International Journal of Innovative Computing, Information and Control
Volume **x**, Number **x**, **x** **2008**
 ISSN 1349-4198
pp. **0–0**

Multiple Solutions for a System of Nonlinear Equations

Crina Grosan
Department of Computer ScienceSchool of Technology
Babes-Bolyai University
Kogalniceanu 1, Cluj-Napoca 400084, Romania
cgrosan@cs.ubbcluj.ro

Ajith Abraham
Centre for Quantifiable Quality of Service in Communication systems
Centre of excellence
Norwegian University of Science and Technology
Trondheim, Norway
ajith.abraham@ieee.org

Received August 2007; revised February 2008

ABSTRACT. *Finding several feasible solutions for a constrained nonlinear system of equations is a very challenging problem. Fundamental problems from engineering, chemistry, medicine, etc. can be formulated as a system of equations. Finding a solution for such a system requires sometimes high computational efforts. There are situations when these systems having multiple solutions. For such problems, the task is to find as many solutions as possible. This task can be complicated by adding several inequalities and/or variable bound constraints. In this paper, we deal with such systems of equations, which have multiple solutions and we try to solve them using two different approaches. Both approaches transform the problem into an optimization problem. One approach uses a line search based technique and the other one an evolutionary algorithm technique. Several experiments are performed in order to emphasize the advantages and disadvantages of the two methods.*

Keywords: Polynomial systems, multiple roots, optimization, line search, evolutionary algorithms

1. **Introduction.**

A nonlinear system of equations is defined as:

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix}$$

where $x=(x_1, x_2,\ldots, x_n)$, $f_1,\ldots,f_n$ are nonlinear functions in the space of all real valued continuous functions on $\Omega = \prod_{i=1}^{n} [a_i, b_i] \subset \Re^n$.

Some of the equations can be linear, but not all of them.Finding a solution for a nonlinear system of equations $f(x)$ involves finding a solution such that every equation in the nonlinear system is 0:

$$
\begin{cases}
f_1(x_1, x_2, ..., x_n) = 0 \\
f_2(x_1, x_2, ..., x_n) = 0 \\
\quad \vdots \\
f_n(x_1, x_2, ..., x_n) = 0
\end{cases}
$$

The assumption is that a zero, or root, of the system exists. The solutions we are looking for are those points (if any) that are common to the zero contours of f_i, i=1,...,n.

Polynomials are popular in curve and surface representations and many critical problems arising in computer aided geometric design such as surface interrogation are reproduced to find a zero set of a system of nonlinear equations [19].

There are several ways to solve nonlinear equation systems. Probably the most famous techniques are Newton type techniques. Other techniques are: Trust-Region Method, Broyden method, Secant method and Halley method [3][4][5][10]. Some recent work can be found in [1][2]. Fekik et al. [7] implemented neural-network based system identification techniques for nonlinear systems. A direct adaptive fuzzy backstepping control approach for a class of unknown nonlinear systems is developed in [27]. Most of the root finding algorithms experience difficulties in dealing with roots with high multiplicity such as performance deterioration and lack of robustness in numerical computation. Newton methods, for instance, require a good initial approximation of the roots of the system and fail to provide full assurance that all roots have been found [19].

Most of the times, the solution of a system of equations is not unique. Several practical problems require finding multiple solutions for a system. We refer to this type of equations systems in this research. Not only the problem of computing all solutions for a nonlinear constrained system of equations is NP-hard, but it is also possible that there exist exponentially many such solutions [14][16]. Also, simply checking if a solution exists is NP-hard [15][16].

In this paper, we develop two approaches, which treat the problem in two different ways. The first approach is a line search based technique, which is able to obtain one approximate solution in one run. This technique is applied several times in order to get multiple solutions [8][9][12]. The second approach transforms the system into a multiobjective optimization problem [10] [11] and a population based meta-heuristic (evolutionary algorithm) is then applied. Pareto dominance concept is used and a set of feasible solutions (Pareto optimal) are obtained in a single run. Rest of the paper is organized as follows. Section 2 presents the two optimization techniques. Experiment results, analysis and discussions are provided in Section 3. Finally, conclusions are provided towards the end.

2. Optimization Techniques Used

A modified line search and evolutionary algorithms are used and the way in which they treat the problem is presented in detail in the following sub-sections.

