

## Online Optimization of Industrial FCC Unit Based on PSO Algorithm and RBF Neural Network

Yi Deng

Department of Chemical & Biochemical Engineering,  
College of Chemistry & Chemical Engineering  
Xiamen University  
Xiamen, Fujian Province, China  
denincool@sina.com

\* Qingyin Jiang and Zhikai Cao

Department of Chemical & Biochemical Engineering,  
College of Chemistry & Chemical Engineering  
Xiamen University  
Xiamen, Fujian Province, China  
\* xdceds@xmu.edu.cn

**Abstract - The Particle Swarm Optimization (PSO) and the Genetic Algorithm (GA) are two of the most powerful methods to solve the unconstrained and constrained global optimization problems. In this paper, these two methods are briefly introduced firstly, and then the online rolling optimization of industrial FCC unit is carried out based on the RBF Neural Network predictive model. The results of simulation based on the two optimization methods are compared. The comparative results show that the PSO can perform well as the GA in searching the global optimal position. Furthermore, the PSO runs much faster which makes it more effective in online optimization.**

**Keywords-** PSO; GA; RBF Neural Network; Online Optimization; FCC.

### I. INTRODUCTION

Online optimization aims to increase the industrial potential of chemical processes and achieve the greatest economic efficiency. Both of the GA and the Swarm Intelligence (SI) algorithms are current research hotspots in chemical process optimizations. They have caused wild public concern since the 1980s and the 1990s respectively. They are two of the most powerful methods to solve the unconstrained and constrained global optimization problems. GA can be easily applied to the problems involving nondifferentiable functions and discrete search spaces. They have been applied to the fields such as engineering, economics and artificial intelligence in various areas and their application in chemical industry is becoming more and more widespread [1]-[2]. SI is an emerging area in the field of optimization and various algorithms have been developed by modeling the behaviors of different swarm of animals and insects such as ants, termites, bees, birds and fishes. The theoretical investigation of this kind of algorithm is in the initial stage at present. But they have outstanding advantages [3] such as: (1) the individual agents of a swarm are distributed and they can behave without supervision and each of these agents has a stochastic behavior. This enables the swarm to use their environmental resources effectively. (2) Without the control and restraint from the center, the system is more robustious. The failure of one or several individual agents will not affect the whole solving of the problem. (3) The individual agents collaborate through stigmergy but not

direct transmission. This makes the system have better scalability, because the addition of agents will cost little transmission computation of the system. (4) The computation of each agent is simple and fast. So the algorithm can be applied in various complicated problem easily and it runs very fast which is important for online optimization. PSO and Ant Colony Optimization (ACO) are the two most popular kinds of SI algorithm. Both of them were introduced in the early 1990s and they have been applied to solve optimization problems in various areas during these two decades. However, the convergence proof of ACO algorithm needs to be improved and its convergence rate is unsatisfactory. Also, it is weak in optimizing the complicated hybrid problem [4].

Based on the current research status mentioned above, this paper mainly focuses on the online applications of the PSO and the GA in chemical industry. In Section II, the PSO and the GA are introduced briefly. Section III presents the RBF neural network model of FCC unit and the optimization results of the two algorithms were compared experimentally. Analyses and discussions on the two optimization algorithms are made in Section IV. Finally, conclusions are drawn in Section V.

### II. PSO AND GA

#### A. Genetic Algorithm

GAs are adaptive search techniques based on the principles and mechanisms of natural selection and 'survival of the fittest' from natural evolution. Unlike various constructive procedures which use sophisticated heuristics to obtain a good single solution, GAs are dealing with a set of solutions and tend to manipulate each solution in the simplest way.

The basic steps of GA are [5]:

1) Generation of initial population of randomly constructed solutions (a solution is represented by a problem-specific structure of characters or bits).

2) Random selection of two solutions and generation of new solutions using crossover procedures which are supposed to provide inheritance of some basic properties of parent structures by the offsprings.

3) Mutation of child structures with probability  $P_m$ , which implies exchange of elements between two randomly selected positions in a structure.

4) Decoding/evaluation of the child structures to obtain the solutions of objective functions.

5) Selection procedures including comparison of child solutions with the worst solutions in the population and replacement of the worst solution by the new one if it is better. If the population after selection contains equal structures, the redundant structure is being removed and the population size is decreased.

6) Termination of the algorithm after repetition of steps 2-5 R times, where R is an initially specified parameter.

### B. Particle Swarm Optimization [6-7]

PSO, which was introduced by Kennedy and Eberhart in 1995, is one of the most important swarm intelligence paradigms. The PSO uses a simple mechanism that mimics swarm behavior in birds flocking and fish schooling to guide the particles to search for globally optimal solutions. Due to PSO's easily implement, it is progressing rapidly in recent years and with many successful applications emerged in solving real-world optimization problems.

