Application of a genetic simulated annealing algorithm for data reconciliation

Bundit Kottititum [1], Phonphimon Lotangtrakun [2], Thongchai Srinophakun [1, 2*]
[1]Department of Chemical Engineering, Faculty of Engineering, Kasetsart University
[2]Center of excellence on petroleum Petrochemicals and Materials Technology
E-mail : fengtcs@hotmail.com*

Abstract

To elucidate the propagation of error or bias from the measurement of a process, the data requires reconciliation. In this article, a genetic simulated annealing algorithm (GSA) based program is proposed for solving data reconciliation (DR) problems. The proposed GSA utilizes simultaneous simulated annealing and modified cross-generational probabilistic survival selection (CPSS) in a genetic algorithm. Validation is performed with linear and nonlinear DR problems. The test starts with the study of appropriate GSA parameters of constraint problems. The performance of GSA with the appropriate parameters is then compared to the genetic algorithm (GA) method, the specific method, and the commercial software DATACON. The proposed GSA, with its ability to give more accurate reconciled data, is a promising choice as an optimization tool for data reconciliation problems.

Key words

data reconciliation, genetic simulated annealing algorithm, optimization

1. Introduction

Data on the real state of physical variables and processes can be obtained from measurements; these, however, are only estimates. Process data measurements are generally corrupted with two types of errors–random and gross–causing violation of process constraints defined by the mass and energy balances [1]. Therefore, the elimination of such errors would improve the accuracy of measured data. Data reconciliation adjusts process measurements with random errors by supplying satisfactory material and energy balance constraints. Elimination of the less frequent gross errors, which can severely bias the estimates and the reconciled data, is achieved by gross error detection.

Therefore, simultaneous data reconciliation and gross error detection have emerged [2]. In chemical engineering, Kuehn and Davidson [3] proposed the first solution to the steady-state data reconciliation problem. Later, Crowe et al. [4] investigated measured and unmeasured variable linear systems using matrix projection. A solution of the nonlinear data reconciliation problem via successive linearization is described by Knepper and Gorman [5]. Tjoa and Biegler [6] proposed an efficient hybrid successive quadratic programming algorithm to solve the bi-variable objective function of DR problems. Sánchez and Romagnoli [7] applied Q-R factorization to analyze, decompose and solve linear and bilinear reconciliation problems. To avoid complex calculations resulting in a complicated process, Zhao and Jiang [8] proposed a stochastic search method for solving the linear steady-state DR problem. The significant advantage of this method is that it does not depend on any particular model structure and only requires simple algebraic calculation.

A popular stochastic search, genetic algorithm (GA), is widely used to solve optimization problems. Wongrat [9] developed GA for the DR problem defined by a redescending estimator and weighted least squares. This GA can solve the DR problem without the complex calculations required by conventional optimization methods, but the calculation time is longer and the data result is not satisfactory for complex systems. GA provides a solution to the complicated problem of discontinuous and non-convex objective functions determined without using derivatives. A notable characteristic of GA is that it can find the region of optimal values rapidly; however, the ability to accurately search in this region is not satisfactory for complex systems [10].

For this reason, many researchers have tried to design new hybrid algorithms by combining GA with other optimization algorithms. Simulated annealing (SA) is an approach for simulating the thermodynamic process of annealing (cooling of a solid). It has been observed that this method could also be used to search for feasible solutions of optimization problems with the objective of converging to an optimal solution. SA can search accurately in certain regions; however, it is difficult to explore the entire search space. Therefore, the difficulty of solving complex problems can be overcome by improving GA by combining it effectively with an SA algorithm to avoid the common defect of early convergence. This is known as a genetic simulated annealing algorithm (GSA).

GSA will be proposed in this article as the best method for handling the discontinuous and non-convex properties of objective function in steady-state DR problems. In the following section we describe the general formulation and objective function for DR. In Sections 3 and 4, the basic theory and procedure of using GSA for DR will be discussed. In section 5. DR problems are solved with GSA and also compared with other methods.

