2009 International Nuclear Atlantic Conference - INAC 2009
Rio de Janeiro,RJ, Brazil, September27 to October 2, 2009
ASSOCIAÇÃO BRASILEIRA DE ENERGIA NUCLEAR - ABEN
ISBN: 978-85-99141-03-8

Particle Swarm Optimization applied to Data Reconciliation in Nuclear Power Plant

Eduardo Damianik Valdetaro[1], Roberto Schirru[2]

[1] Eletrobrás Termonuclear S.A. - ELETRONUCLEAR
Departamento GOD.O
Rod. Mário Covas (BR-101) Km 522
23948-000 Angra dos Reis, RJ
valdtar@eletronuclear.gov.br

[2] COPPE/UFRJ – Programa de Engenharia Nuclear
Caixa Postal 68509
21945-970 Rio de Janeiro, RJ
schirru@lmp.ufrj.br

ABSTRACT

Mass and energy balance are important issues that needs to keep into account in Nuclear Power Plants. Data Reconciliation and Parameter Estimation (DRPE) and gross errors detection are techniques of increasing interest. Works using Genetic Algorithm (GA) have been successfully used in the Data Reconciliation (DR) nonlinear optimization problem, and it seems that Evolutionary Algorithms performs well without the complex calculations used by the conventional methods. The aim of this paper is to present the Particle Swarm Optimization Algorithm (PSO) as an alternative to the use of modified GA, which was applied to Data Reconciliation with simultaneous gross errors detection. In this paper, the DR formulation uses a redescending estimator as objective function and simulation results show that PSO applied to DRPE problem is faster than modified GA presented in literature, do not involve complex calculations and do not need complex parameters to adjust. The PSO algorithm is also able to handle the non-diferentiable characteristics of the redescending estimator.

1. INTRODUCTION

The application of data reconciliation (DR) and gross error detection techniques in chemical and petrochemical industry [1] are already important problems in monitoring and process optimization. Particularly, the importance of DR applied to keep the mass and energy balance into account in Nuclear Power Plants (NPP) is increasing in interest, growing fast and bringing direct and indirect financial benefits.

The principal way of determining the thermal reactor power in NPP is by heat balance and is based in single parameter measurements. Instrumentation errors lead to an uncertainty of several percent and by regulatory laws, a NPP has a legal license to operate a 100% plus 2%, which is the range analyzed by emergency cooling analysis [2].

The Data Reconciliation and Parameter Estimation (DRPE) technique applied to determining the thermal reactor power seems to be a promising way to diminish uncertainty and have a quantified and better accuracy. The smaller the uncertainty, the more margin that can be used for operating at high level of power within the safety level.

During process monitoring, plant data can be inaccurate due to random errors or even gross errors. These errors introduce uncertainties that could interfere in a safe operation, in the closure of mass and heat balance, in the performance calculation or in the data quality for another process application [3].

Measurements contaminated with random errors and systematic errors which do not satisfy process constraints need to be reconciled, i.e., adjust data to satisfy process constraints while minimizing the error in the least square sense [4]. DRPE is the technique that perform these adjustments so as the reconciled values can be used in the model parameters.

Gross errors typically reflect in a process as a measurement bias in one or more variables and can seriously affect the reconciled values. Detection of gross errors or outliers permits to identify the unfavorable influence in order to reduce their effect in the reconciled data or even completely remove them.

There are a variety of sources of gross errors that can be enumerated, for example, leaks, stuck sensor, miscalibrated sensor. Kong et al. [5] refer to gross error as a disturbance of two classes: a) measurement bias and b) outliers, where bias are situations in which the measurements values are too low or too high and outliers are measurements abnormal behavior, like peaks or unmeasured disturbances.

Data Reconciliation was first proposed in 1961 by Kuehn and Davidson [6] and after that, extensively studies about DR and gross errors detection have been developed.

Methods based on Lagrange multipliers [7], matrix projection method to decompose the problem in order to evaluate measured and unmeasured variables sequentially [8], quadratic programming technique [9] and methods that exploit the least squares structure were proposed and many others DR and gross error detection methods are described in literature, for example, in Pai and Fisher [10], Tjoa and Biegler [4].

