Journal of Applied Sciences

ISSN 1812-5654

Journal of Applied Sciences 9 (10): 1889-1897, 2009
ISSN 1812-5654

Intelligent System for Continuous Gas Lift Operation and Design with Unlimited Gas Supply

[1]E. Khamehchi, [1]F. Rashidi, [2]H. Omranpour, [2]S. Shiry Ghidary, [3]A. Ebrahimian and [1]H. Rasouli
[1]Faculty of Chemical Engineering, Amirkabir University of Technology, Tehran, Iran
[2]Faculty of Computer Engineering and Information Technology, Amirkabir University of Technology, Tehran, Iran
[3]Faculty of Petroleum Engineering, Amirkabir University of Technology, Tehran, Iran

Abstract: Gas lift is one of a number of processes used to artificially lift oil or water from wells where there is insufficient reservoir pressure to produce the well. The process involves injecting gas through the tubing-casing annulus. Injected gas aerates the fluid to reduce its density; the formation pressure is then able to lift the oil column and forces the fluid out of the wellbore. Gas may be injected continuously or intermittently, depending on the producing characteristics of the well and the arrangement of the gas-lift equipment. To enhance the financial revenues this operation has usually always been a subject for optimization to reach the most rewarding design before its operational establishment. Evolutionary approaches have recently been successfully applied to almost every aspect of engineering problems. This study reviews the general facts and ideas related to the gas lift and its optimization and further focus on the application and evaluation of genetic programming for such a purpose. It has been concluded that genetic programming is fully capable in aiding faster gas lift optimizations while is also stable and applicable to a very broad range of operating conditions. The merits and draw backs are finally compared with the neural network approach.

Key words: Genetic programming, mutation, cross over, gas lift, optimization, depth of injection

INTRODUCTION

Artificial lift is used in oil production when the energy of the reservoir is not enough to sustain the flow of oil in the well up to the surface with a satisfactory economic return. Selection of the proper artificial lift method is critical to the long-term profitability of an oil well; a poor choice will lead to low production and high operating costs. There is a very little margin for error when one is designing lift systems for oil fields. There is a strong need for reliable An extensive overview of artificial lift design considerations is presented in (Clegg *et al*., 1993). Rod pumps, electric submersible pumps and gas lift schemes are the most common artificial lift systems, but plunger lift, hydraulic and progressing cavity pumps are also used. Gas lift is a widely used method among artificial lift methods, in which gas is injected in the production well providing energy to the flow. Continuous gas lift being cost-effective, easy to implement, very effective in a wide range of operating conditions and requiring less maintenance in comparison to other alternatives, is one of the most typical forms of artificial lift in oil production. It is a usual one where there is an abundance of natural gas resources (Taheri and Hooshmandkoochi, 2006).The basic principle consists of decreasing the pressure gradient in the liquid via the injected gas (Fig. 1). The resulting mixture becomes less heavy than the original oil so that it eventually starts flowing (Pafalox-Hern, 2005) and (Poblan, 2002) can be named for further understanding of the mechanisms (Fig. 2) (http://www.weatherford.com). In gas lift operations, two problems are the most important ones. The first one is finding the optimal position for injection point and the other is estimating the optimal gas injection rate (Santos *et al*., 2001). These parameters are interrelated, the more the rate of gas injection the deeper would be the optimum injection point. In other words, the deeper the injection location the more gas volumes would be needed for maximum production. In the present study, the data of 40 wells that are under gas lift operation were used. These wells were selected from one of the huge oil reservoirs in Southwestern Iran.

This reservoir has more than 150 wells a portion of which are producing under gas lift system. For each of these wells the following set of data were gathered for

Corresponding Author: Fariborz Rashidi, Faculty of Chemical Engineering, Amirkabir University of Technology, P.O. Box 15914, Tehran, Iran Tel: +98 21 64543190