2.1. Line search

It is known that Line Search (LS) technique uses a starting point. There are also versions which allow the use multiple points and the search starts separately from each of these points. In the proposed approach, multiple arbitrary starting points are used. Each point is randomly generated over the definition domain $[\min_1, \max_1] \times [\min_2, \max_2] \times \ldots \times [\min_n, \max_n]$

For direction, we use a random value between -0.5 and 0.5.

The step value is given by $2+\frac{3}{2^{k^2}+1}$, where k denotes the iteration number.

After a given number of iterations, the search process is restarted. In order to restart the algorithm, the best result obtained in the previous set of iterations is taken into account and by following the steps given below:

- Among all the considered points, the solution for which the objective function is obtaining the best value is selected. If there are several such solutions, one of them is randomly selected. This solution is a multi-dimensional point in the search space and denoted by x for an easier reference.
- For each dimension i of the point x, the first partial derivative with respect to this dimension is calculated. This means the gradient of the objective function is calculated which is denoted by g.

  Taking this into account, the bounds of the definition domain for each dimension is re-calculated as follows:

  if $g_i = \frac{\partial f}{\partial x_i} > 0$ then $\max_i = x_i$;

  if $g_i = \frac{\partial f}{\partial x_i} < 0$ then $\min_i = x_i$
- The search process is re-started by re-initializing a new set of arbitrary points but between the newly obtained boundaries (between the new $\max_i$ or new $\min_i$).

The line search is a very useful optimization tool. Therefore, the equations system is transformed into an optimization problem as follows [15][16][21][22]:

minimize $\sum_{i=1}^{n} f_1^2(x)$.

2.2 Multiobjective Evolutionary approach

The Evolutionary Algorithm (EA) approach transforms the system of equations into a multiobjective optimization problem as follows:

$$\text{Minimize} \begin{bmatrix} abs(f_1(x)) \\ abs(f_2(x)) \\ \vdots \\ abs(f_n(x)) \end{bmatrix}$$

We generate the initial solutions over the given problem domain. These solutions are then evolved in an iterative manner. In order to compare the solutions, Pareto dominance relationship is used [26]. Real encoding of solutions, tournament selection, convex crossover and Gaussian mutation are used [9][24][27][25]. An external set is used for storing all the non-dominated solutions found during the iteration process. Tournament selection is applied. n individuals are randomly selected from the unified set of current population and external population. Out of these n solutions the one which dominated a greater number of solutions is chosen. If there are two or more 'equal' solutions then one of them is picked at random. At each iteration, this archive is updated by introducing all the non-dominated solutions obtained at the respective step and by removing all solutions that might become dominated.

3. Experimental Results and Analysis

We consider 5 systems of equations having between one and nine solutions. In Table 1, the details of these systems are provided.

TABLE 1. Benchmarks used in experiments

Problem	Number of variables	Ranges
Brown	5	$[-2, 2]^5$
Bullard	2	$[5.49 \cdot e^{-6}, 4.553] \times [0.0021961, 18.21]$
Ferrais	2	$[0.25, 1] \times [1.5, 6.28]$
Himmelblau	2	$[-5, 5]^2$
Steady state CSTR	2	$[0, 1]^2$

Each algorithm was run 10 times. For the EA approach, all the nondominated sets obtained at the end of each run were unified. For the LS approach, we consider all the solutions (out of the 10 obtained), which are different. We also illustrate the evolution of the best merit function obtained by LS in all the 10 runs. Parameters used by LS and EA for all benchmarks are given in Table 2.

TABLE 2. Parameters used in experiments by LS and EA.

Parameter	Setting				
	Brown	Bullard	Ferrais	Himmelblau	Steady State CSTR
LS					
No of starting points	100	100	100	100	100
No of re-starts	10	10	10	10	10
No of iterations per re-start	5	5	5	5	5
EA					
Population size	500	100	100	500	200
Number of generations	500	500	500	500	200
Size of nondominated set	100	100	100	100	100
Sigma (for mutations)	0.1	0.1	0.1	0.1	0.1
Tournament size	3	3	3	3	3

3.1. Equation System 1 (Brown)

This benchmark is given by the following system of equations:

$$\begin{cases} 2\cdot x_1 + x_2 + x_3 + x_4 + x_5 - 6 = 0 \\ x_1 + 2\cdot x_2 + x_3 + x_4 + x_5 - 6 = 0 \\ x_1 + x_2 + 2\cdot x_3 + x_4 + x_5 - 6 = 0 \\ x_1 + x_2 + x_3 + 2\cdot x_4 + x_5 - 6 = 0 \\ x_1 \cdot x_2 \cdot x_3 \cdot x_4 \cdot x_5 - 1 = 0 \end{cases}$$