Some disadvantages from these methods are the need of derivatives calculations, Jacobian actualization, matrix inversion, which can causes numerical problems, slow rate convergence, high computational effort. These methods also require complex calculation and usually use linear or linearized models.

Recent methods based on stochastic search and evolutionary theory, like Genetic Algorithm (GA) have characteristics that can surpass the disadvantages mentioned above.

One is the method proposed by Zhao and Jiang [11] that is based on stochastic search and does not need complex calculation or information about any particular model structure and has only algebraic calculation.

Other methods are based on Genetic Algorithms. Moros et al. [12] used GA for generating the initial parameter for a kinetic model of a catalytic process and recently, Wrongat et. al. [13] applied successfully a modified GA proposed by Wasanapradit [14] for Nonlinear Data Reconciliation.

Data Reconciliation is based on the assumption that noise is distributed with zero mean and known variance. The aim of DR process is to minimize quadratic error subject to process constraints and it usually uses the objective function as a weighted least square.

When measurement contains gross errors or outliers, the reconciled values can be seriously biased estimated in the case above. In order to avoid such effect in the estimated data, some authors proposes to use a robust estimator, like the Fair function, Huber estimator, the three part redescending estimator of Hampel, or another robust estimator [15]. These robust estimators can be used in place of weighted least square and they put less weight on large residuals that corresponds to outliers.

Arora and Biegler [15] compared the redescending estimator with least squares and fair function and concluded that the first is more robust and has a cutoff point that permits more efficient outliers detection.

Wrongat et al. [13] joined the robustness of M-estimators and the relative simplicity of genetic algorithm and proposed a modified GA applied to nonlinear data reconciliation, which performs well and uses evolutionary technique.

The modified GA contains also some adaptation for enhancing its performance and avoids premature convergence using Cross Generational Probabilistic Survival Selection (CPSS), method first proposed by Wasanapradit [14]. Some adaptations and improvements in this GA led to new parameters to adjust in order to become appropriate to nonlinear data reconciliation.

Observing the wide range of application of evolutionary algorithms and the promising results from the work of Wrongat et. al [13], we propose in this paper, 1. Apply another related evolutionary algorithm – the Particle Swarm Algorithm (PSO) - as an alternative to the use of the modified GA presented in [13]; 2.Verify the properties of the redescending estimator; 3. Discuss similarities and differences between two approaches (GA x PSO), which in the first instance, the PSO algorithm do not need special parameters to adjust; 4. Solve nonlinear data reconciliation example. The paper is divided in the following parts: In section 2, Data Reconciliation and Redescending Estimator are discussed. In section 3 and 4, we present the Particle Swarm Optimization (PSO) and its application to DRPE and gross errors detection. Finally, in section 5, we conclude the paper with an example with simulation results. Section 6 is for final conclusions.

2. DATA RECONCILIATION AND GROSS ERROR DETECTION

A general Data Reconciliation and Parameter Estimation problem is defined as

$$\min_{x,u,p} \Im(x^M, x) \quad s.t. \quad \begin{array}{ll} h(x,u,p)=0 & x^L \le x \le x^U \\ g(x,u,p)\le 0 & u^L \le u \le u^U \\ & p^L \le p \le p^U, \end{array} \qquad (1)$$

where, $\Im$ is an objective function, which depends on the difference between the measurements x^M and the values of the estimate variable **x** , **p** the set of parameters, **u** the set of unmeasured variables, **h** the set of equalities constraints, **g** the set of inequalities constraints, the L and U indices corresponds to the lower and upper limits of the variables **x**, **u** and **p**.

Various DRPE methods assume that the measurements are normally distributed with zero mean and known variance. In this case the objective function takes the form:

$$\Im = (x^M - x)^T . V^{-1} . (x^M - x) \quad , \tag{2}$$

where **V** is the covariance matrix. Usually, the covariance matrix is not known from the process and needs to be estimated. In this case, considering the maximum likelihood principle, an equivalent problem is to minimize the function $\Im$ below subject to the constraints in equation (1).

$$\Im = (x^M - \mu)^T . \Sigma^{-1} . (x^M - \mu) \quad , \tag{3}$$

where, Σ is the real unknown covariance matrix and μ is the real mean value from the measurements x^M .