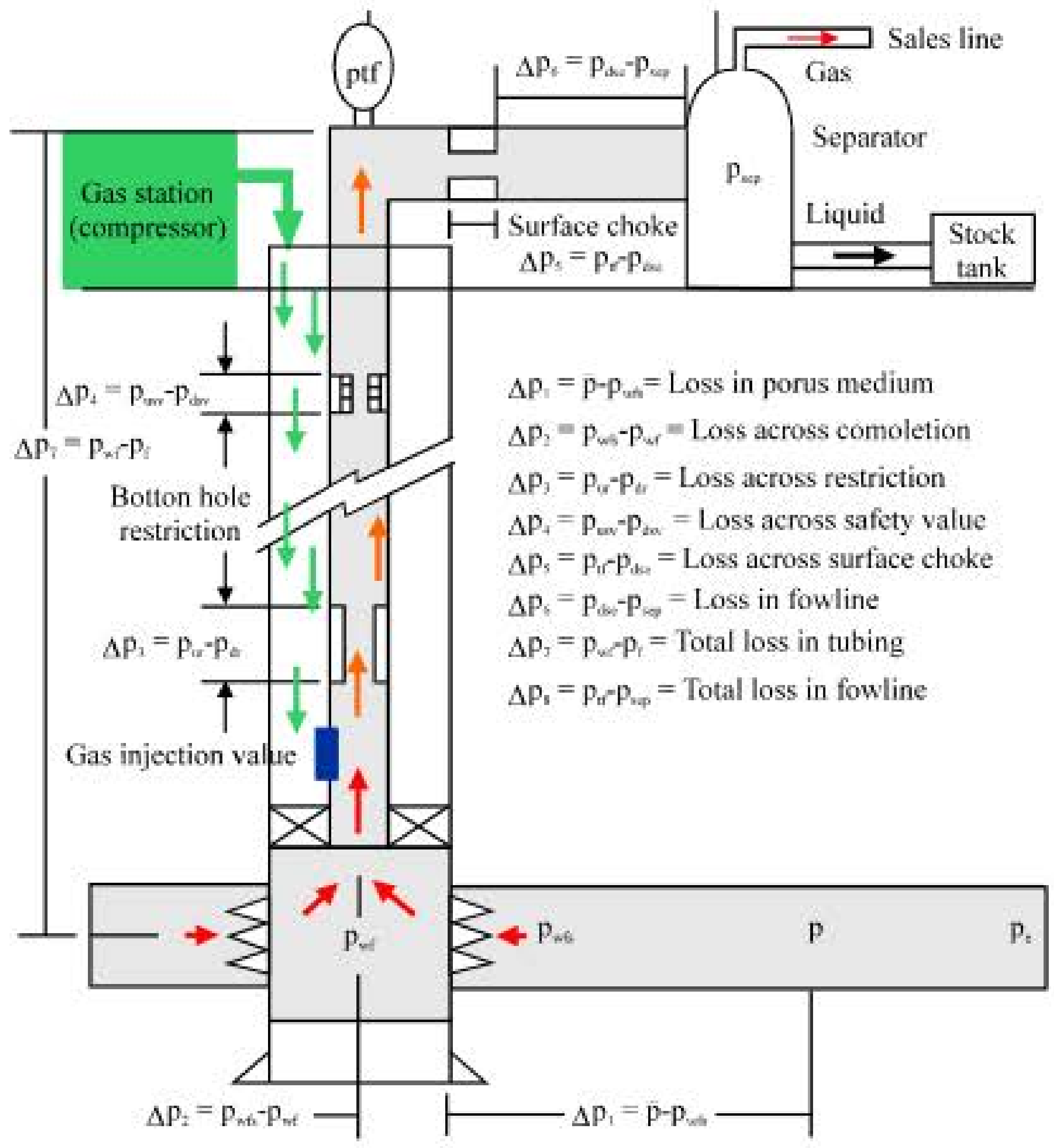

Fig. 1: Schematic of a gas lift well

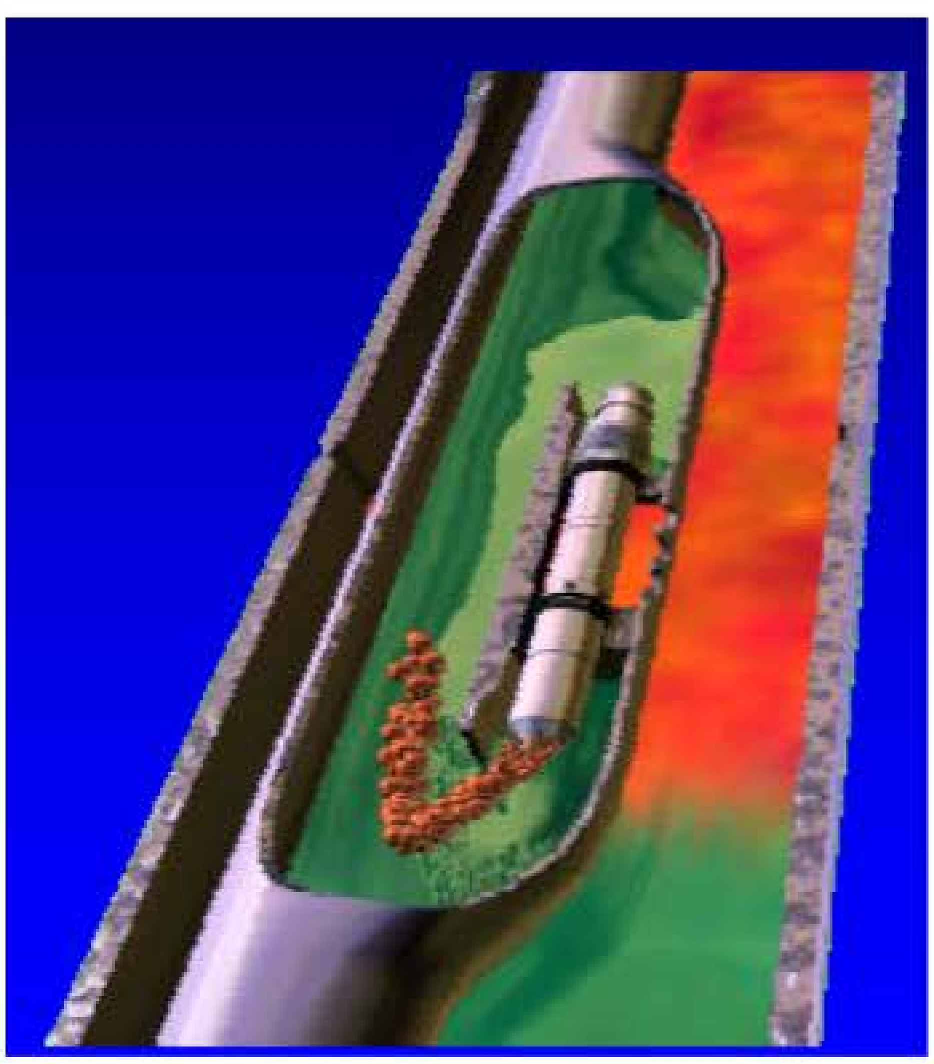

Fig. 2: Enlarged view of a typical gas injection valve

subsequent use in the study: Bottom Hole Static Pressure (BHSP), Well Head Flowing Pressure (WHFP), Bottom Hole Flowing Pressure (BHFP), Productivity Index (PI), Well-bore size, tubing size, water-cut percent (WC%), oil production rate (QOil), gas injection rate (QGasInj) and depth of injection. The range of the data is given in Table 1. The last two variables are considered as output data and the others as input data.

Table 1: Range of training data

Parameters	Min	Max
BHSP (psi)	44300	5933
WHFP (psi)	411	670
BHFP (psi)	4157	5300
Pi	0.43	3.56
Well-bore size (in)	5"	6.1/8"
Tubing size (in)	3.12"	7"
WC%	0	2
Q_{oil} (STB/day)	200	3200
Q_{gasInj} (MMSCFD)	0.9	3.2
Injection depth (ft)	8090	10988
Well depth (ft drilled depth)	11508	13650

Though, Gas Oil Ratio (GOR) is also a role-playing parameter but since the considered reservoir is an under-saturated, spatially well-communicating one, so, GOR is the same for all wells and therefore is not taken as a variable here.

The two aforementioned crucial factors of the design are conventionally usually estimated through the use of multiphase flow simulation packages available in the market and subsequently they are implemented in a field scale plan. Evolutionary Algorithms has proved to be helpful and interesting for petroleum engineering purposes. For Gas allocation problems in cases where there is a limited source of gas these algorithms has shown promising results (Ray and Sarker, 2008). In the present study, the Genetic programming approach has been applied to gas lift optimization, through predicting the two most important parameters of the design as described above.