LS is only obtaining a single exact solution in all independent runs. This solution is (1, 1, 1, 1, 1) for which the functions values are all equal to 0. The evolution of the merit function is depicted in Figure 1. As evident, the merit function converges to 0 after 4 iterations (we consider 5 iterations in one re-start). This means there is not even a need to use derivatives and to restart the line search. The EA approach is obtaining multiple solutions but none of the solutions are close to the results obtained by LS. In Figure 2, the solutions obtained by EA in all the 10 runs are plotted. It can be observed that only few of them are having the Euclidian norm less than 1.

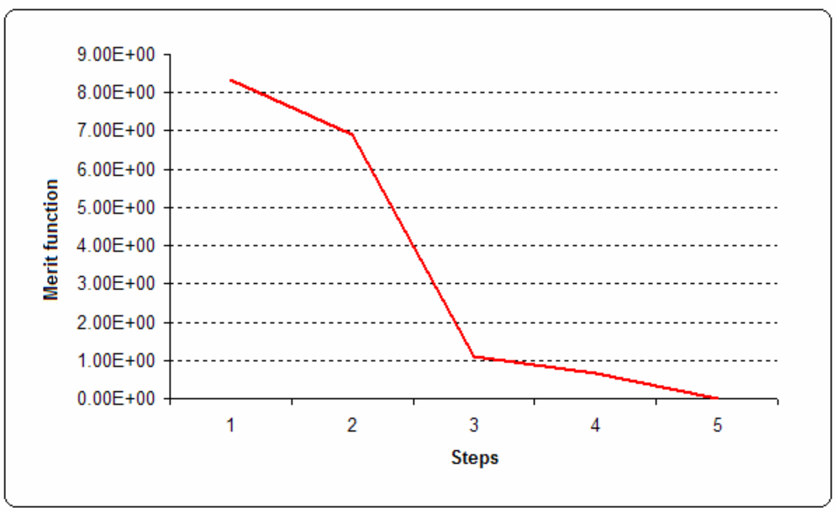

FIGURE 1. The evolution of the merit function for LS approach for Brown benchmark.

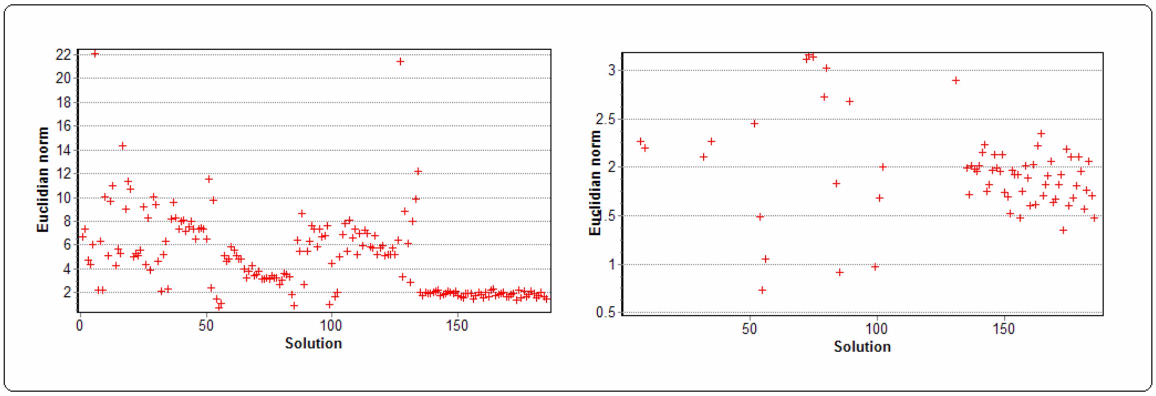

FIGURE 2. Solutions obtained by EA for Brown benchmark.

3.2. Equation System 2 (Bullard)

This benchmark consists of a system of two equations given by:

$$\begin{cases} 10000 \cdot x_1 \cdot x_2 - 1 = 0 \\ e^{-x_1} + e^{-x_2} - 1.001 = 0 \end{cases}$$

For Bullard benchmark, both algorithms obtain only approximate solutions. 10 best solutions obtained are depicted in Figure 3. Even though the evolution of the merit function (for LS) is very close to 0 (as evident from Figure 4) the values of the equations are in the range of 10^{-3}-10^{-5}.