Comparing (2) and (3), we note that the DRPE problem is equivalent to find x, the reconciled value, which is by (3) the real mean value from the measurements, depending upon estimates of process covariance matrix Σ [16].

These data in the presence of gross errors or outliers can severely bias the covariance estimation and the mean value, which in turn will affect unfavorably data reconciliation. An important task to avoid bias on reconciled data is to perform the previous removal from gross errors and outliers during DRPE problem.

When data are corrupted by outliers is often difficult to determine a probability distribution function and use an estimator that is also derived from a fixed probability distribution [15]. In order to circumvent this limitation, robust estimator is used as an alternative to a fixed distribution estimator because, it is largely distribution independent, produces unbiased estimates derived from an ideal distribution and it is not sensitive to a wide range of deviations. In fact, Robust estimators add less weight to outlying measurements, protecting others measurements from being corrupted.

Arora and Biegler [15] and Wrongat et. al. [13] applied successfully a three part redescending estimator in a simultaneous DRPE problem, with a straightforward application of outlier detection. In the next section we present the redescending estimator and procedure for tuning its constants according mentioned in [13. 15].

2.1. The Redescending Estimator

The three part redescending estimator of Hampel is of M-estimator type. The M-estimator has associated a characteristics function ψ, which quantify the effect in the residuals in the estimated data.

When the function $\psi \rightarrow 0$ as $x^M \rightarrow \pm\infty$, the estimator is called redescending and such estimator has the property of complete rejection of gross outliers.

Other references concerning the use of redescending estimator in DRPE and gross error detection can be seen in Arora and Biegler [15], Wrongat et. al. [13] and Ozyurt and Pike [17].

The redescending estimator of Hampel is as follow:

$$F_H \begin{cases} \frac{1}{2}\varepsilon_i^2, & 0 < |\varepsilon_i| \leq a \\ a.|\varepsilon_i| - \frac{1}{2}a^2, & a < |\varepsilon_i| \leq b \\ a.b - \frac{1}{2}a^2 + (c-b)\frac{a^2}{2}\left[1 - \left(\frac{c - |\varepsilon_i|}{c-b}\right)^2\right], & b < |\varepsilon_i| \leq c \\ a.b - \frac{1}{2}a^2 + (c-b)\frac{a^2}{2} & |\varepsilon_i| > c \end{cases} \quad (4)$$

where F_H is the redescending estimator objective function, ε_i the data residual ($x_i^M - x_i$) and a, b, c are the tuning constants satisfying

$$c \geq b + 2.a \quad . \quad (5)$$

We can note that between the interval [-a, a] the estimator behavior is similar to the least squares function. The probability falls rapidly in the intervals [-c,-b] and [b,c]. The probability remains almost constant from (-∞,c) and (c,∞). This long tail is responsible for the robustness.

Adjusting the constants a, b and c can be considered an drawback from the redescending estimator, but at the same time, this adjustment serves to cope with the tuning to the system being monitored in order to get the best fit to the data and also detect outliers [13].

One advantage is that the redescending estimator (4) has an explicit cutoff point, *c*, i.e., a measurement considered to have an outlier if its residual are greater than c,

$$|\varepsilon_i| \geq c \quad . \quad (6)$$

Other estimators do not have such cutoff points, like least squares or Fair function so, for outlier detection we thus have to resort to exploratory data analysis before proceeding with data reconciliation.

2.2. Tuning the Redescending Estimator

The redescending estimator comes from a broad family of distributions, so it is necessary adjust the constants a, b, c in order to fit the corrected data to an individual member of these families of probability distribution.

Arora and Biegler [15] used the Akaike Information Criteria (AIC) [18] and an iterative procedure to adjust the redescending estimator constants. Several combinations of a, b, and c are considered in order to get a minimum of AIC bounded. Once the minimum are bounded, a *golden section search* is performed to obtain the best a, b, and c that minimizes AIC criteria.

