Multiple bad data detection using genetic algorithms

Emtethal N. Abdallah, Amr A.Ghazala and Norhan Hanafy
Electrical Eng. Dept., Faculty of Eng., Alexandria University, Egypt

State estimation and bad data detection is a fundamental requirement for control and monitoring of electrical power systems. In this paper, a new approach for multiple bad data detection and identification using genetic algorithm is introduced. The identification problem is formulated by picking bad data from a set of suspected measurements in order to fulfill the requirements of maintaining observability and eliminating the minimum number of measurements that forced measurements residual below a threshold limit. The algorithm was applied to a case study which demonstrated high ability and robustness of the proposed algorithm in different multiple bad data types.

يعتبر تحديد حالة منظومة القوى الكهربية واكتشاف القياسات الخاطئه مطلب اساسى في تشغيل منظومة إدارة الطاقة للتحكم واستشعار امن المنظومة. يقدم هذا البحث تقنية جديدة لاكتشاف وتحديد القياسات الخاطئة المتعددة الضرورية لتحديد حالة منظومة القوى الكهربية وذلك باستخدام الخوارزم الجينى. تمت صياغة مشكلة اكتشاف القياسات الخاطئة في الخوارزم المقترح لالتقاط هذه القياسات من مجموعة القياسات المشتبه فيها وذلك للتحقق من بقاء الملا حظية وحذف اقل عدد من القياسات التي تدفع بواقي القياسات إلى قيمة اقل من الحد المشرف. تم تطبيق الخوارزم المقترح على منظومة قوى كهربية بسيطة وأوضحت النتائج فاعلية الخوارزم وقوته في اكتشاف أنواع مختلفة من القياسات الخاطئة المتعددة.

Keywords: State estimation, Bad data, Genetic algorithms

1. Introduction

Power System State Estimation (PSSE) is an essential tool in power system analysis. State Estimators (SE) are the heart of modern energy management systems. The performance of any other application program (e.g., security analysis, economic dispatch, optimal power flow, etc) strongly depends on the accuracy of data provided by the SE.

The ability to detect and identify bad measurements is a great benefit of PSSE. Transducers may have been wired incorrectly or the transducer itself may be malfunctioning so that it simply no longer gives accurate readings. If a measurement is grossly erroneous or bad, it should be detected and then identified so that it can be removed from the estimator calculations. Otherwise it will corrupt the accuracy of estimated system states.

The statistical properties of the measurement errors facilitate the detection and identification. Single bad data detection and identification is relatively an easy task [1], it can be detected by setting a threshold for the measurements residual $J(x)$, and identified by maximum individual measured-estimated difference. However, multiple bad data detection and identification is a complex process. This is due to the nature of interactive effect of multiple bad data on all related system states, which makes the identification process not an easy task. Many efforts were devoted to the issue of multiple bad data detection and identification during the years [2,3]. The successive elimination of suspected bad data based on the value of the Normalized Measurement Residual (NMR) is the most common approach of identification by elimination. In particular, the statistical criterion based on the (NMR) may have problems in correctly identifying and eliminating multiple interacting bad data especially when they are of the conforming type. In such a case, successive elimination of the measurement with the largest normalized residual may result in the suppression of correct measurements instead of the bad data [4].

A different approach to the bad data elimination problem relies on the use of solution algorithms which behave in a more robust way than weighted least square; some of them are based on the idea of minimizing a nonquadratic function of the measurement residuals [5]. The weighted absolute value

(WLAV) method, [6], belongs to this same category and has widespread popularity thanks to its automatic bad data rejection property [7]. However, nonquadratic state estimation is prone to convergence problems and is more computationally demanding with respect to least square estimators. More recently, non-deterministic approaches such as artificial neural network [8] have been applied and tabu search has been proposed as a viable strategy for the solution of the bad data identification problem [9].