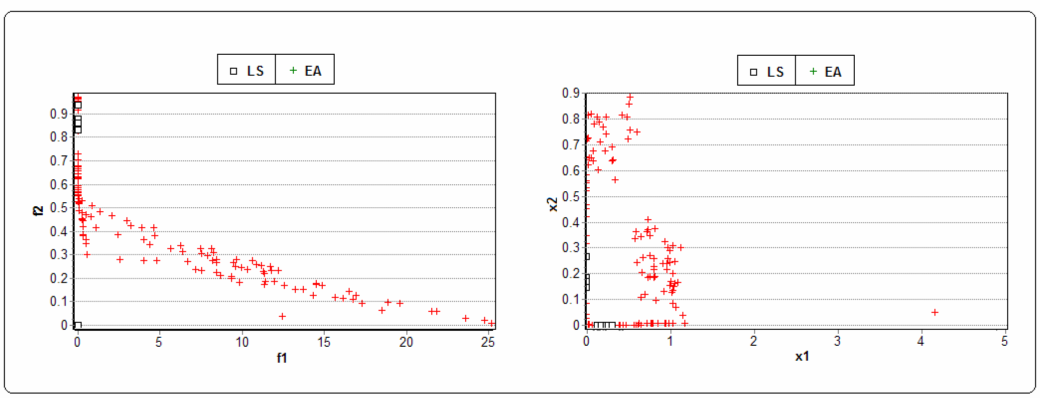

FIGURE 3. Solutions obtained by LS and EA for Bullard benchmark (objectives space-left, variables space-right).

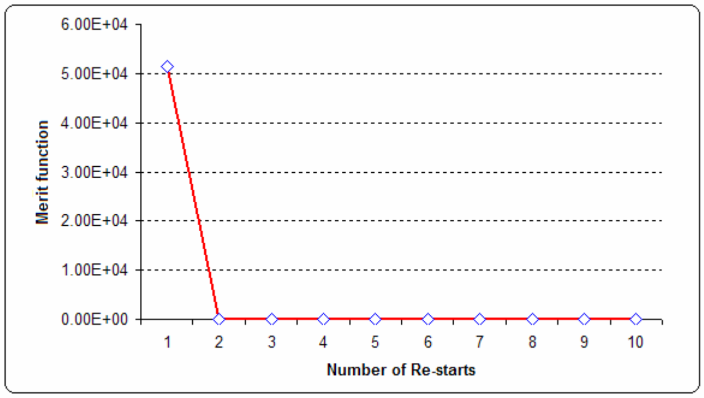

FIGURE 4 The evolution of merit functions for LS approach for the Bullard benchmark.

3.3. Equation System 3 (Ferrais)

This benchmark consists of the following system of equations:

$$\begin{cases} \dfrac{0.25}{\pi} x_2 + 0.5 \cdot x_1 - 0.5 \cdot \sin(x_1 \cdot x_2) \\ \dfrac{e}{\pi} \cdot x_2 - 2 \cdot e \cdot x_1 + \left(1 - \dfrac{0.25}{\pi}\right)\left(e^{2 \cdot x_1} - e\right) = 0 \end{cases}$$

For this example, LS obtained a single solution which is (0.5, 3.14) for which the functions values are 0.0001265 and 0.0137805. Solutions obtained by EA are plotted in Figure 5. The evolution of the merit function for LS is depicted in Figure 6.