2. General formulation of the data reconciliation problem

The estimation of unmeasured variables, and model parameters, are obtained as part of the reconciliation problem. The estimation of unmeasured values based on the reconciled measured values is also known as data coaptation. In general form of data reconciliation problems.

$$\begin{aligned} &\min_{x,u} \rho\left(x^M, x\right) \\ &h(x,u) = 0 \\ &g(x,u) \le 0 \\ &x^L \le x \le x^U \\ &u^L \le u \le u^U \end{aligned} \qquad (1)$$

where: ρ is an objective function depending on the difference between the measurement of a variable and its value for any measured variables; EMBED Equation.3 is the set of measurement data of the corresponding variable x; u is the set of unmeasured variables; h is the set of equality constraints; g is the set of inequalities; and superscripts L and U are the lower and upper bounds of the variables.

The objective function usually uses a weighted least squares objective function. However if there are gross errors in the data, the objective function is biased, leading to incorrect data reconciliation. The common procedure is to identify and eliminate gross errors. Thus data reconciliation and gross error detection are applied together to improve the accuracy of measured data. This paper uses weighted least squares and a redescending estimator as objective functions – the weighted least squares method is used for data reconciliation problems, and a redescending estimator [11] for data reconciliation of measured data containing gross errors.

Objective Function for data reconciliation

We discuss the form of M-estimators for the objective function EMBED Equation.3 in Eq. (1). The M-estimators are defined as:

Least squares estimator

$$2c_F^2\left[\frac{|\varepsilon_i|}{c_F} - \text{In}\left(1 + \frac{|\varepsilon_i|}{c_F}\right)\right] \qquad (2)$$

where C_F = 1.3998

Hampel's redescending M-estimator

$$\begin{array}{ll} \frac{1}{2}\varepsilon_i^2, & 0 \le |\varepsilon_i| \le a \\ a|\varepsilon_i| - \frac{1}{2}a^2, & a < |\varepsilon_i| \le b \\ ab - \frac{a^2}{2} + (c-b)\frac{a^2}{2}\left[1 - \left(\frac{c-|\varepsilon_i|}{c-b}\right)^2\right], & b < |\varepsilon_i| \le c \\ ab - \frac{a^2}{2} + (c-b)\frac{a}{2}, & |\varepsilon_i| > c \end{array} \qquad (3)$$

where $c \ge b + 2a$

The consequence of this tuning requirement is that in data reconciliation and gross error detection, the performance of two different EMBED Equation.3 functions can be compared properly only for (nearly) equal efficiency cases. This means that, for instance, Fair function Eq.(2) with 95% efficiency can be compared with 95% efficient Hampel's redescending M-estimator in Eq.(3) and the value of QUOTE and c are 1.35, 2.7 and 5.4.

3. Genetic Simulated Annealing (GSA) Algorithm

In normal GA, only simple crossover, mutation and selection are used as the genetic operators; normal GA cannot embody a complicated evolutionary process. Because of the limitation of population size and number of generations, the algorithm can easily produce truncation error, which will lead to precocity. For this reason, many researchers have attempted to design new hybrid algorithms by combining GA with other optimization algorithms. All of these hybrid algorithms are able to improve the efficiency of GA to a certain degree.

SA is a powerful optimization technique which can theoretically converge asymptotically to the global optimum solution with probability '1' when the initial temperature is high enough and the temperature decrease is infinitely slow. The process of this evolutionary operation adopts the new values based on the Boltzmann rule.

$$\exp\left(\frac{-\Delta E}{kT}\right) \geq \text{random}(0,1) \qquad (4)$$

This rule is expressed in the following form: Here, EMBED Equation.3 is the difference of fitness value between the child generation and the parent generation, and T is the annealing temperature. The main drawback of SA in practice is that the parameters in SA are hard to control. Firstly, there is a contradiction between optimization efficiency and computation time in the selection of the annealing rate. If the cooling rate is excessively fast, the evolutionary operation may miss the extremity. However, the convergence rate of this algorithm will be very slow. Secondly, for every temperature, SA by itself cannot easily determine if it has reached the balanced state. This means the length of the Markov chain cannot be controlled easily. Moreover, when it is reflected to the algorithm, the timing of the Metropolis process is difficult to control. Selection of the initial temperature is another difficulty in the application of SA [10].

A simple integration method for GSA algorithms is to generate new individuals with GA; then these individuals are processed with SA, and the results are used as the initial individuals of the next generation.