Among non-deterministic methods, genetic algorithms are known to possess enough flexibility and generality to handle complex optimization problems and have been successfully applied in other fields of power system analysis. The genetic algorithm based procedures behave satisfactorily in identifying multiple bad data [10]; they also present the nice feature that a correct state estimation, if not the best solution, is often found since the early iterations of the procedure, thus enabling the operator to get a viable solution even before the end of the whole computation. This paper introduces multiple bad data detection technique using genetic algorithms.

2. Multiple bad data detection and identification technique

The magnitude of $J(x)$ indicates the presence of bad measurements. If the value of $J(x)$ exceeds the threshold, t_J, of the chi-square distribution of $J(x)$ at degree of freedom k and signification level a, there is a reason to suspect the presence of at least one bad measurement.

The identification of multiple bad data can be handled as an optimization problem of combinatorial nature. The suspected measurement set is represented by an m-dimensional decision vector, b, in which:

$b_i = 1$ if the ith measurement is a bad data.

$b_i = 0$ if the ith measurement is good.

Therefore, a system with m measurements has 2^m possible decision vectors where each decision vector will represent a possible combination of good and bad measurements. For any possible decision vector, the removal of bad measurements from the measurement set is checked by the chi-square test.

The problem of identification of bad data can be formulated as follows: For any given decision vector b, Set(b) denotes the corresponding measurement set assuming that only the "good" data are taken into account and the suspected bad data have been eliminated. After re-estimation according to the Set (b) measurement set, $x'(b)$ is the new state vector, $J[x'(b)]$ is the corresponding value of the residual, and t_J (b) is the updated detection threshold.

Bad data identification can now be formulated as the following combinatorial problem, in which the objective function (Obj. Fun.) is equal to the total number of suspected bad data:

$$\min F(b) = \sum_{i=1}^{m} b_i \,, \tag{1}$$

subject to:

Set (b) is observable. (2)

$$J[x'(b)] < t_J(b). \tag{3}$$

At the end of the optimization process, individual residuals can be checked to ensure optimal performance.

3. Application of genetic algorithms to the problem

By generating random initial population of binary bits, each individual of Genetic Algorithm (G.A) population could represent a possible solution to the problem. The individuals being selected according to their fitness are applied to crossover and mutation operators to create a new and improved population from the old one. By incorporating elitism, the string with the best fitness value is always preserved in the next generation.

3.1. Representation and initial population

Each individual of the genetic algorithm population is represented by a string of binary

bits. The initial population is generated randomly.

3.2. Fitness function (Fit)

As the identification of bad data problem is a constrained problem, the penalty technique is the most common technique used in G.As for constrained optimization problems, [11]. This technique transforms a constrained problem into an unconstrained problem by penalizing infeasible solutions, in which a penalty term is added to the objective function for any violation of the constraints. Consequently, the bad data identification problem may be reformulated as the unconstrained minimization of the following objective function:

$$\tilde{F}(b) = \sum_{i=1}^{m} b_i + P_1 J[x'(b)] + P_2 \,. \qquad (4)$$

Where P_1 is a penalization coefficient, while P_2 introduces a large penalization term when the measurement layout, corresponding to the decision vector b, makes the system unobservable. In the proposed procedures P_1 has been taken equal to 1 and P_2 equal to 1000.

Since genetic algorithms try to increase the fitness of their population, the problem of minimization of eq. (4) is converted to that of maximizing the function $1/\tilde{F}(b)$.

3.3. Genetic operators

In each generation (gen.) a new and improved population is generated from the old one by applying the three genetic operators, selection, crossover and mutation.

3.3.1. Selection

The new individuals are selected from the old population according to their fitness using the roulette wheel selection technique. The elitist strategy is used to guarantee that the best individual will exist in the next generation.

3.3.2. Crossover

One point crossover operator is used where the new individuals are created by combining substrings from the selected individuals and takes place according to crossover probability value.

3.3.3. Mutation

By using the conventional mutation operator, a new individual can be created by changing the value written in a random location of its string.