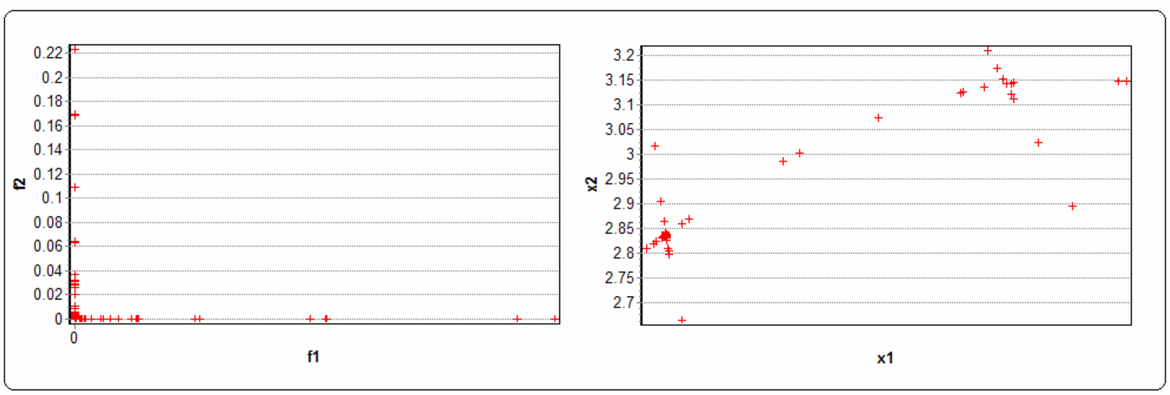

FIGURE 5. Solutions obtained by EA for Ferrais benchmark (objectives space – left, variables space-right).

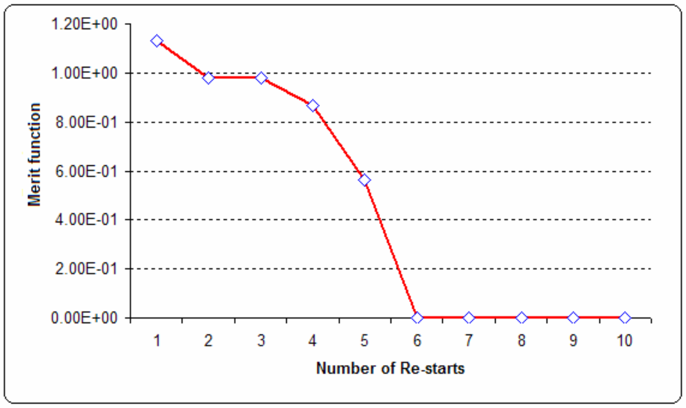

Figure 6. Evolution of merit function for LS for Ferrais benchmark.

3.4. Equation System 4 (Himmelblau)

This example is given by the following system of equations:

$$\begin{cases} 4x_1^3 + 4x_1x_2 + 2x_2^2 - 42x_1 - 14 = 0 \\ 4x_2^3 + 2x_1^2 + 4x_1x_2 - 26x_2 - 22 = 0 \end{cases}$$

For this example, there are 9 known solutions found so far. In 10 independent runs, LS is able to detect 7 solutions. The 7 solutions obtained by LS are given in Table 3. Solutions obtained by both LS and EA are plotted in Figure 7 (objectives space) and Figure 8 (variables space). As evident from Figure 8, in the variables space the solutions obtained by EA are centered on one of the solutions obtained by LS. The convergence of the merit function for the best result obtained in 10 runs (which is 0.2E-8) is depicted in Figure 9.

TABLE 3. Solutions obtained by LS for Himmelblau benchmark.

Solution		**Functions values**	
x_1	x_2	f_1	f_2
-0.270841381989373	-0.923036977925469	1.51325E-4	4.19365E-5
0.0867085036783106	2.88423339327931	1.19474E-3	1.212087E-3
-3.07304526798170	-0.0813371350117495	1.58640E-3	3.61678E-4
3.38519412590167	0.0735892503227077	-1.93149E-4	-3.80767E-3
3.00003916764214	1.99979209998351	1.25948E-3	6.28423E-3
3.58441811907790	-1.84770519568769	-1.85554E-3	-1.22804E-2
-0.127609787275348	-1.95313642658286	2.22599E-2	-8.22744E-3