4. Methodology

Procedure of Genetic Simulated Annealing Algorithm for data reconciliation

Selected data reconciliation problems have two kinds of variables: measured and unmeasured. But data reconciliation problems are only concerned with measured variables. Thus we need to eliminate unmeasured variables from the constraint equations. Since the unmeasured variables are present in the equations, the simplest strategy for solving the problem is to eliminate them from the constraints. This will not affect the objective function since it does not involve unmeasured variables. After that the reconciled values of unmeasured variables can be calculated using the original constraints. Thus, GSA will act as a tool for searching the solution space for only measured variables. Usually GSA can randomly produce solutions that satisfy equality constraints. GSA has also been used to solve constrained optimization problems. Although different methods for handling constraints have been suggested, penalty function methods have been most widely used. The GSA procedure for data reconciliation used in this work follows the steps below and show in the Fig. 1:

Step 1 Choose real coding to represent problem parameters and roulette wheel selection methods, intermediate recombination, and real value mutation. Typically a binary representation is used in GA, but this has disadvantages when applied to multidimensional high-precision numerical problems. In these cases, the binary GA becomes inefficient and solutions require a large number of computational operations. Real number representation can deal with this problem and is more understandable than binary representation.

Step 2 Assign appropriate GSA parameters consisting of:

- Mutation probability, Pm. Mutation operates on an individual (chromosome), producing offspring very different from the parent. Certain genes in the offspring chromosome are randomly altered with a mutation rate, Pm. Pm controls the rate at which new genes are introduced into the population. A low rate will prevent the introduction of potential genes, while a high rate will allow too much random perturbation. Mutations encourage a population that is converging onto some optimum to jump into a different part of the solution space, thus increasing the probability of detecting a different point leading to the global optimum solution. In this paper, multi-position mutation is used for the mutation operator.

- Crossover probability, Pc. Crossover is used to create two new chromosomes from two existing chromosomes selected from the current population using a crossover rate (PC). The crossover rate is defined as the ratio of the number of offspring produced in each population size.

This ratio controls the expected number of chromosomes which undergo the crossover operation. A higher crossover rate allows more exploration of the solution space and reduces the probability of a local optimum. However if this rate is too high, unnecessary computational time will be consumed in exploring unpromising regions of the solution space. For real number representation, crossover is determined by using a linear combination of two vectors, termed arithmetic crossover.

- Population size, N
- Generation number, g
- Decrease in temperature, α
- Initial temperature, T0
- Final temperature, Tf
- Annealing function

One way to decrement the temperature is a simple linear method. An alternative is a geometric decrement where , where $t = t\alpha$, where $\alpha < 1$.

From other research has shown that α should be between 0.8 and 0.99, with better results being found in the higher end of the range. Of course, the higher the value of α, the longer it will take to decrement the temperature to the stopping criterion.

And the annealing function is the function to generate new points for the next iteration. The choices are "annealingfast" and "annealingboltz"

- annealingfast : The step has length temperature, with direction uniformly at random. This is the default.
- annealingboltz : The step has length square root of temperature, with direction uniformly at random.

Step 3 Set generation number g=0, temperature T=T0.

Step 4 Find initial solution P(g) to population size N.

Step 5 If g > gmax, terminate.

Step 6 Evaluate each chromosome of P(g) from the objective function and evaluate the fitness function, which is normally used to transform the objective function value into a measure of relative fitness, while the selection algorithm selects populations for reproduction on the basis of their relative fitness. The selection process is used to choose chromosomes from the parent and the offspring to make up a new population with the same number as the existing population. Wasanapradit [13] adopted CPSS (cross-generational probabilistic survival selection) as the selection method.

Step 7 Perform reproduction on the population, crossover on random pairs of strings, and mutation on every string.

Step 8 Calculate energy (E) using the equation:

$$E = E\left(x^{(g+1)}\right) - E\left(x^{(g)}\right) \qquad (5)$$

where : $E(x^{(g+1)})$ = objective function value of new string

$E(x^{(g)})$ = objective function value of old string

If E is negative, then the new string is better than the old one, and is accepted for the next generation, if not recalculate in Step 1.