It should be mentioned that due to the industrial scale of the events, data availabilities are usually choked by confidentiality limitations.

OPTIMIZING GAS LIFT FOR INDIVIDUAL WELLS

Gas lift is a costly, however indispensable means to recover oil from high-depth reservoirs that entails solving the gas-lift optimization problem (GOP) often in response to variations in the dynamics of the reservoir and economic oscillations (Camponogara *et al.*, 2005).

As the relative oil and gas superficial velocities in a pipe vary the flow regime in the pipe changes according to some empirical vertical flow pattern maps. For example Duns and Ros (1963) and Kaya *et al.* (1999) usually try to avoid entering the slug regime area. If you take a well under tubing head pressure control and gradually increase the supply of lift gas to it, the production rate at first increases due to the reduced density of the mixture in the tubing. But as the lift gas supply is increased further, friction pressure losses in the tubing become more important and the production rate starts decline (Fig. 3). For an individual well with no constraints other than a tubing head pressure limit, with an unlimited free supply of lift gas, the optimum lift gas injection rate is the value at the peak (point A).

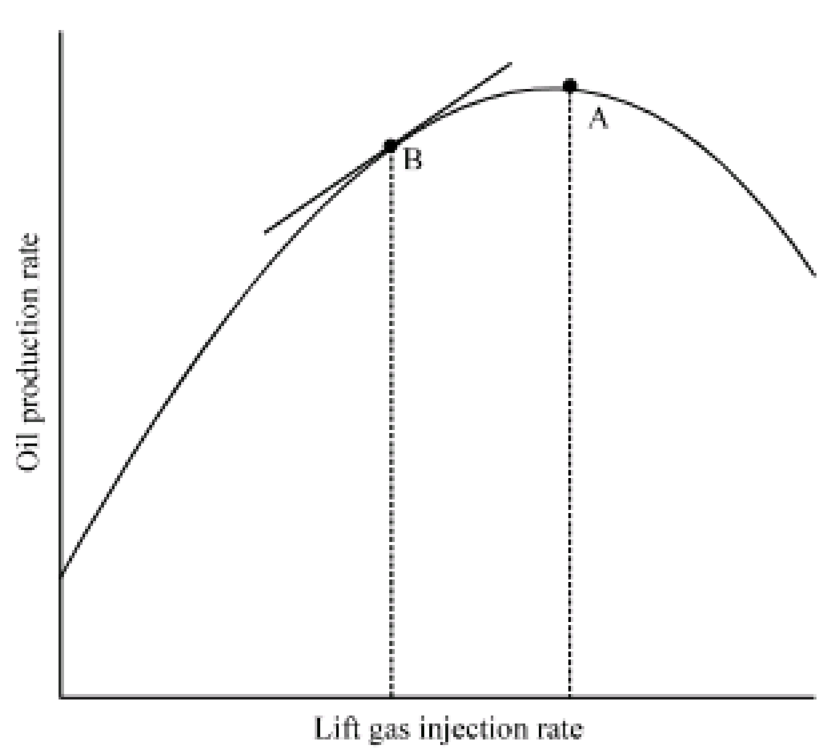

Fig. 3: Economic point and optimum point in gas lift performance curve (Eclipse Technical Description v2004 A)

In reality though, lift gas is never free. Compression costs can be expressed as a cost per unit rate of lift gas injection (for example, dollars/day per MMscf/day). This must be balanced against the value of the extra amount of oil produced. Thus there is a minimum economic gradient of oil production rate versus lift gas injection rate, at which the value of the extra amount of oil produced by a small increase in the lift gas injection rate is equal to the cost of supplying the extra amount of lift gas. The optimum lift gas injection rate is then somewhat lower than the peak value, at the point on the curve where its gradient equals the minimum economic gradient (point B). However in this study we assumed that gas supply is unlimited and free-source and therefore we have tried to obtain point A.

In reality though, lift gas is never free. Compression costs can be expressed as a cost per unit rate of lift gas injection (for example, dollars/day per MMscf/day). This must be balanced against the value of the extra amount of oil produced. Thus there is a minimum economic gradient of oil production rate versus lift gas injection rate, at which the value of the extra amount of oil produced by a small increase in the lift gas injection rate is equal to the cost of supplying the extra amount of lift gas. The optimum lift gas injection rate is then somewhat lower than the peak value, at the point on the curve where its gradient equals the minimum economic gradient (point B). However, in this study we assumed that gas supply is unlimited and free-source and therefore we have tried to obtain point A.

Having influential parameters quantitatively available from a real field we optimized the injection position and gas injection rate in the individual wells using the Wellflo software. The obtained results were in good agreement with the real world designs already implemented in the field. This way we were assured of having our available field data as a reliable set of reference optimum criteria. We then tried adopting the best genetic structures, their output results were then compared with the outputs of the available field data and the Wellflo outputs. This was done through the use of the data from 36 wells for the training step and testing the networks with the rest 4 remaining wells.

Conventional methodology: Nodal analysis: The system analysis approach, often called NODAL analysis has been applied for many years to the systems composed of interacting components. Its application to well producing systems was first proposed by Gilbert in 1954 and was discussed by Nind in 1964 and Brown in 1978 (Mach *et al.*, 1979). NODAL analysis requires first selecting a node and calculating the node pressure, starting at the fixed or constant pressure existing in the system. These fixed pressures are usually mean Reservoir Pressure (PR) as the inlet pressure and either wellhead pressure (Pwh) or separator pressure (Psep) as the outlet pressure. The node may be selected at any point in the system (Fig. 1).

Components begin with the static reservoir pressure, ending with the separator and including inflow performance, as well as flow across the completion, up the tubing string (including any down hole restrictions and safety valves), across the surface choke (if applicable), through horizontal flow lines and into the separation facilities are analyzed (Brown, 1985). The expressions for the flow into the node and for the flow out of the node can be expressed as:

$$P_{node} = P_{inlet} - \Delta P \text{ (upstream components)}$$
$$= P_{outlet} + \Delta P \text{ (downstream components)} \quad (1)$$

The two criteria that must be met are: 1- Flow into the node equals flow out of the node and 2-Only one pressure can exist at the node for a given flow rate. The performance of a gas lift well can also be treated similar to a flowing well with the only difference that the tubing string is divided into two sections with the dividing point placed at the depth of the gas injection. The section below the gas injection point contains the gas produced from the formation only, whereas the one above the injection point contains the injected gas volume as well.