In PSO, a swarm of particles are represented as potential solutions, and each particle  $i$  is associated with two vectors, i.e., the velocity vector  $V_i = [v_i^1, v_i^2, \dots, v_i^D]$  and the position vector  $X_i = [x_i^1, x_i^2, \dots, x_i^D]$ , where  $D$  stands for the dimensions of the solution space. The velocity and the position of each particle are initialized by random vectors within the corresponding ranges. During the evolutionary process, the velocity and position of particle  $i$  on dimension  $d$  are updated as:

$$v_i^d = \omega v_i^d + c_1 r_1^d (pBest_i^d) - x_i^d + c_2 r_2^d (nBest^d - x_i^d). \quad (1)$$

$$x_i^d = x_i^d + v_i^d. \quad (2)$$

Where  $\omega$  is the inertia weight,  $c_1$  and  $c_2$  are the acceleration coefficients,  $r_1^d$  and  $r_2^d$  are two uniformly distributed random numbers independently generated within  $[0, 1]$  for the  $d$ th dimension. In (1),  $pBest_i$  is the position with the best fitness found so far for the  $i$ th particle, and  $nBest$  is the best position in the neighborhood.

Given its simple concept and effectivity, the PSO has become a popular optimizer and has been widely applied in practical problems. Thus, theoretical studies and performance improvements of the algorithm are important and attractive.

## III. MODELING AND ONLINE OPTIMIZATION

### A. Modeling of Industrial FCC Unit

FCC (Fluid Catalytic Cracking) unit is one of the most important units in oil refinery. Its economic benefit is attractive. So how to improve its operation level is paid a close attention by the circle of petrochemical works. No doubt one of the effective methods is to implement its online optimization and advanced control.

Based on the description in [8], the RBF-PLS model of an industrial FCC unit is built. The output of this network is the total percent conversion, which is defined as:

$$conversion = \frac{gasoline + carbon\ coke + diesel\ oil}{fresh\ raw\ oil + recycle\ oil}. \quad (3)$$

According to the analysis in [8], the eight inputs of this RBF-PLS model include the outlet temperature of riser ( $T_{out}$ ), the pressure drop of regeneration valve ( $\Delta P_{rv}$ ), the regeneration valve opening ( $D_{rv}$ ), the pressure of riser ( $P$ ), the property factor of raw materials ( $f$ ), the recycle ratio ( $R$ ), the feeding of the first stage ( $F_1$ ) and the feeding of the second stage ( $F_2$ ). The data used in this paper are all collected from the oil refinery of Guangzhou Branch Company of the China Petroleum & Chemical Corporation (SINOPEC). With the help of the self-organizing pattern library mentioned in [9], 110 pairs of input-output data points are selected as the training samples and the other 330 pairs are selected as the testing samples. Table I shows that the RBF-PLS model is effective and reliable. Besides, its learning speed is very fast. The average time of 30 training tests is 0.83s.

TABLE I The Modeling Results of RBF-PLS Network

Model	Mean Square Error (MSE)	
	Simulation	Generalization
RBF-PLS	0.068	0.100

### B. The Comparative Tests of Online Optimization

Considering the final purpose of this paper is to verify the effectivity of the optimization algorithms, the following assumptions are made:

1) Taking no account of the controllability of the variables. It means that any variable selected as controlled variable is assumed controllable.

2) Using the generalization results of the RBF-PLS network but not the original data to compare the two optimization algorithms.

Comparative tests have been performed and the results are presented in the following section.

#### a) Single-Objective Optimization

Firstly, the regeneration valve opening ( $D_{rv}$ ) is chosen as the target of the optimization test. The results of GA and PSO are shown in Fig. 1 and Fig. 2, respectively. The red curves represent the conversion after the optimization while the black ones denote the original data. The figures show that the optimization performance of PSO is similar to that of GA. The two algorithms both increase the conversion and they are more effective when the original conversion is low. Besides, the PSO is tested using the population size of 20, a value commonly adopted in PSO [7]. For a fair comparison, the two algorithms use the same number of  $2.0 \times 10^3$  FEs. Relatively speaking, the settings of the GA are much more difficult. Except the population size and the max number of iteration, the cross coefficient, the cross rate, the mutation rate and the way of mutation also need to be selected seriously.

Table II shows the difference of GA and PSO with regard to the speed of this optimization. It's quite obvious that the PSO, owing to its simplicity and high efficiency,

runs faster than the GA.

The values of  $D_{rv}$  before and after the PSO optimization are shown in Fig. 3. The blue curve represents the value after the optimization while the black one denotes the original data. It illustrates that higher  $D_{rv}$  is good for improving the conversion. This is consistent with the actual condition in which higher  $D_{rv}$  means more catalysts, such

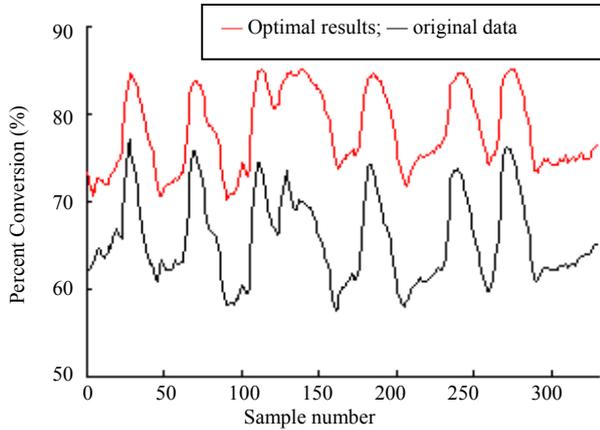


Fig. 1 The results after  $D_{rv}$  been optimized by PSO.