Step 9 Calculate the acceptance probability using the acceptance algorithm:

$$Pr_E = \exp\left(-E/T(g)\right). \qquad (6)$$

a. Generate a random number (r) between (0, 1).

b. If $r \le Pr_E$, accept new (child) chromosome in a new generation; if not, use the old (parent) chromosome.

Step 10 Evaluate strings in the new population. Set g = g+1, then lower T according to the schedule and go to Step 6.

5. Results and Discussion

5.1 Case study

In this paper, weighted least squares and a redescending estimator are used as the objective functions for data reconciliation problems. Both objective functions and constraint equations require very complex calculations for data reconciliation EMBED Visio.Drawing.11 using conventional optimization methods. Therefore, a genetic simulated annealing algorithm (GSA) is proposed as an optimization tool for data reconciliation problems in this work. We perform the GSA based on a typical weighted least squares objective function for simple data reconciliation

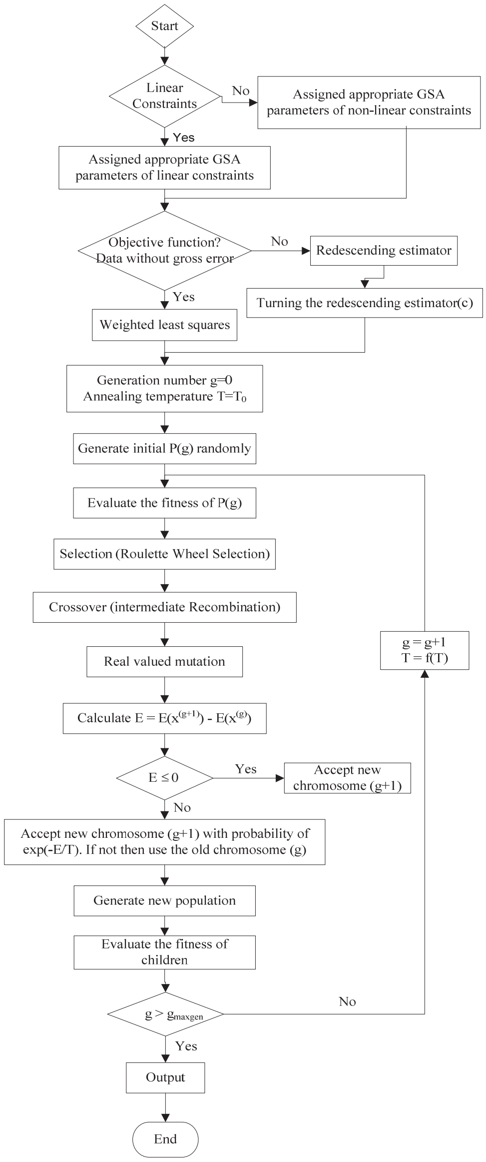


Fig. 1. Genetic simulated annealing algorithm (GSA).

problems where the measured data contain only random errors. (The objective function for data reconciliation problems in which the measured data contain random errors and gross errors is important because it can eliminate combinatorial procedure for gross error detection and use straightforward methods to identify them.) Moreover, gross error detection using statistical tests based on linear or linearized models is not suitable for nonlinear data reconciliation problems. The proposed code for data reconciliation problems is developed using MATLABTM. All examples have been solved using a computer with an AMD Athlon^{TM} XP 2000+ 1.67 GHz microprocessor and 256 MB of RAM.

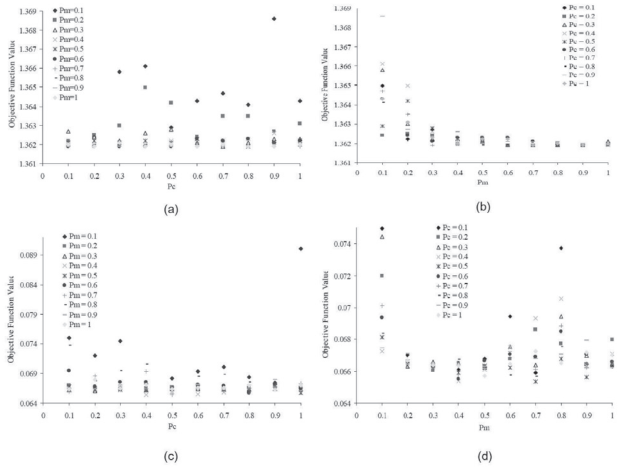
5.2 Appropriate GSA parameters

This section will investigate and select the appropriate parameters for GA and SA, which are considered in two cases as linear and nonlinear constraint problems (Tables 1). These parameters will be employed in the following case studies