Simulation tools: WellFlo: WellFlo software is a powerful, stand-alone application for designing, modeling, optimizing and troubleshooting individual oil and gas wells, whether naturally flowing or artificially lifted. With this software, the engineer builds well models, using a guided step-by-step well configuration interface. These accurate and rigorous models display the behavior of reservoir inflow, well tubing and surface pipeline flow, for any reservoir fluid. Using WellFlo software results in more effective capital expenditure by enhancing the design of wells and completions, reduces operating expenditure by finding and curing production problems and enhances revenues by improving well performance. The WellFlo software package is a single well tool which uses Nodal analysis techniques to model reservoir inflow and well outflow performance. By using the program's specialized capabilities for gas lift, engineers can design and model gas lift installations and determine the number and position of the gas lift valves, as well as the optimum injection rate by taking into account the available injection pressure. This program will allow you to incorporate gas injection rate or gas-liquid ratio terms as preferred (www.ep-solutions.com).

GENETIC PROGRAMMING

One of the hot topics in computer science is finding ways to have a computer do a specific job without describing for it how to do that job. Genetic programming has proved to be an approach to this issue. Genetic programming was introduced in early 1990s and has gradually been developed mostly by its innovator John Koza (Andries and Engelbrecht, 2007). This method produces a program for a high level problem. Genetic programming develops a population of computer programs that are modeled as graphs and are grown up based on the Darwin's evolution theory (Genetic Algorithm). Development of this population is derived from selection and reproduction operations of biological systems. These operations include mutation and cross over phenomena combined with parent and survivors selection procedures. As genetic programming models as graphs and trees it can be used for function estimation purposes (Koza, 1992).

First Steps for Genetic Programming are the following:

- We determine a set of terminals. For converting a function into chromosome forms needed for genetic algorithm that is tree, we need to determine leaf elements. These elements are also called 'Terminals'. Terminals can be independent variables of the problem, functions without input parameters or constants that are generated stochastically
- A set of primary functions based on which the tree should be constructed
- Fitness criteria, for the purpose of evaluating and recognition of best trees produced by genetic algorithm
- Parameters that are need for controlling the genetic algorithm execution. For example the way of parent selection and generations, mutation rate and rate of cross over
- Stopping condition
- The flow-diagram deployed for this study is depicted in Fig. 4

Two issues should be noted; firstly the closure condition should be fulfilled. This means that each function that takes a value should be able to manage all output results. A typical example is avoiding dividing by zero.

Secondly, the problem we are to program must be soluble by a combination of terminals and functions we consider. For example if we want to write a program that calculates logarithm of a number, so if we limit ourselves

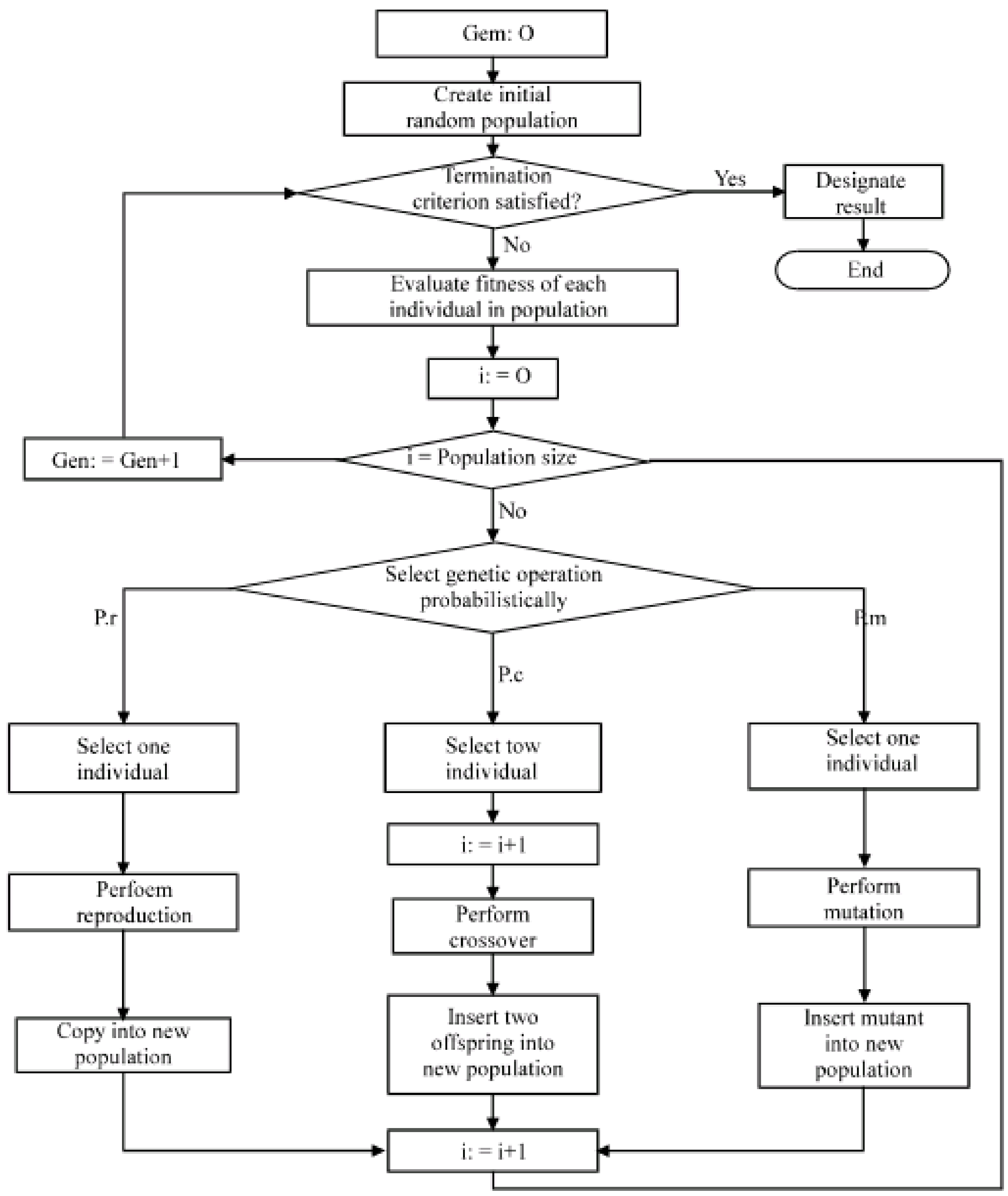

Fig. 4: Genetic programming flowchart

by algebraic operators and cardinal terminals even in the best situations we are not capable of calculating the logarithm with an acceptable approximation (Reily *et al*., 2005).