Here we use the following relationship for the AIC calculation derived in Yamamura et al. [19] with the assumption that the errors are normally distributed and the outliers are removed.

$$AIC = \sum_{i=1}^{n_m - n_{out}} \left(\frac{x_i^M - x_i}{\sigma_i} \right)^2 + 2.n_{out} \quad , \tag{7}$$

where x_i^M is the *i*th measured variable, x_i is the estimated value, σ_i the standard deviation of the *i*th variable, n_m the total number of measured data, n_{out} the number of detected gross errors. The first term of eq. (7) is the goodness of fit and the second term is the penalty. After finding a, b and c optimal, outliers are detected comparing the residual with constant c as indicated in eq. (6).

3. PARTICLE SWARM OPTIMIZATION

Many evolutionary techniques are motivated by nature evolution, for example, Genetic Programming and Genetic Algorithms (GA). Individuals in a population encode the problem solution and are manipulated using "genetic" operations, such crossover, mutation, selection and reproduction according to the fitness of each individual [20].

Eberhart and Kennedy [21] proposed a different algorithm through simulating social behavior. This algorithm is called particle swarm optimization ***(PSO)*** since its behavior is similar to a school of flying birds. In a particle swarm optimizer, no genetic operators are used, these individuals are "evolved" by cooperation and competition between themselves [20].

Each generation, particles adjusts its flying according to its own flying experience or own best position and its companions flying experience or the global best position. Each individual act as a "particle" and each one represent a potential solution to a problem.

Each particle is treated as a point in a n-dimensional space. The ith particle is represented as $\mathbf{X_n} = (\mathbf{x_{i1}} , \mathbf{x_{i2}}, \mathbf{x_{i3}}, \dots , \mathbf{x_{in}})$. The best previous position of the individual or the position giving the best fitness value of any particle is recorded and represented as $P_n = (\mathbf{p_{i1}} , \mathbf{p_{i2}}, \mathbf{p_{i3}}, \dots , \mathbf{p_{in}})$. The index of the best particle among all the particles in the population is represented by the symbol ***g***. The rate of the position change (velocity) for particle *i* is represented as $\mathbf{V_n} = (\mathbf{v_{i1}} , \mathbf{v_{i2}}, \mathbf{v_{i3}}, \dots , \mathbf{v_{in}})$.

The particles are manipulated according to the following equation:

$$v'_{in} = w.v_{in} + c1.rand_1.\left[p_{in} - x_{in}\right] + c2.rand_2.\left[p_{gn} - x_{in}\right] \quad (8)$$

$$x'_{in} = x_{in} + v'_{in} \quad (9)$$

where c1 and c2 are two positive constants, $rand_1$ and $rand_2$ are random numbers in the range [0,1] and w is the inertia weight.

The second part of the equation (8) represents the "cognition" part of the individual or private thinking of the particle itself.

The third part is the "social" or "global" part, which represents the collaboration among the particles [20]

Equation (8) adjusts the new velocity according to: a) the velocity from the last iteration; b) from the difference between the actual position and its own best position; c) from the difference between the actual position and the global best position.

The particle flies toward a new position according to equation (9). The performance of each particle is measured according to a predefined fitness function, which is related to the problem to be solved.

The particle swarm optimizer has been found to be robust and fast in solving nonlinear, non-differentiable, multi-modal problems.

The inertia weight and the constants c1 and c2 can be adjusted with usual recommended values in [22, 23]: w=0.7298, c1=2.05, c2=2.05 .

4. PARTICLE SWARM OPTIMIZATION APPLIED TO DRPE PROBLEM

The methods using stochastic search and evolutionary algorithm (GA) have the ability to overcome the disadvantages of techniques that uses derivative calculations and they do not require complex calculation. The modified genetic algorithm implemented by Wrongat et al. [13] is able to handle the non-differentiable term of the redescending estimator and are effective in solving DR problem.

The modified GA used in Wrongat et al. [13] and proposed by Wasanapradit [14] was implemented for real number representation and to prevent premature convergence used a Cross-generational Probabilistic Survival Selection (CPSS).