As one can see in Fig.2 (a) and 2(c), we select the value of Pc that has a least scatter of Pm for linear constraint and nonlinear constraint (0.2 and 0.9). In the same way, in Fig. 2(b) and 2(d) the Pm at 0.9 and 0.3 is selected. We also find that the calculation time has no significant. The objective function value is 1.3619 and 0.06646.

Table 1. GSA parameters for problems

Parameter	Linear constraint	Nonlinear constraint
	Value	Value
Mutation probability, P_m	0.9	0.3
Crossover probability, P_c	0.2	0.9
Population size, N	40	40
Generation number, g	5000	5000
Decrease in temperature, α	0.99	0.99
Initial temperature, T_0	10000	10000
Final temperature, T_f	1.00E-02	1.00E-02



(a) (b)
(c) (d)

Fig. 2. The objective function value of problems: (a) linear constraint problem at fixed Pc and variances Pm (b) linear constraint problem at fixed Pm and variances Pc (c) nonlinear constraint problem at fixed Pc and variances Pm (d) nonlinear constraint problem at fixed Pm and variances Pc.

5.3 Simple linear data reconciliation

This problem is used to illustrate a number of basic features of data reconciliation. Fig. 3 shows four units with eight stream processes designed to perform a reconciliation of the mass flows.

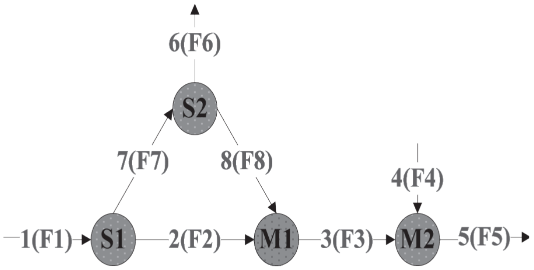


Fig. 3. Mass balance problem.

Let us assume that streams 7 and 8 are unmeasured, streams 1 and 3 are measured with a standard error of 1%, streams 2, 5 and 6 are measured with an error of 2%, and stream 4 is only approximately measured with a 5% error (Table 2).

The appropriate GSA parameters used in this case were formulated with linear constraints, and with weighted least squares as the objective function. A comparison of the performance of GA, DATACON and GSA is shown in Table 2.

The proposed GSA technique gives the best objective function value (1.3619), which is very close to the DATACON result. Therefore, the SA can support the GA and provide a significant search advantage. In addition, all three approaches have reconciled the data and calculated the values of streams 7 and 8.

Table 2. Comparison of simple linear data reconciliation methods

Stream	Standard deviation	Measured value	Calculation value		
			DATACON	GA	GSA
1	1.001	100.1	99.287	99.400	99.288
2	0.822	41.1	41.1	41.1	41.1
3	0.790	79.0	79.359	79.321	79.361
4	1.530	30.6	30.048	29.938	30.042
5	2.166	108.3	109.407	109.271	109.403
6	0.396	19.8	19.927	20.083	19.927
7	-	-	58.187	58.3	58.188
8	-	-	38.259	38.217	38.261
Objective function value			1.362	1.5548	1.3619
$F_1 - F_2 - F_7 =$			0	0	0
$F_7 - F_6 - F_8 =$			0.001	0	0
$F_2 + F_8 - F_3 =$			0	0.004	0
$F_3 + F_4 - F_5 =$			0	-0.012	0

Table 3. Redescending estimator tuning constant values for simple linear data reconciliation with error