Representation: In this state we construct a binary tree for each function (Fig. 5).

Internal nodes are constructed by the following set, {*, -, +, /, ^, Sin, Cos, Exp, tgh} and the leaves are constructed using the following set, {Real Numbers + Properties}.

For any vector of data set like x, the tree i estimates its output according to its internal nodes. So, it's obvious that we would have as many trees as the number of functions; this means that any chromosome is composed of some several trees (Fig. 6) which are two for this specific case. But if each tree evolutes in a separate population we would reach better individual outputs and

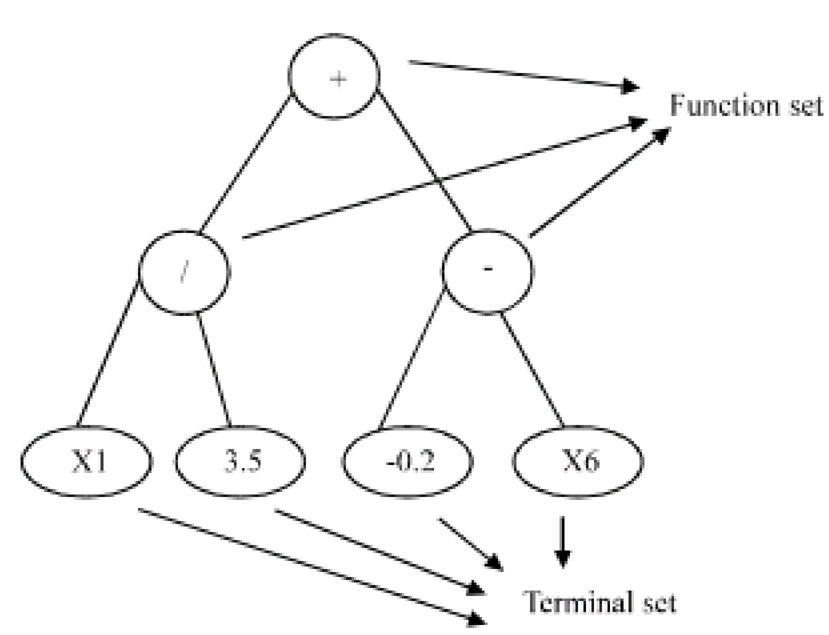

Fig. 5: A tree type chromosome

it was observed that if the poor elements of one population are transferred into another one this can yield better mutual performance.

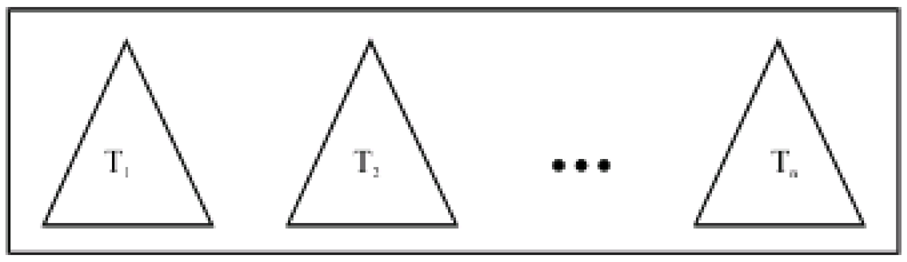

Fig. 6: A Typical chromosome in genetic programming

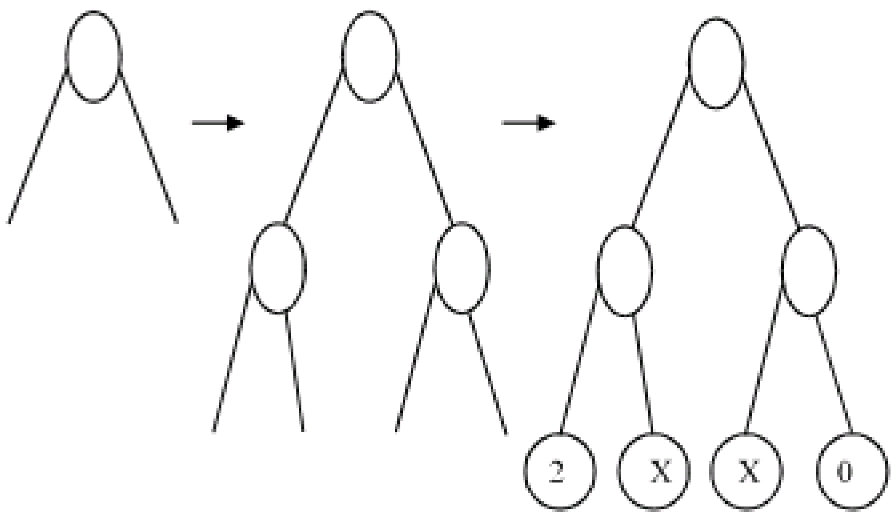

Fig. 7: How to generate dense trees

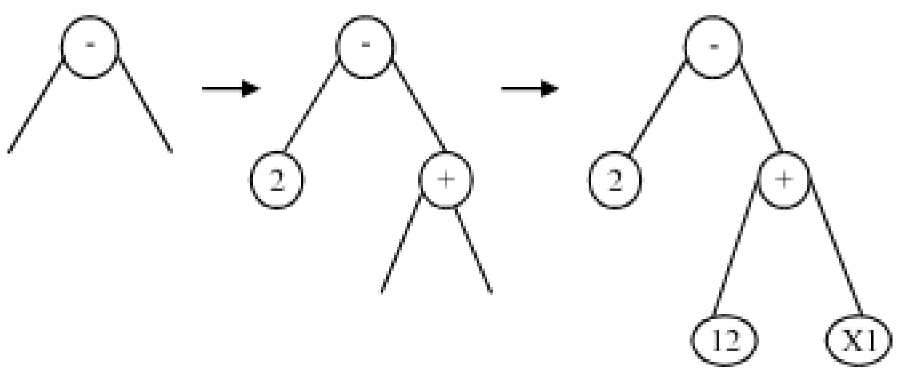

Fig. 8: How to generate non-homogeneous trees

Based on a procedure like Fig. 4, we seek optimized trees and chromosomes that are capable of acceptable and good predictions.