Fig. 1 depicts the genetic algorithm flow chart used to solve the bad data identification problem.

4. Case study

The program was implemented on a six bus sample power system shown in fig. 2, [1].

In order to test the program and finding a good set of the G.A parameter values, the following parameter values proved to be a good start [12]:

Population size (popsize) = 20, gene length = 62 (which is equal to the measurements number), crossover probability = 0.7, mutation probability = .02, maximum generation (maxgen) number = 100.

Introducing one bad measurement into the measurement set, P_{12} = -31.5 instead of 31.5 and rerun the program, the program succeeded to find this bad error after 35 generations. The program has been tested at other values of the crossover probability beginning from 0.7 to .99 and gave a better performance on crossover probability of 0.93. The mutation probability also has been tested beginning from the value of 0.007 to the value of 0.025 and the program gives the better performance on the value of 0.02.

Multiple bad data case is then investigated in both cases, non-interacting and interacting bad data (conforming and non-conforming type), where the measurements may influence each other. Twelve tests were run to investigate the technique performance in different cases. In test 1, two bad measurements are introduced assuming that the values of these measurements are reversed. The two bad measurements are the real and reactive injected powers at bus #1, [P_1, Q_1]. The actual per unit values are P_1 = 1.079 and Q_1 = 0.16 and the assumed bad measurements are -1.131 and -0.202, respectively.