	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	Best solution
C	40	20	16	8	4	2	0.8	0.4	1.65
n_{out}	0	0	0	0	0	0	2	2	1
OP	0	0	0	0	0	0	1	1	1
AVTI	0	0	0	0	0	0	1	1	0
AIC	4.648E+05	4.648E+05	4.648E+05	4.648E+05	4.648E+05	4.648E+05	38.899	55.285	38.087
F_1	107.942	107.942	107.942	107.942	107.942	107.942	98.849	97.508	99.139
F_2	41.100	41.100	41.100	41.100	41.100	41.100	41.100	41.100	41.100
F_3	80.548	80.548	80.548	80.548	80.548	80.548	78.970	77.700	79.179
F_4	29.184	29.177	29.180	29.183	28.770	31.997	29.376	30.600	30.876
F_5	109.732	109.725	109.730	109.731	109.318	112.545	108.346	108.30	110.054
F_6	20.576	20.576	20.576	20.576	20.576	20.576	19.879	19.808	19.960
F_7	66.842	66.842	66.842	66.842	66.842	66.842	57.749	56.408	58.039
F_8	46.266	46.266	46.266	46.266	46.266	46.266	37.870	36.600	38.079

Let us assume another set of data from this same system with an error induced in the measurement of stream 1, with a value of 110.1 recorded instead of the previous value of 100.1. In general, the engineer would not know that such an error existed in the raw data. In this case, a redescending estimator is used as the objective function in the comparison of GSA, simple GA and DATACON, as summarized in Table 3.

Table 3 shows that the M1 to M6 cannot detect the outliers (nout = 0). At M7 the outlier starts to be detected, which can be seen from a sharp decrease in the AIC value. M7 and M8 detect 100% of the measurements as being corrupted. This is due to the increase in type 1 errors with increasing robustness of the estimator. The AIC also increases from M6 to M8. A plot of the AIC with the values of C shows a minimum at M7, indicating that the best estimator lies close to it. With this knowledge the estimator is tuned to C = 0.81. A comparison with the other methods is shown in Table 4.

There is no significantly different result in steam 1 with the introduced gross error. Here, the proposed technique gives the objective function value (2.437), which is not close to the DATACON result (0.221) because of the influence of the redescending estimator objective function. However, simultaneous gross error detection and reconciliation of data are performed well with the redescending estimator. In addition, DATACON normally performs data reconciliation using a global test and a measurement test to detect gross errors before continuing with data reconciliation.

5.4 Binary distillation column

This problem is a simple example of reconciled flows and compositions of a binary distillation column with no gross error in measured data (Narasimhan & Jordache, [1]).

The flows and component mole fractions of feed (F), distillate (D), and bottom streams (B) are measured. The measured flow rates under noise are based on data from 100 measurements, with 5% standard deviation.

The measured mole fractions are found in the same way, with 1% standard deviation from true values. A set of the true values, measured values, and standard deviations is presented in Table 5.

If we represent the value of the flow of feed, distillate and bottom streams by the variables EMBED Equation.3 and the measured compositions by the variables EMBED Equation.3 , we will get one overall flow balance and two component balances, as follows:

$x_F - x_B - x_D = 0$

(8a) $x_F y_{F1} - x_B y_{B1} - x_D y_{D1} = 0$

(8b) $x_F y_{F2} - x_B y_{B2} - x_D y_{D2} = 0$

(8c) $y_{F1} + y_{F2} = 1$

(8d) $y_{B1} + y_{B2} = 1$

(8e) $y_{D1} + y_{D2} = 1$

(8f)

In this case, Wongrat [9] used a nonlinear constraint parameter with weighted least squares as the objective function in the comparison of GSA and GA. The result is shown in Table 5.

It can be seen that the GSA produces a better solution than the GA in Wongrat's study, in which the selection process was modified with CPSS. In principle, both techniques aim to improve the stage of selection of general GA, but the approach of simulated annealing in GA (GSA) seems to work more efficiently.