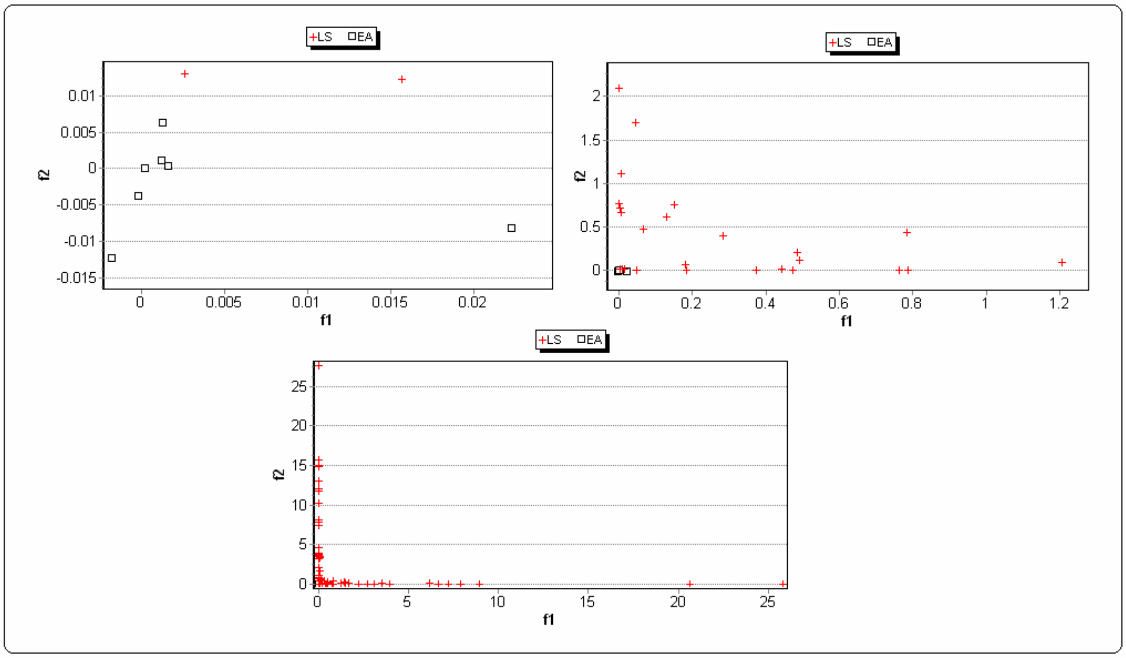

FIGURE 7. Solutions obtained by LS and EA for Himellblau benchmark represented in the objectives space (different sizes of the domain are considered for a better visualization).

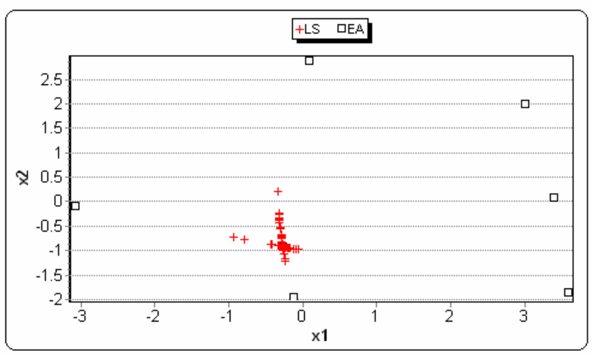

FIGURE 8. Solutions obtained by LS and EA for Himellblau benchmark represented in the variables space.

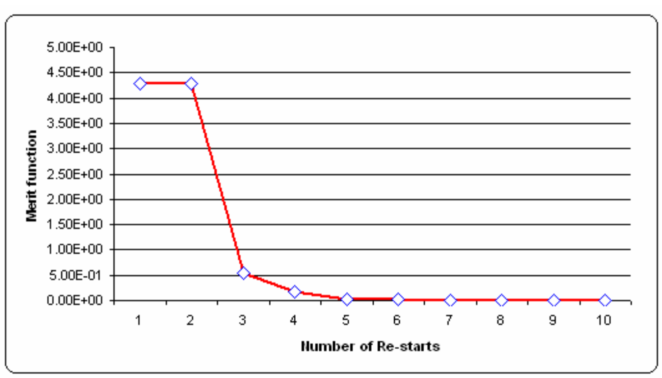

FIGURE 9. Evolution of merit function for LS for Himmelblau benchmark.

3.5. Equation System 5 (Steady state CSTR)

This problem is given by the following system of equations:

$$\begin{cases} x_1 - (1-R)\left(\dfrac{D}{10(1+\beta_1)} - x_1\right)\cdot \exp\left(10\cdot \dfrac{x_1}{1+\dfrac{10}{\gamma}\cdot x_1}\right) = 0 \\ x_1 - (1+\beta_2)x_2 + (1-R)\left(\dfrac{D}{10} - \beta_1\cdot x_1 - (1-\beta_2)\cdot x_2\right)\cdot \exp\left(10\cdot \dfrac{x_2}{1+\dfrac{10}{\gamma}\cdot x_2}\right) = 0 \end{cases}$$

where:

$D = 22$; $\beta_1 = 2$; $\beta_2 = 2$; $R = 0.935$; $\gamma = 1000$

but different other values for R may be considered. For this benchmark, LS obtained one solution (0.00752614, 0.05989059) for which the functions values are (-0.0433387, 0.09334198) and the merit function is 0.01059. The evolution of the merit function is depicted in Figure 10 and the solutions obtained by EA in all the 10 runs are depicted in Figure 11.