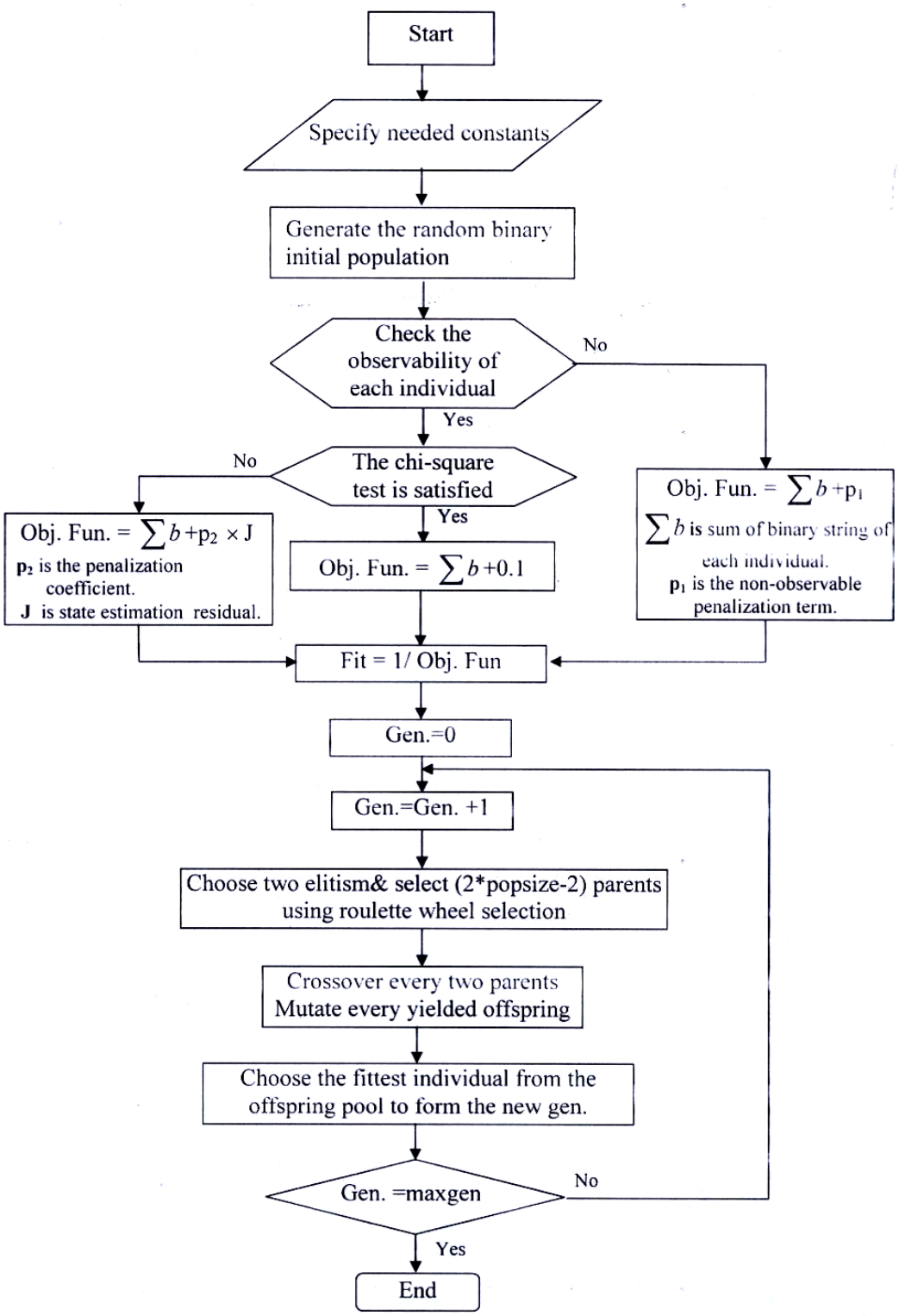

Fig. 1. G.A flow chart for bad data detection.

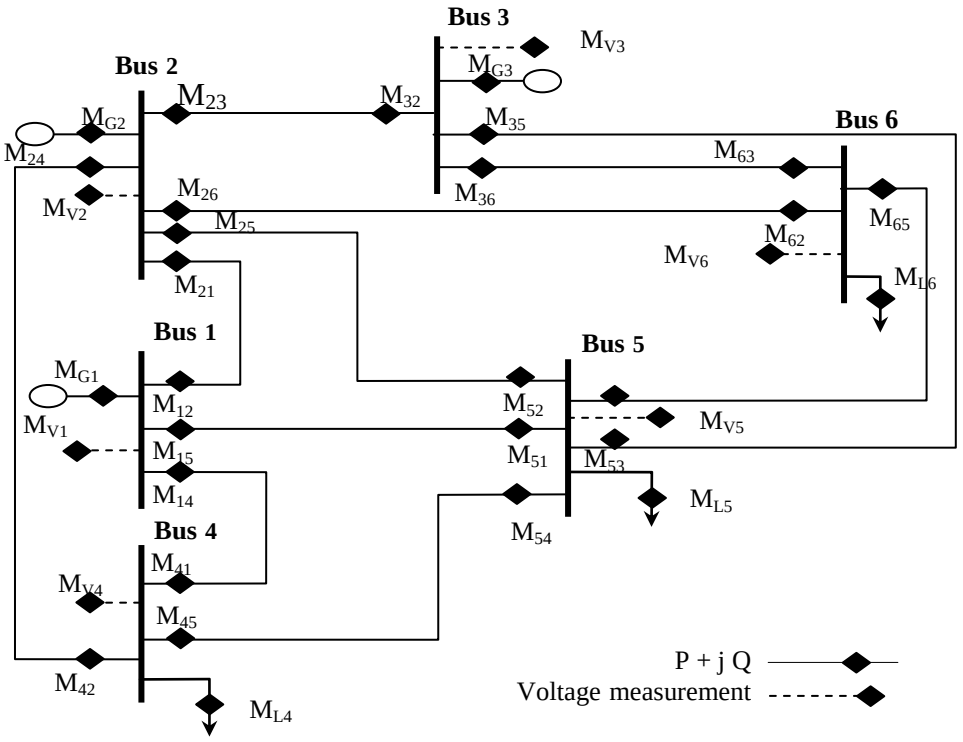

Fig. 2. Case study sample power system.