Table 4. Comparison of simple linear data reconciliation with error

I	Standard deviation	Measured value	Calculation value		
			DATACON	GA	GSA
1	1.101	110.1	98.694*	98.609*	99.139*
2	0.822	41.1	41.1	41.1	41.1
3	0.790	79.0	78.894	79.046	79.179
4	1.530	30.6	30.203	30.004	30.876
5	2.166	108.3	109.097	109.063	110.055
6	0.396	19.8	19.800	19.571	19.960
7	-	-	57.594	57.509	58.039
8	-	-	37.794	37.938	38.079
Objective function value			0.221	4.786	2.437
$F_1 - F_2 - F_7 =$			0.000	0.000	0.000
$F_7 - F_6 - F_8 =$			0.000	0.000	0.000
$F_2 + F_8 - F_3 =$			0.000	-0.008	0.000
$F_3 + F_4 - F_5 =$			0.000	-0.013	0.000

Table 5. Comparison of GA and GSA results for a binary distillation column

Stream	Variables	STD	Measured value	Calculation value	
				GA by Wongrat WLS	GSA WLS
	Flow	50.000	1006.3	1003.62	1003.530
Feed	Component 1 (%)	0.480	48.060	48.047	47.991
	Component 2 (%)	0.520	52.065	52.028	52.009
	Flow	24.731	497.72	496.261	496.262
Distillate	Component 1 (%)	0.950	95.119	95.145	95.003
	Component 2 (%)	0.050	4.998	4.998	4.997
	Flow	25.269	504.470	507.359	507.268
Bottom	Component 1 (%)	0.020	1.999	1.980	1.999
	Component 2 (%)	0.980	97.965	98.029	98.001
Objective function value			-	0.93273	0.06773
$x_F - x_B - x_D = 0$			4.038	0.000	0.000
$x_F y_{F1} - x_B y_{B1} - x_D y_{D1} = 0$			0.117	-0.004	0.001
$x_F y_{F2} - x_B y_{B2} - x_D y_{D2} = 0$			4.850	0.001	-0.001
$y_{F1} + y_{F2} = 1$			100.125%	100.075%	100%
$y_{B1} + y_{B2} = 1$			100.117%	100.143%	100%
$y_{D1} + y_{D2} = 1$			99.964%	100.009%	100%

Table 6. Refinery problem results – mass and energy balance

i	Variable	Standard deviation	Measured value	Calculation value	
				DATACON	GSA
1	Rate, ton(s)/h	6.600	330.000	322.970	322.951
	Temp, °C	1.00	154.00	154.662	154.723
2	Rate, ton(s)/h	-	-	131.696	131.537
	Temp, °C	-	-	154.662	154.723
3	Rate, ton(s)/h	-	-	191.274	191.414
	Temp, °C	-	-	154.662	154.723
4	Rate, ton(s)/h	2.640	132.000	131.696	131.537
	Temp, °C	1.00	170.00	170.492	170.531
5	Rate, ton(s)/h	3.840	192.000	191.274	191.414
	Temp, °C	1.00	185.00	185.339	185.376
6	Rate, ton(s)/h	-	-	322.970	322.951
	Temp, °C	-	-	179.339	179.388
7	Rate, ton(s)/h	-	-	322.970	322.951
	Temp, °C	1.00	221.00	219.346	219.464
7A	Rate, ton(s)/h	-	-	322.970	322.951
8	Rate, ton(s)/h	1.360	68.000	67.914	67.855
9	Rate, ton(s)/h	1.320	66.000	65.969	65.899
	Temp, °C	1.00	200.00	200.061	200.040
10	Rate, ton(s)/h	1.300	65.000	65.365	65.352
	Temp, °C	1.00	285.00	285.213	285.205
11	Rate, ton(s)/h	2.420	121.000	123.722	123.845
	Temp, °C	1.00	370.00	370.693	370.700
12	Rate, ton(s)/h	-	-	123.722	123.845
	Temp, °C	1.00	275.00	274.378	274.325
13	Rate, ton(s)/h	-	-	65.365	65.352
	Temp, °C	1.00	200.00	199.810	199.800
14	Rate, ton(s)/h	-	-	65.969	65.899
	Temp, °C	1.00	169	168.941	168.969
Objective function value				7.01896	7.0022

5.5 Simple refinery

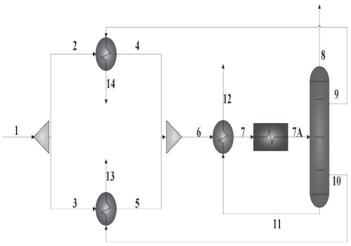


Fig. 4. Simple refinery.