Initialization: It should be noted that the depth of new born trees has to be controlled. There are two strategies for doing so, the first method is that all the initial trees have the same depth and be dense. Figure 7 shows how such trees are generated. The second method is to generate trees without any limitations other than the maximum depth control. An example of such a procedure is given in Fig. 8.

We adopted and suggest a combination of the both above to ensure enough versatility.

Fitness function: A fitness function is adopted for the evaluation purposes. For this case the least square is used (Eq. 2).

$$\text{Fitness} = \sum_{i=1}^{n} (\text{Evaluate}(x_i) - y_i)^2 \tag{2}$$

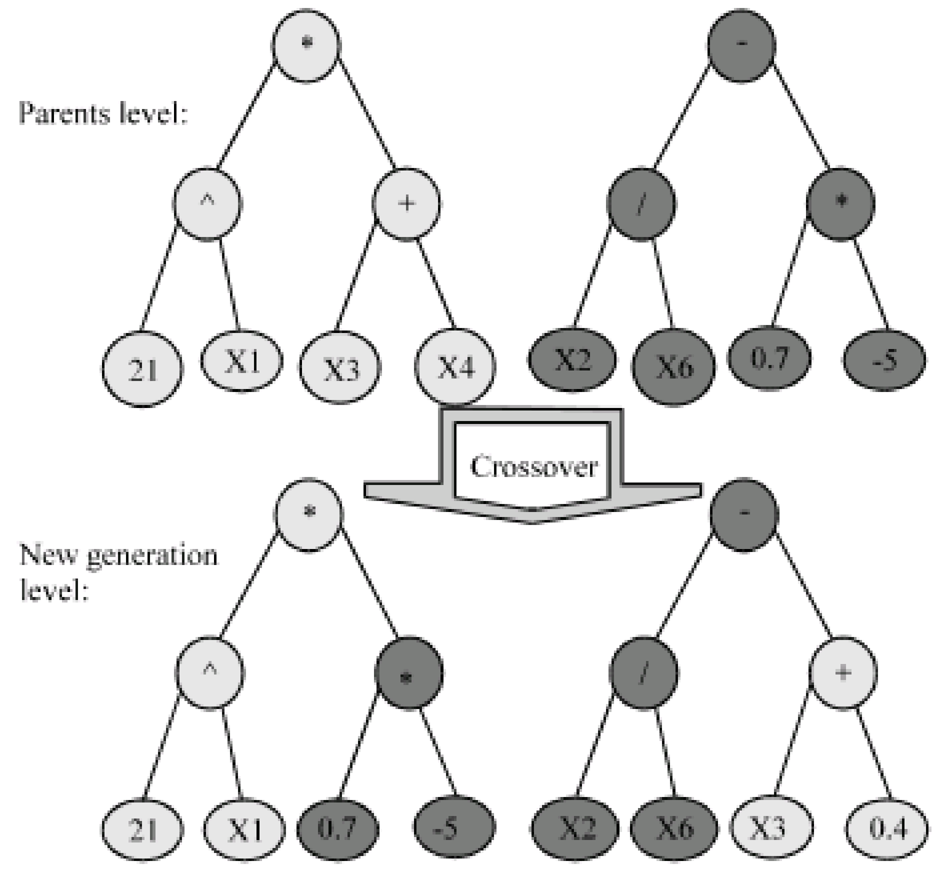

Fig. 9: Cross over

Selection: For selecting among the generated population a tournament method is usually adopted. This has proved better performances than the Rolet wheel and the SUS methods. This is simply executed by stochastic selection of Q initial elements and a final element among them based on the best fitness. This procedure is repeated several times to reach enough number of selections. This method ensures a global search due to its diversity in selection number of selections. This method ensures a global search due to its diversity in selection (Miller and Goldberg, 1995).

Cross over: Cross over is an operation performed on some several chromosomes and produces a new generation. Each child takes a combination of characteristics from both its parents. This is simulated in programming by substituting stochastically selected elements (sub-trees or terminals) taken from the parent trees (Fig. 9).

Mutation: Mutation is an evolution fundamental which causes new creatures in a novel search space. Any function or terminal in the tree can be replaced by any other arbitrary function or terminal. Mutation changes are of four types:

- Removal of a sub-tree (Fig. 10)
- Adding a sub-tree (Fig. 11)
- Changing the function of an internal node (Fig. 12)
- Changing the value or characteristic of a leaf (Fig. 13)

The fourth type mutation can help a lot toward a better fitness. Here we tried changing the constant