Using CPSS at selection phase, GA avoids premature convergence, but became necessary to adjust two new constants (α, s) in the selection probability, both of them need to be externally adjusted with appropriate values for a specific problem [13].

In the implementation of the modified GA to DR problem, it is indicated that the tuning of the constants a, b and c of the redescending estimator needs to be choosed in each iteration. There are complex steps in generating new population before applying the CPSS method with the elimination of duplicate chromosomes and later after CPSS new chromosomes needs to be created to complete the population number.

All these steps seem to take a longer time to execute the modified GA as verified in [13], which in practice can be a considerable drawback.

The implementation of the PSO algorithm is straightforward and simpler than the modified GA, which in turn is less time consuming when applied to DRPE problem. In the same way, as the GA algorithm, they do not need derivative or complex calculation and its characteristics permits to handle the non-convex characteristics of the redescending estimator.

4.1. Particle Swarm Algorithm for Handling Nonlinear DRPE

Data reconciliation problem (1) has inequality and equality constraints and the GA do not generate solutions to (1) randomly.

Prakotpol [24] used a simple technique to solve the constraints equations and address feasibility; all variables have to be divided into two groups: independent and dependent variables. The values of independent or measured variables are generated from the PSO algorithm, whereas the values of dependent variables come from simultaneous solutions of a set of equality constraints after assigning the values of measured variables.

With the procedure above, the DRPE problem using PSO handles only measured variables and calculates the unmeasured variables subject to the problem constraints. The following steps summarize the PSO applied to DRPE problem:

1) Set the initial PSO usual parameters or constants: a) w=0.7298; b) c1=2.05 and c) c2=2.05, Nº of iterations and Nº particles.
2) Select the *m* independent variables of the problem and initiate within the bounds randomly a set of N particles to form the population. $X_i = (x_{i1} , x_{i2},. x_{i3}, \ldots , x_{im})$, i=1,...N.
3) Solve the constraints to solve the rest of the remaining unmeasured variables.
4) Determine the redescending estimator constants a, b and c satisfying (5) and minimizing AIC criteria (7).
5) For $X_I(t)$, Let t=t+1 and Perform DRPE problem with PSO algorithm minimizing the objective function with the redescending estimator; keep the gbest position $p_{gn}(t)$ (8).
6) Use $p_{gn}(t)$ to determine gross errors and outliers (6) of the measurements.
7) Find the values of unmeasured values solving a set equalities constraints simultaneously.
8) Return to step 5 for on line monitoring.

In the next section we tested our proposed PSO application in DRPE problem proposed by Pai and Fisher [9] , which is used widely in literature for testing purposes. We thus compare some aspects to the results from Wrongat et. Al. [13] and Arora and Biegler [15] despite of the measurements are generated differently.

5. EXAMPLE OF NONLINEAR DRPE PROBLEM USING PSO

Here we validated the use of PSO algorithm using the redescending estimator considering using the problem presented by Pai and Fisher [10] and tested in Wrongat et al. with the modified GA [13], Arora and Biegler [15] with redescending estimator with conventional optimization method, Tjoa and Biegler [4], Zhou et al. [25].

This problem has five measured variables (x^M) and three unmeasured variables (u) and the six nonlinear constraints are:

$$0.5x_1^2 - 0.7x_2 + x_3.u_1 + x_2^2.u_1.u_2 + 2x_3.u_3^2 - 255.8 = 0 \quad , \tag{10a}$$

$$x_1 - 2x_2 + 3x_1.x_3 - 2x_2u_1 - x_2.u_2.u_3 + 111.2 = 0 \quad , \tag{10b}$$

$$x_3.u_1 - x_1 + 3x_2 + x_1.u_2 - x_3.u_3^{0.5} - 33.57 = 0 \quad , \tag{10c}$$

$$2x_1 + x_2.x_3.u_1 + u_2 - u_3 - 126.6 = 0 \quad , \tag{10d}$$

$$x_4 - x_1 - x_3^2 + u_2 + 3u_3 = 0 \tag{10e}$$

$$x_5 - 2x_3.u_2.u_3 = 0 \quad , \tag{10f}$$

The exact values of these variables are:

$$x_e = [\, 4.5124,\ 5.5819,\ 1.9260,\ 1.4560,\ 4.8545 \,] \quad , \tag{11}$$

$$u_e = [11.070,\ 0.61467,\ 2.0504] \tag{12}$$

The simulated values are produced according eq. (13), where the value of x^M are simulated adding noise (η) with a standard deviation σ of 0.1 and gross error (υ) of 25σ. Each set of measured variables is corrupted by 20 gross errors. The first 20 gross errors are added to x_1, other 20 gross errors are added to x_2 and so on, totalizing 100 gross errors.

$$x^M = x_e + \eta + \upsilon \tag{13}$$

The objective function for DRPE problem is described as eq. (14) and presented below

$$\min_x \sum_{i=1}^{5} \sum_{j=1}^{100} F_H \left(x_i - x_{ij}^M \right), \tag{14}$$

s.t. equation (10) and F_H according eq.(4).

This example is used for test the behavior and performance of the PSO algorithm with the redescending estimator, as well its tuning procedure.

5.1. Tuning Procedure Using PSO

The usual parameters from PSO are used and the procedure presented in item 2.2 is executed. The PSO code was tested with various tuning constant following the work of Arora and Biegler [15] and Wrongat et al. [13].

The results from the tuning procedure in the problem of DRPE are presented in Table 1. The results reflect the tuning procedure of the work from Arora and Biegler [15] and Wrongat et al [13]. The results are not exactly due to differences in the random noise in data generation.

Table 1 presents the tuning constants a, b and c; the Akaike Information Criteria normalized (AICnorm) by the total measured values and the number of outliers detected (n_out).

Table 1. Results of Data Reconciliation (tuning redescending estimator)

	x_1	x_2	x_3	x_4	x_5	u_1	u_2	u_3
True Values	4.5124	5.5819	1.9260	1.4560	4.8545	11.0700	0.61467	2.0504
Redescending Estimator (a, b, c, AICnorm, n_out)								
R1 (2, 4, 8, 424.27, 20)	4.6736	5.5770	2.4257	1.8942	4.9577	8.8443	0.3696	2.7646
R2 (1, 2, 4, 47.32, 60)	4.5061	5.5782	2.0878	1.6967	4.8628	10.2415	0.5259	2.2141
R3 (0.5, 1, 2, 1.300, 100)	4.5311	5.5923	1.9376	1.4604	4.8471	10.9831	0.6031	2.0739
R4 (0.375, 0.75, 1.5, 1.150, 100)	4.5215	5.5943	1.9199	1.4491	4.8602	11.0780	0.6184	2.0467
R5 (0.25, 0.5, 1, 1.279, 100)	4.5175	5.5968	1.9460	1.4277	4.8710	10.9314	0.5980	2.0930
R6 (0.1875, 0.375, 0.75, 1.156, **160**)	4.5167	5.5710	4.8150	1.4808	4.8644	4.7057	0.0579	8.7207
R7 (0.125, 0.25, 0.5, 1.352, **160**)	4.5189	5.5703	1.9207	1.4699	12.1363	10.9490	2.0009	1.5790
R8 (0.369, 0.739, 1.478, 1.21, 100)	4.5146	5.5728	1.9352	1.4497	4.8453	11.0377	0.6053	2.0681

Note that R1 and R2 estimators did not detect the correct number of outliers because of the high value of AICnorm. The R6 and R7 estimator detected more outliers than simulated, because AICnorm starts to increase and at the same time decrease the quality of the estimation and outlier detection. The behavior of the tuning strategy presented here and in the previous work of Arora and Biegler [15] and Wrongat et al. [13] is similar, but can not be exactly compared due to the different form of data generation.

5.2. Example from Pai and Fisher [9] – revisited

The DRPE problem was solved using the constants a, b, c found in Arora and Biegler [13] and presented in Table 1 in R8 estimator. The R5 estimator was the best value found during the simulations, but R5 and R8 estimator are also a near optimum solution and both can be used in the simulation. For comparison purposes, we chose to use R8 in the test with PSO applied in DRPE problem.