Running the state estimator gives a residual $J(x)$ value of 1024.6 which indicates that there are bad measurements within the measurement set. The G.A program identifies and eliminates one bad data measurement, the real injected power P_1. The value of the residual $J(x)$, after eliminating the measurement P_1 from the measurement set, is equal to 69.61. While the corresponding threshold t_J = 76.15. The presence of the other bad measurement, Q_1, is not in gross error that makes the value of $J(x)$ to be greater than the threshold t_J. Thus the program doesn't detect the presence of that error. The estimated per unit values of P_1 and Q_1 are 0.984 and -0.0197 respectively. Fig. 3 shows a sample plot for the convergence of the fitness value during generations running. A system free from errors is obtained at the 5th generation but with the elimination of 25 measurements, considered as suspect measurements, from the measurement set. Then the number of suspected measurements to be eliminated is reduced until reaching the optimal solution (O.S) at the 43rd generation. Table 1 through table 3 shows the output results for different cases.

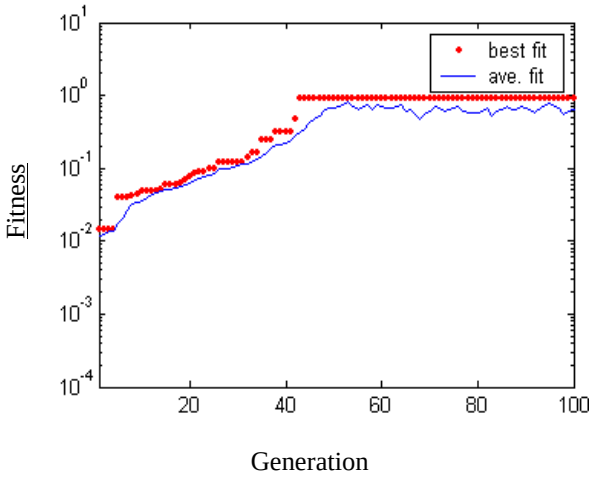

Fig. 3. The best fitness and average fitness obtained at each generation in test 1.

Table 1
Non-interacting multiple bad data identification

Introduced bad measurements	Actual values	Error values	Estimated values	J_{before}*	J_{after}*	t_J	Notes
Test 1							
P_1	1.079	-1.131	0.984	1024.6	69.61	76.15	P_1 was detected O.S was obtained at gen.= 43
Q_1	0.16	-0.202	-.0197				
Test 2							
P_{41}	-0.425	-0.8	-0.435	116.95	68.188	76.15	P_{41} was detected O.S was obtained at gen. = 38
Q_{41}	-0.199	-0.5	-0.237				
Test 3							
P_{12}	0.287	0.65	0.3	239.85	60.372	74.92	P_{12} and P_{24} were detected O.S was obtained at gen. = 49
P_{24}	0.331	-0.328	0. 33				
P_{53}	-0.18	-0.42	-0. 19				
Test 4							
P_{24}	0.331	-0.328	0. 324	531.98	37.77	73.68	All bad data were detected O.S was obtained at gen. = 40
Q_{24}	0.461	-0.383	0. 466				
P_{35}	0.191	-0.177	0.193				
Q_{35}	0.232	-0.239	0. 231				
Test 5							
P_{12}	0.287	-0.315	0.303	913.65	62.63	72.44	All bad data were detected except Q_{12} and P_{45} O.S was obtained at gen. = 48
Q_{12}	-0.154	0. 132	- 0.124				
P_{45}	0.041	-0.007	0.043				
Q_{45}	-0.049	0.174	- 0.047				
P_{36}	0.438	-0.433	0.433				
Q_{36}	0.607	-0.583	0.569				

* J_{before} :Value of state estimation residual before eliminating the bad measurements from the measuring set
* J_{after} : Value of state estimation residual after eliminating the bad measurements from the measuring set
O.S: Optimal solution