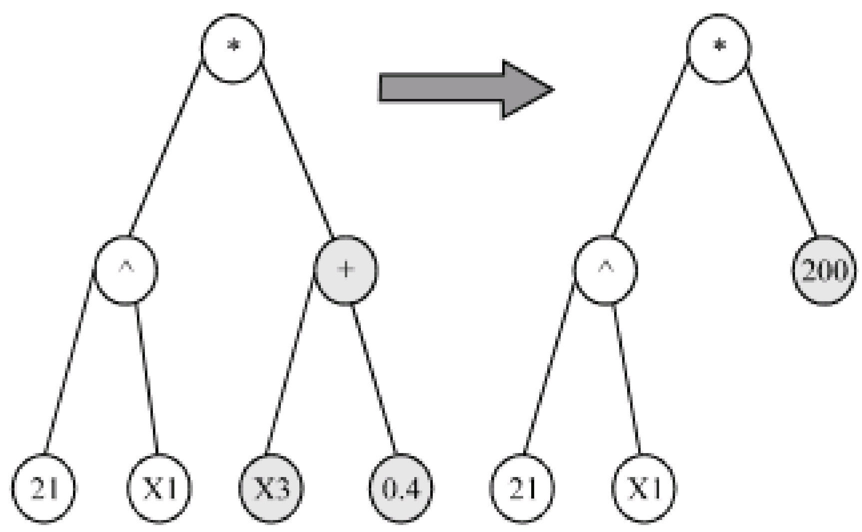

Fig. 10: First type mutation

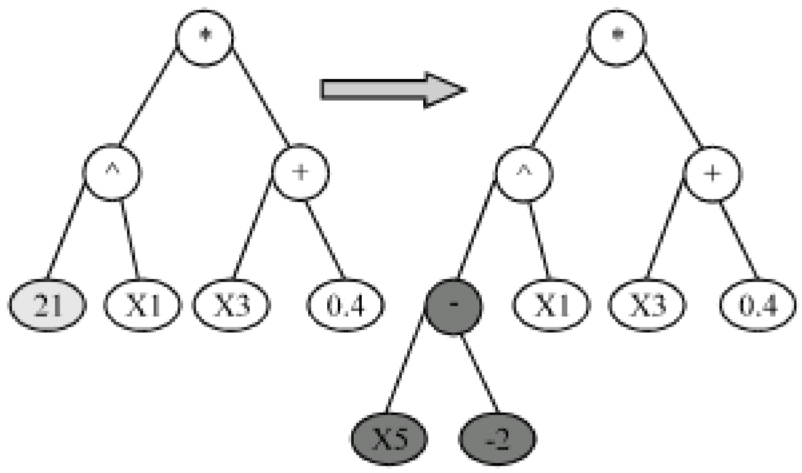

Fig. 11: Second type mutation

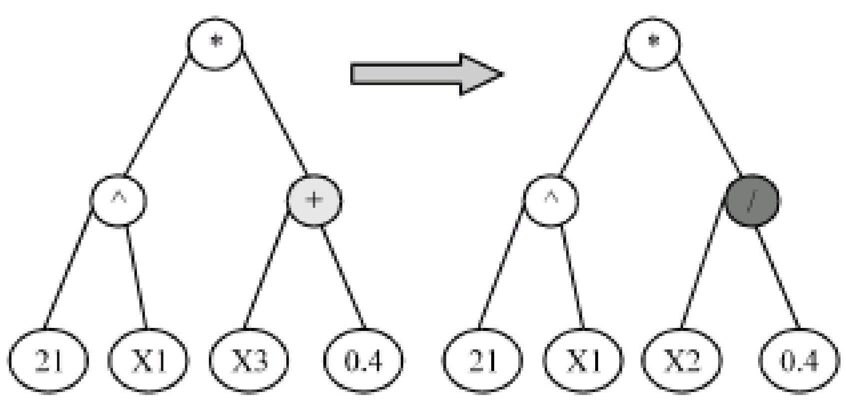

Fig. 12: Third type mutation

terminals by a stochastic positive and negative ε number. This number is larger in the first generations and approaches zero in final generations. This ensures global searches and also local refinements. Also, the probability of this type of mutation changes ascending by the generation number to allow other types of mutation to come into help for better global searches.

Finalization criteria: Selection of finalization criteria may be more difficult than it appears since we do not know weather the algorithm has reached the optimum answer or not. A specific average fitness, a specific generation number or that the fitness variation is no more improvable can define the conditions when we can abandon the process. The maximum generation number yielded the result for this specific study.

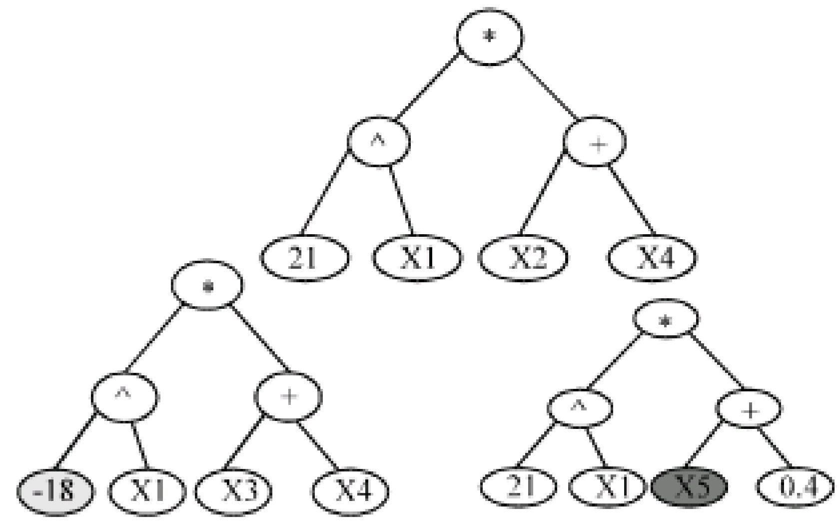

Fig. 13: Fourth type mutation

A control on the depth of the tree should exist as uncontrolled growth of the depth causes the following:

- It takes considerable amount of memory out of the system
- The convergence time increases as the speed of genetic operations decreases

For such a purpose the following two approaches are in use:

- Defining a maximum depth for the trees can be simply programmed by harvesting branches exceeding the maximum allowed number of levels. If the answer is not obtained in this method the allowable limit may be increased
- Penalizing long trees can be done to lower the chance of survive for them. This method has shown better performances than the former

SPECIFIC EVALUATION OF PARAMETERS

For a genetic programming to succeed, a proper set of operators, functions and terminals must be applied with a proper combination of reproduction, cross over and mutation probabilities that not only guarantees the convergence but also expedite it. In this study, a mutation probability of 0.2, Cross over probability of 0.7 and reproduction probability of 0.1 is adopted and could yield satisfactory results. The maximum depth for the trees were taken 10 levels and with a penalizing strategy the chance of survive for those deeper than 20 levels was reduced to zero. The number of generations was 5000 and the population number also was 100.