In equation (14) the size of measurement horizon is 100 so, in order to generate data in a similar mode of a real time system we use 200 measurements. Each set of measured variables is corrupted by 20 gross errors. The first 20 gross errors are added to x_1, other 20 gross errors are added to x_2 and so on, totalizing 200 gross errors. The first set of 100 measurements is equal of the second set of 100 measurements.

At each time step, for one new measured value read, leads to a 'drop' of the oldest measured in the data horizon. The PSO parameters are the usual values: w=0.7298, c1=2.05, c2=2.05. The simulation used 40 particles (np) and 150 iterations (ni) each step.

The results of a typical simulation run are presented in Table 2, where gbestx is the global best value of the measured values (reconciled data), xmeas are the raw values, p_out is the outlier position indicative, gbestu the global best value of unmeasured variables.

Table 2 – Typical Tests Results of DRPE problem using PSO algorithm

step= 45	x_1	x_2	x_3	x_4	x_5
gbestx	4.501496	5.582011	1.936328	1.453960	4.852185
xexact	4.512400	5.581900	1.926000	1.456000	4.854500
xmeas	4.565699	5.685185	4.815000	1.492211	4.850821
xexact-gbestx	0.010904	0.000111	0.010328	0.002040	0.002315
xmeas-gbestx	0.064203	0.103174	2.878672	0.038251	0.001363
p_out	0	0	1	0	0
	u_1	u_2	u_3		
gbestu	11.014648	0.607275	2.063209		
uexact	11.070000	0.614670	2.050400		
uexact-gbestu	0.055352	0.007395	0.012809		
gbest= 0.075370					
	PSO iter. time (s): 75.81 acum. time (s): 3393.89				

The results show that even in the presence of outliers the estimated values are very close of the true values. All outliers are corrected detected showing an efficiency of 100 %. A typical test run takes about 75 seconds to execute and solve the DRPE problem in a Core 2 Duo machine, 1.6 GHz. The complete test of calculating the 100 DRPE problems simultaneously with gross errors and outlier detection takes about 2:10 h., which indicated a potential performance to use the PSO algorithm to online DRPE problem.

6. CONCLUSIONS

The Particle Swarm Optimization Algorithm was tested in a DRPE problem and the results showed that the PSO can handle the non-differentiable characteristics of the redescending estimator, the method do not need complex calculations and is simpler than GA implementation. The results also showed that PSO applied to DRPE is faster than the modified GA [13], do not need special constants adjustments and presents another perspective in solving the DRPE and gross errors detection, even in real time DRPE online calculations.

Some issues to future work are the automatic adjustments of the constants a, b, and c of the redescending estimator using the PSO algorithm and the code optimization for faster calculations and adaptations to real-time monitoring and solution of the DRPE problem.