Table 2
Interacting multiple bad data identification, non-conforming type

Introduced bad measurements	Actual values	Error values	Estimated values	J_{before}*	J_{after}*	t_J	Notes
Test 6							
P_{12}	0.287	-0.315	0.306	503.7	66.83	73.68	All bad data were detected except Q_{12} O.S was obtained at gen. = 46
Q_{12}	-0.154	0.132	-0.123				
P_{14}	0.436	-0.389	0.45				
Q_{14}	0.201	-0.212	0.23				
Test 7							
P_{21}	-0.278	0.349	-0.288	677.87	51.87	71.2	All bad data were detected except Q_{21} O.S was obtained at gen. = 47
Q_{21}	0.128	-0.097	0.122				
P_{24}	0.331	-0.328	0.326				
Q_{24}	0.461	-0.383	0.456				
P_{25}	0.155	-0.174	0.156				
Q_{25}	0.154	-0.22	0.144				
Test 8							
P_{32}	-0.029	0.021	-0.026	848.47	49.89	72.44	P_{35}, Q_{35}, P_{36}, and Q_{36} were detected O.S was obtained at gen. = 34
Q_{32}	0.057	-0.102	0.047				
P_{35}	0.191	-0.177	0.193				
Q_{35}	0.232	-0.239	0.22				
P_{36}	0.438	-0.433	0.435				
Q_{36}	0.607	-0.583	0.575				
Test 9							
P_{51}	-0.345	0.366	-0.349	355.38	66.72	73.68	All bad data except P_{56} were detected O.S was obtained at gen. = 34
Q_{51}	-0.135	0.175	-0.133				
P_{52}	-0.15	0.117	-0.142				
Q_{52}	-0.18	0.222	-0.172				
P_{56}	0.016	0.021	0.02				
Q_{56}	-0.097	0.08	-0.099				
Test 10							
P_{41}	-0.425	0.401	-0.441	722.64	51.362	72.44	All bad data were detected except P_{45} and Q_{45} O.S was obtained at gen. = 34
Q_{41}	-0.199	0.143	-0.206				
P_{42}	-0.316	0.298	-0.314				
Q_{42}	-0.451	0.443	-0.431				
P_{45}	0.041	-0.007	0.042				
Q_{45}	-0.049	0.174	-0.042				

Table 3
Interacting multiple bad data identification, conforming type

Introduced bad measurements	Actual values	Error values	Estimated values	J_{before}*	J_{after}*	t_J	Notes
Test 11							
P_2	0.50	0.968	0.351	163.15	72.19	73.68	P_2 , Q_2, and P_{24} were detected O.S was obtained at gen. = 51
Q_2	0.744	1.44	0.70429				
P_{24}	0.331	0.656	0.285				
Q_{24}	0.461	0.766	0.503				
Test 12							
P_5	-0.70	–1.26	-0.523	162.81	61.39	72.44	All bad data were detected except Q_5 and Q_{51} O.S was obtained at gen. = 42
Q_5	-0.7	–1.015	-0.916				
P_{51}	-0.345	–0.641	-0.31				
Q_{51}	-0.135	-0.306	-0.188				
P_{53}	-0.18	-0.439	-0.137				
Q_{53}	-0.261	-0.523	-0.306				

4.1. Non-interacting multiple bad data identification

Tests from 1 to 5 show that the program successfully identifies most of the non-interacting bad measurements even in the worse seldom case when a six bad measurements are involved with the measurement set. Thus the best state estimate of the system could be obtained by running the state estimator with the remaining measurements after eliminating the identified bad measurements. Some bad measurements are introduced by assigning negative values to these measurements. This type of bad measurements may occur from connecting the transducers up backwards, thus, giving the negative of the values being measured.

In some cases, as test 1 and 3, when the measurements are not grossly in error, the program couldn't sense its existence due to limited increase in the residual value. Although they are not identified as bad measurements, the state estimator gives accepted values for these measurements. This is the role of the state estimator to relieve any small errors that exist within the measurement set. The large number of error measurements in test 5, although rarely happened, shows the robustness of the introduced algorithm.