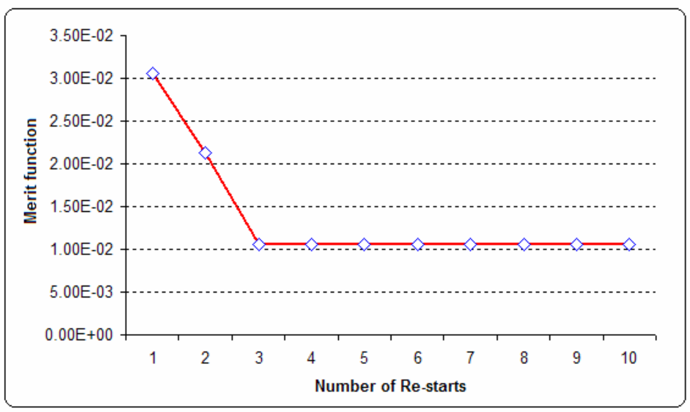

FIGURE 10. The evolution of merit function for LS for Steady state CSTR benchmark.

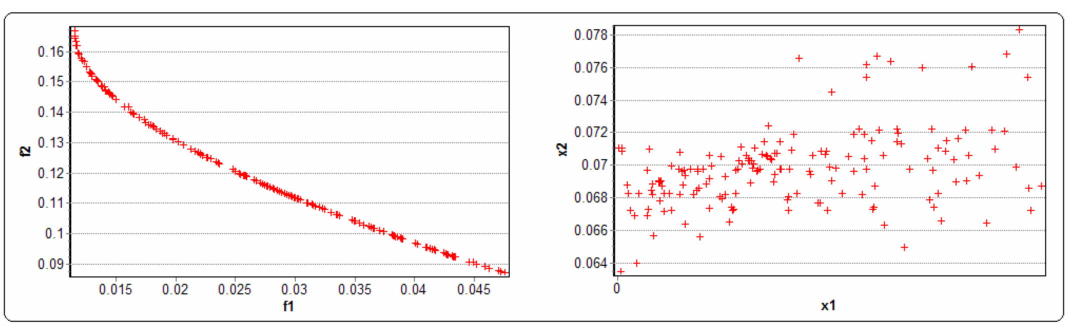

FIGURE 11. Solutions obtained by EA for steady state CSTR benchmark (objectives space – left, variables space-right).

5. Conclusions

One of the most studied problems in applied mathematics, engineering and sciences is funding multiple solutions of a set of nonlinear equations. Two different techniques are considered in this paper: line search based approach (LS) and an Evolutionary Algorithm (EA) based approach. Both techniques transform the system of equations into an optimization problem: LS transforms the systems of equations into a single objective optimization problem and EA transforms the system into a multiobjective optimization problem. Several equations systems having more than one solution are considered in the experiments. The numerical results reveal that LS can approximate solutions better than EA even though LS detect only one solution at one time and has to be applied multiple times while EA detects a set of solutions in one single run. Still the advantage of EA is that it can obtain multiple solutions and sometimes the user really needs a set from where the desired solution could be chosen.

REFERENCES

[1] M. Basin, J. Perez, D. Calderon-Alvarez, Optimal Filtering for Linear Systems over Polynomial Observations, International *Journal of Innovative Computing, Information and Control*, 4(2), pp. 313-320, 2008

[2] Michael Basin, Joel Perez, Rodolfo Martinez-Zuniga, Optimal Filtering for Nonlinear Polynomial Systems over Linear Observations with Delay, *International Journal of Innovative Computing, Information and Control*, 2(4) pp. 863-874, 2006

[3] C.G. Broyden, A class of methods for solving nonlinear simultaneous equations, *Mathematics of Computation*, 19, 577-593, 1965.

[4] A.R. Conn, N.I.M. Gould, P.L. Toint, *Trust-Region methods*, SIAM, Philadelphia, 2000.

[5] A. Cuyt, P.van der Cruyssen, Abstract Pade approximants for the solution of a system of nonlinear equations, *Computational Mathematics and Applications*, 9, 139-149, 1983.

[6] L.J. Eshelman, R.A. Caruna, J.D. Schaffer, Biases in the crossover landscape. In *Proceeding of the Third International Conference on Genetic Algorithms*, J. Schaffer (ed.), Morgan Kaufmann Publisher, Los Altos, CA, 10-19, 1989.

