876, 2001.
- [5] S. Alici and T. F. Edgar, "Nonlinear dynamic data reconciliation via process simulation software and model identification tools," *Industrial and Engineering Chemistry Research*, vol. 41, no. 16, pp. 3984–3992, 2002.
- [6] D. R. Kuehn and H. Davidson, "Computer control," *Chemical Engineering Progress*, vol. 57, pp. 44–47, 1961.
- [7] J. H. Taylor, "Statistical performance analysis of nonlinear stochastic systems by the monte carlo method," *Mathematics and Computers in Simulation*, vol. XXIII, 1981.