4.2. Interacting multiple bad data identification – non conforming type

If more than one bad measurement is incident on the same bus, but they are not congruent, they will be interacting of non conforming type. The tests from 6 to 10 in table 2 give examples of that kind of error. As an illustration, in test 6 the errors happening in reading the power flows P_{12}, Q_{12}, P_{14} and Q_{14} would give a decrease in the real and reactive powers injected at bus one, while the reading measurement of the real and reactive injected powers at that bus indicate that there is no change in the injected powers. Then the errors are not consistent and of non conforming type. Due to contradiction of error types, it was a relatively easy task to detect and identify most of the bad measurements, except when the measurements are not grossly in error, the program couldn't sense its existence due to limited increase in the residual value.

4.3. Interacting multiple bad data identification – conforming type

If more than one bad measurement is incident on the same bus with conforming values, they will be interacting of conforming type. The tests from 11 to 12 in table 3 give examples of that kind of errors. Although bad data identification is more difficult in this case, the proposed technique was able to detect and identify most of the bad measurements.

5. Conclusions

A new algorithm for multiple bad data detection and identification is proposed. The proposed algorithm depends on normal threshold detection method through a threshold t_J based on chi-square distribution. The identification method was formulated as a combinatorial optimization problem where the objective is to minimize the number of bad data forcing the residual to be below the threshold value. Constraints of non-feasible solutions were handled using penalty factor technique. The G.A handled the problem very robustly in all cases, multiple interacting and non interacting bad data, conforming and non conforming types. Test examples showed the success of the algorithm in majority of the cases.

References

[1] J. Wood, "Power Generation Operation and Control", John Wiley and Sons (1984).

[2] H.M. Merrill and F.C. Schweppe, "Bad Data Suppression in Power System Static State Estimation," IEEE Trans. Power App. System, vol. PAS-90, pp. 2718–2725 (1971).

[3] L. Mili, Th. Van Custem and M. Ribbens-Pavella," Bad Data Identification Methods in Power System State Estimation: A Comparative Study," IEEE Trans. Power Apparatus and Systems, vol. PAS-104, (11), pp. 3037-3049 (1985).

[4] A. Monticelli et al.,"Multiple Bad Data Identification for State Estimation by Combinatorial Optimization," IEEE Trans. Power Delivery, Vol. PWRD-1, pp. 361–369 (1986).

[5] R. Baldick et al., "Implementing Nonquadratic Objective Functions For State Estimation And Bad Data Rejection," IEEE Trans. On Power System, Vol. 12, pp. 376–382 (1997).

[6] A. Abur, "A Bad Data Identification Method for Linear Programming State Estimation," IEEE Trans. On Power System, Vol. 5, pp. 894-901 (1990).

[7] W.W. Kotiuga and M. Vidyasagar, "Bad Data Rejection Properties of Weighted Least Absolute Value Techniques Applied to Static State Estimation," IEEE Trans. Power App. System, Vol. PAS-101, pp. 844-853 (1982).

[8] N.H. Abbasy and W. El-Hassawy, "Power System State Estimation: ANN Application to Bad Data Detection and Identification", Proc. 4-th IEEE AFRICON Conference, Vol. 2, pp. 611-615 (1996).

[9] A. Monticelli, State Estimation in Electric Power systems. A Generalized Approach, Boston: Kluwer Academic Publishers, (1999).

[10] S. Gastoni, G.P. Granelli, and M. Montagna, "Multiple Bad Data Processing by Genetic Algorithms" IEEE Bologna Power Tech Conference, June (2003).

[11] Z. Michalewicz, "Genetic algorithms + Data Structures = Evolution Programs", Springer, Berlin (1999).

[12] Amr A. Ghazala, Emtethal Negm, Norhan Hanafy, "Power System State Estimation Using Genetic Algorithms", Proceeding of the Tenth International Middle East Power Systems Conference, (MEPCON 2005), Suez Canal University, Port Said Egypt, pp. 669-676 (2005).

Received December 4, 2005
Accepted January 30, 2006