REFERENCES

1. Morad K.,. Young B. R., Svrcek W. Y.," Rectification of plant measurements using a statistical framework",*Computers and Chemical Engineering,* **29** ,919–940,(2005).
2. S.Streit, M. langenstein, B. Laipple, H.Eitschberger,"A New Method for Evaluation and Correction of Thermal Reactor Power and Present Operational Applications", *Proceedings of ICONE13 13th. International Conference on Nuclear Engineering, Beijing, China, may 16-20, (2005).*
3. T.A. Soderstrom, D.M. Himmelblau,T.F. Edgar,"A mixed integer optimization approach for simultaneous data reconciliation and identification of measurement bias", *Control Engineering Practice*,**v.9**, pp. 869-876, 2000.
4. I.B. Tjoa, L.T, Biegler, "Simultaneous Strategy for Data Reconciliation and Gross Error Detection of Nonlinear Systems", *Computers and Chemical Engineering*. **v. 15(10)**, pp. 679-690
5. M. Kong, B. Chen, X. He, S. Hu,"Gross error identification for dynamic system", *Computers and Chemical Engineering*, **v.29**, pp 191-197 (2004).
6. Kuhen, D.R., Davidson, H., 1961, "Computer control. II. Mathematics for control",*Chemical Engineering Progress*, **v. 57**, pp. 44-47.
7. Serth, R.W., Heenan, W.A., 1986 "Gross errors detection and data reconciliation in steam-metering systems", *AIChE Journal*, **v. 32**, pp. 733-742.
8. Crowe, C.M., Campos, Y.A.G., Hrymak, A, 1983, "Reconciliation of Process Flow Rates by Matrix Projection I: Linear Case", *AIChE Journal*, **v. 29**, pp. 881-888
9. Narasimhan, S., Harikumar, P., 1993, "Method to incorporate bounds in data reconciliation and gross error detection - the bounded data reconciliation problem",*Computers & Chemical Engineering*, **v. 17**, pp. 1115-1120.
10. Pai, C. C. D., & Fisher, G. D. (1988). Application of Broyden's method to reconciliation of nonlinearly constrained data. *American Institute of Chemical Engineering Journal*, ***34*****(5)**, 873.
11. Zhao, P., & Jiang, W. (1996). Application of stochastic search for gross error detection and data reconciliation. In *Proceedings of The IEEE International Conference on Industrial Technology* (pp. 728–730).
12. Moros, R., Kalies, H., Rex, H. G., & Schaffarczyk, S. (1996). A genetic algorithm for generating initial parameter estimations for kinetic models of catalytic processes. *Computers and Chemical Engineering*, *20*(10), 1257–1270.
13. Wongrat, W. Srinophakun, T.,Srinophakun, P., 2005, "Modified genetic algorithm for nonlinear data reconciliation", *Computers & Chemical Engineering*, v. 29, pp. 1059-1067.
14. Wasanapradit, T. (2000). Solving nonlinear mixed integer programming using genetic algorithm. Master Thesis, King Mongkut University of Technology Thonburi, Bangkok, Thailand. Available: fengtcs@ku.ac.th, cited in [13].
15. N. Arora, L.T. Biegler (2001).,"Redescending estimators for Data Reconciliation and Parameter Estimation", *Computers and Chemical Engineering*., **v.25**, pp.1585-1599.
16. Feldman, R. N. (2007)., "Reconciliação de dados em tempo real para monitoração e detecção de falhas em terminal de transporte e armazenamento de derivados de petróleo" , COPPE/UFRJ, M.Sc. Thesis, Chemical Engineering Department.
17. Özyurt, D. B., Pike, R. W., 2004, "Theory and Practice of Simultaneous Data Reconciliation and Gross Error Detection for Chemical Process", *Computers & Chemical Engineering*, **v. 28**, pp. 381-402.
18. Akaike, H. (1974). A new look at the statistical model identification.*IEEE Transactions on Automatic Control*, **AC-19(6)**, 716.
19. Yamamura, K., Nakajima, M., & Matsuyama, H., 1988,"Detection of gross errors in process data using mass and energy balances" *International Chemical Engineering*, ***28*****(1)**, 91.
20. Y. Shi and R. Eberhart (1998). A Modified Particle Swarm Optimizer,*Proceedings of the 1998 IEEE Congress on Evolutionary Computation*, Anchorage, AK.
21. J.Kennedy and R.C. Eberhart, Particle Swarm Optimization[C], Proc. on feedback mechanism IEEE Int'l. Conf. on Neural Networks, **v. VI**, 1942-1948, 1995.
22. Braton, D., and Kennedy J., Defining a Standard for Particle Swarm Optimization, *Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007).*
23. Kennedy, J., Some Issues and Practices for Particle Swarms, Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)
24. Prakotpol, D., & Srinophakun, T. (2003). GAPinch: Genetic algorithm toolbox for water pinch technology. *Chemical Engineering and Processing*,*43*(28), 203–217.l
25. Zhou, A New Method to Solve Robust Data Reconciliation in Nonlinear Process, *Chinese J. Chern. Eng.,* **14(3)**,357-363 (2006)
26. VDI 2048 Part 1, "Uncertainties of measurement during acceptance tests on energy-conversion and power plants-fundamentals", October 2000.