The measured values for the case in Fig. 4 are shown in Table 6 with a 2% standard deviation of measurement. The product streams consist of one component per stream, except for the overhead Naphtha product which contains five pure components and one heavy pseudo-component. This complex case study has bilinear constraints. The weighted least squares method is used as the objective function in the comparison of GSA and DATACON. It can be seen that the simultaneous mass and energy constraint produces better results for complex DR from GSA (7.0022) than from DATACON(7.01896).

6. Conclusion

An optimization tool using GSA for DR problems has been presented. The GSA based program is the optimization tool for data reconciliation using two kinds of objective function. The weighted least square objective function is used for data reconciliation and the redescending estimators and the modified AIC objective function are used for data reconciliation of measured data contained gross error by developing code program in MATLABTM for the difficulty of complex problems. GSA has a better searching ability for the global optimal solution. GSA generates a new individual with GA; then these individuals are processed with SA, and the results are used as the initial individuals of the next generation. The systematic approach of GSA to the DR problem was proposed and implemented. This method started with appropriate GSA parameter formulation and then used these parameters to solve the DR problem. The steady-state DR problem was tested with this proposed program. The results showed that the GSA method could be a promising choice as an optimization tool for the DR problem, with the ability to give more accurate reconciled data.

7. Acknowledgement

This study was supported by the Center of excellence on petroleum Petrochemicals and Materials Technology, and Faculty of Engineering, Graduate School, Kasetsart University.

REFERENCES

1. Narasimhan, S. and Jordache, C., Data reconciliation and gross error detection. Houston, TX: Gulf Publishing (2000).

2. Özyurt, D.B. and Pike, R.W., Computers & Chemical Engineering, 28, 381–402 (2004).

3. Kuehn, D.R. and Davidson, H., Chemical Engineering Progress, 57(6), 44–47 (1961).

4. Crowe, C.M., Garcia Campos, Y.A., and Hrymak,A., American Institute of Chemical Engineering Journal, 29(6), 881–888 (1983).

5. Knepper, J.C. and Gorman, J.W., American Institute of Chemical Engineering Journal, 26(2),260 (1980).

6. Tjoa, I.B. and Biegler, L.T., Computers & Chemical Engineering, 15(10), 679–690 (1991).

7. Sánchez, M. and Romagnoli, J., Computers & Chemical Engineering, 20(5), 483–493 (1996).

8. Zhao, P. and Jiang, W., Application of stochastic search for gross error detection and data reconciliation, Proceedings of the IEEE International Conference on Industrial Technology, December 2–6, Shanghai, China, 728–730 (1996).

9. Wongrat, W., Srinophakun, T. and Srinophakun, P., Computers & Chemical Engineering, 29, 1059 -1067(2005).

10. Yao, R., Yang, B., Cheng, G., Tao, X. and Meng, F., Chemical Engineering Journal, 4150, 1–7 (2003).

11. Arora,N. and Biegler,L.T., Computers & Chemical Engineering, 25(11-12), 1585–1589 (2001).

12. Shimodaira, H., DCGA: a diversity control oriented genetic algorithm, 9th International Conference on Tools with Artificial Intelligence, November 3–8, Newport, CA, 367–374 (1997).

13. Wasanapradit, T., Solving nonlinear mixed integer programming using genetic algorithm. Master's thesis, King Mongkut's University of Technology Thonburi, 90 (2000).

NOMENCLATURE

AIC : Akaike information criterion
AVTI : Average number of type 1 errors
a, b, c : Tuning constants for the redescending estimator
DR : Data reconciliation
ΔE : Difference of fitness values between the child generation and the parent generation
F : Feed flow rate
GA : Genetic algorithm
GSA : Genetic simulated annealing algorithm
g : The set of inequalities constraints
h : The set of equality constraints
N : Number of variables
n : Total number of measured variables
nout : Total number of detected gross error
P : Probability
OP : Optimization of Probability
SA : Simulated annealing
u : The set of unmeasured variables
WLS : Weighted least squares
x^M : The set of measured variables
x : The set of reconciled values
M : Measured value
ε : The magnitude of random error
ρ : Some objective function dependent upon the difference between the measurement of a variable and its value for any measured variables

Superscripts

L : Lower bound of variable
U : Upper bound of variable

Subscripts

i : Variable index