[7] A. Fekih, H. Xu, F. N. Chowdhury, Neural Networks Based System Identification Techniques for Model Based Fault Detection of Nonlinear Systems, International *Journal of Innovative Computing, Information and Control*, 3(5) pp. 1073-1085, 2007

[8] C. A. Floudas, P.M. Pardalos, C. Adjiman, W.R. Esposito, Z. H. Gümüs, S. T. Harding, J. L. Klepeis, C.A. Meyer, C.A. Schweiger, *Handbook of Test Problems in Local and Global Optimization (Nonconvex Optimization and Its Applications)*, Kluwer Academic Publishers, Dordrecht, 1999.

[9] D. E. Goldberg, *Genetic algorithms in search, optimization and machine learning*. Addison Wesley, Reading, MA, 1989.

[10] W. Gragg, G. Stewart, A stable variant of the secant method for solving nonlinear equations, *SIAM Journal of Numerical Analysis*, 13, 889-903, 1976

[11] Grosan, C., Abraham, A., Hybrid line search for multiobjective optimization, *International Conference on High Performance Computing and Communications (HPCC-07)*, Springer Verlag, Germany, R. Perrott et al. (Eds.): HPCC 2007, LNCS 4782, pp. 62-73, 2007.

[12]Grosan, C., Abraham, A., Exploration of Pareto Frontier Using a Fuzzy Controlled Hybrid Line Search, *7th International Conference on Hybrid Intelligent Systems*, Kaiserslautern, Germany, IEEE Computer Society press, USA, ISBN 07695-2662-4, pp. 366-371, 2007.

[13]C. Grosan, A. Abraham, A new approach for solving nonlinear equations systems, *IEEE Transactions on Systems, Man and Cybernetics - Part A*, Volume 38, No. 3, 2008 (in press).

[14]Grosan, C., Abraham, A., Chang, T.G., Kim, D.H. Solving Nonlinear Equation Systems Using an Evolutionary Multiobjective Optimization Approach, *5th Mexican International Conference on Artificial Intelligence*, Mexico, Lecture Notes in Computer Science, Springer Verlag, Germany, A. Gelbukh and C.A. Reyes-Garcia (Eds.): MICAI 2006, LNAI 4293, pp. 283-293, 2006.

[15]C. Grosan, A. Abraham, Exploration of Multiple Roots for a Polynomial System, *International Conference on Digital Information Management (ICDIM)*, Lyon, France, IEEE Press, USA, ISBN 1-4244-1476-8, pp. 133-137, 2007.

[16]M.J. Hirsch , C.N. Meneses , P.M. Pardalos,M.G.C. Resende, Global optimization by continuous grasp, *Optimization Letters*, 1(2), 201-212, 2007.

[17]R. Horst, P.M. Pardalos, N.V. Thoai, *Introduction to global optimization*, Kluwer Academic Publishers, 1995.

[18]R. Horst, P.M. Pardalos, *Handbook of global optimization*, Kluwer Academic Publishers, 1995.

[19]K.H. Ko, T. Sakkalis, N.M. Patrikalakis, Nonlinear polynomial systems: multiple roots and their multiplicities, Proceedings on Shape Modeling Applications, pp. 87-98, 2004

[20]C.D Maranas, C.A. Floudas, Finding all solutions of nonlinearly constrained systems of equations, *Journal of Global Optimization*, 7, 143-182, 1995.

[21]P.Y. Nie, A null space method for solving system of equations, *Applied Mathematics and Computation,* 149(1), 215-226, 2004.

[22]P.Y. Nie, An SQP approach with line search for a system of nonlinear equations, *Mathematical and Computer Modelling*, 43, 368-373, 2006.

[23]J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*. New York: Academic Press, 1970

[24]W.M. Spears, K.A. De Jong, On the virtues of uniform crossover. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, Morgan Kaufmann Publisher, 230-236, 1991.

[25]G. Syswerda, Uniform crossover in genetic algorithms. In *Proceedings of the third Conference in Genetic Algorithms*, J. Schaffer (ed.), Morgan Kaufmann Publisher, Los Altos, CA, 2-9, 1989.

[26]Steuer, R. E. Multiple Criteria Optimization. Theory*, Computation, and Application*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. New York: John Wiley Sons, Inc, 1986.

[27]S. Tong, Y. Li, Direct Adaptive Fuzzy Backstepping Control for a Class of Nonlinear Systems, *International Journal of Innovative Computing, Information and Control*, 3(4) pp. 887-896, 2007