A COMPARISON WITH THE NEURAL NETWORK APPROACH

The following could be concluded by comparison with a research based on a neural network optimization approach (Khamehchi *et al*., 2008):

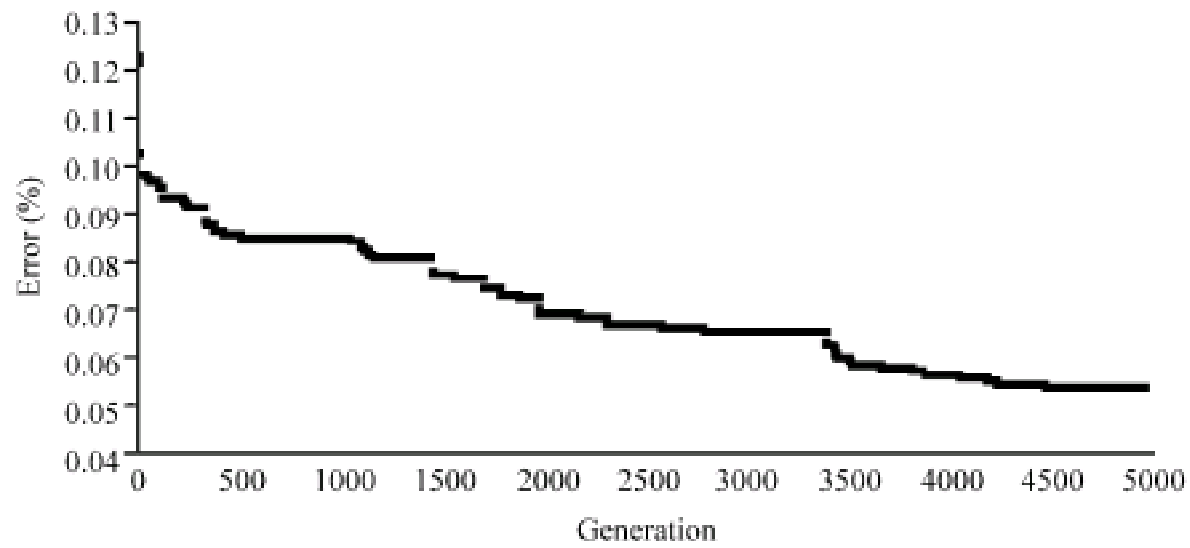

Fig. 14: Error vs. generation Number

- In the genetic programming there is the possibility to use variety of functions and operators combinations but in neural network you have to use a single mathematical function or operator and therefore the coverage of operators in genetic programming is broader
- In multilayer neural networks the number of layers and nodes are initially adopted and remains constant through out the calculations this is while in Genetic Programming the depth of trees can vary and where the search space is voluminous the Genetic Programming yields faster convergences while for smaller search spaces ANN is more suitable
- Artificial neural networks give have better outputs for the training data but for the testing data genetic programming is more desirable as it utilizes variety of functions and operators dynamically
- As in the neural network method a fixed network is continually updated and the error back propagation is used this method gives lower average errors

RESULTS

With the parameter values given in section 4 a final error of 5.3% was reached at the generation number 5000. The trend of changes in error is depicted in Fig. 14.

CONCLUSION

It can be concluded that genetic programming is fully capable in aiding faster gas lift optimizations purposes while is also stable and applicable to a very broad range of operating conditions. There can be observed some merits and also draw backs compared to the artificial neural networks approach.

ACKNOWLEDGMENTS

Researchers would like to thank National Iranian Oil Company-Petroleum Engineering and Development Company (NIOC-PEDEC) for its support of this research.

REFERENCES

Andries, P. and Engelbrecht, 2007. Computational Intelligence. 2nd Edn., John Wiley and Sons Ltd., USA, ISBN: 978-0-470-03561-0.

Brown, K.E., 1985. System analysis of oil and gas wells. J. Petroleum Technol., 101: 1751-1763.

Camponogara, E., 2005. Solving a gas-lift optimization problem by dynamic programming. Eur. J. Operat. Res., 174: 1220-1246.

Clegg, J.D., S.M. Bucaram and J. Hein, 1993. Recommendations and comparison for selecting artificial lift methods. J. Petroleum Technol., 45: 1128-1167.

Duns, H.J. and J.C.J. Ros, 1963. Vertical Flow of Gas and Liquid Mixtures in Wells. Proceeding of the 6th World Petroleum Congress, 1963, Frankfurt, pp: 451-465.

Kaya, A.S., C. Sarica and J.P. Brill, 1999. Comprehensive Mechanistic Modeling of Two-Phase Flow in Deviated Wells, SPE 56522. Proceeding of the SPE Annual Technical Conference and Exhibition held in Huston, TX, Oct. 3-6, Society of Petroleum Engineers, pp: 11.

Khamehchi, E., F. Rashidi, H. Rasouli, Rasouli and A. Ebrahimian, 2008. A new approach to continuous gas lift optimization using artificial neural networks. J. Eng. Comput.

Koza, J., 1992. Genetic Programming. Massachusetts Institute of Technology, New York.

Mach, J., E. Pmano and K.E. Brown, 1979. A Nodal Approach for Applying Systems Analysis to the Following And Artificial Lift Oil or Gas Well. Society of Petroleum Engineers, Richardson, TX.

Miller, B.L. and D.E. Goldberg, 1995. Genetic Algorithms, Tournament Selection and the Effects of Noise. Complex Systems, USA.

Pafalox-Hern, P., 2005. A new approach for continuous gas-lift simulation and optimization. Proceeding of the Presented at SPE Annual Technical Conference and Exhibition, Oct. 9-12, Dallas, Texas, pp: 1.

Poblan, E., 2002. Stability analysis of continuous-flow gas lift wells. Proceeding of the SPE Annual Technical Conference and Exhibition, Sept. 2- 29, San Antonio, Texas, pp: 12-12.

Ray, T. and R. Sarker, 2008. Evolutionary Algorithms deliver promising results to gas lift optimization problems. World Oil, 229: 141-142.

Reily, U., T. Yu, R. Riolo and B. Worzel, 2005. Genetic Programming Theory and Practice II. Springer Science Business Media, USA, ISBN: 978-0-387-23253-9.

Santos, G., N. Bordalo and F.J.S. Alhanati, 2001. Study of the dynamics, optimization and selection of intermittent gas-lift methods-a comprehensive model. J. Pet. Sci. Eng., 32: 231-248.

Taheri, A. and A. Hooshmandkoochi, 2006. Optimum selection of artificial-lift system for Iranian heavy-oil fields. Proceeding of the SPE Western Regional/AAPG Pacific Section/GSA Cordilleran Section Joint Meeting, May 8-10, Anchorage, Alaska, USA, pp: 12-12.