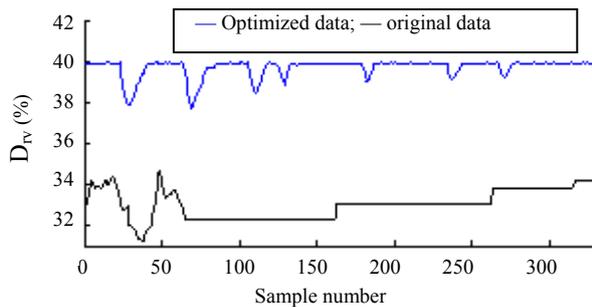


Fig. 3 The values of  $D_{rv}$  before and after it been optimized by PSO.

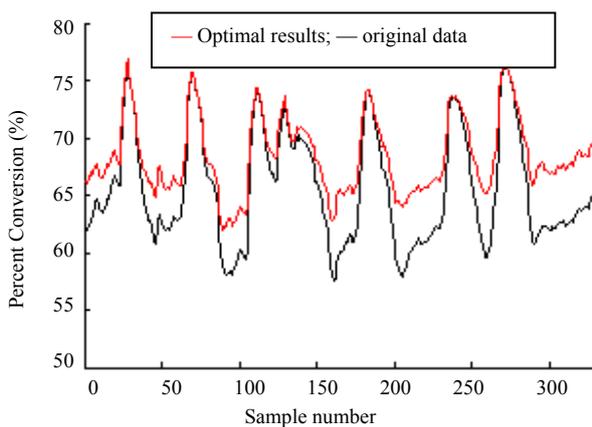


Fig. 4 The results after F2 been optimized by PSO.

that more active center is available in the catalysts for the fuel oil, which is equal to extending the reaction time. That is why the higher conversion is obtained.

Other tests are carried out on the feeding of the second stage ( $F_2$ ). And the results which are similar to the optimization of  $D_{rv}$  are listed in Fig. 4, Fig. 5 and Table III.

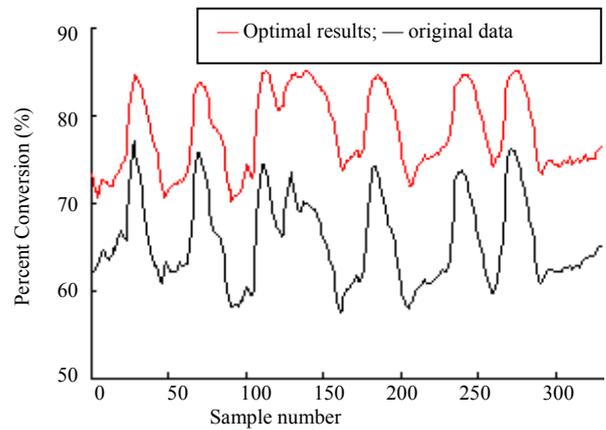


Fig. 2 The results after  $D_{rv}$  been optimized by GA.

TABLE II The Comparison of Conversion Rate and Computing Time after  $D_{rv}$  been Optimized

Algorithm	Average Conversion Rate (%)		Average Computing Time(s)
	Model Data	Optimization Result	
PSO	65.56	77.93	0.54
GA	65.56	77.96	18.56

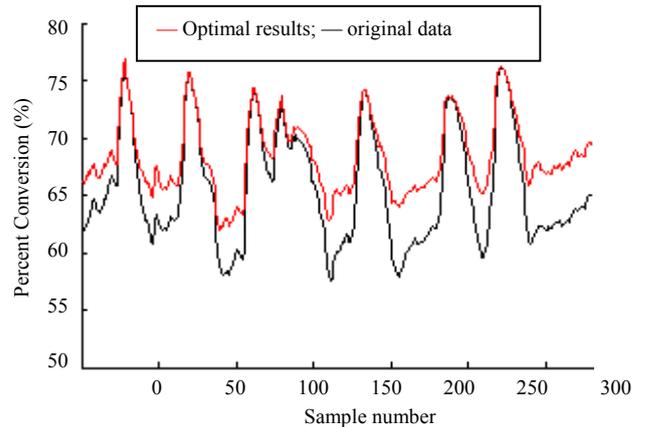


Fig. 5 The results after F2 been optimized by GA.

TABLE III The Comparison of Conversion Rate and Computing Time after  $F_2$  been Optimized

Algorithm	Average Conversion Rate (%)		Average Computing Time(s)
	Model Data	Optimization Result	
PSO	65.56	69.24	0.70
GA	65.56	69.31	16.34

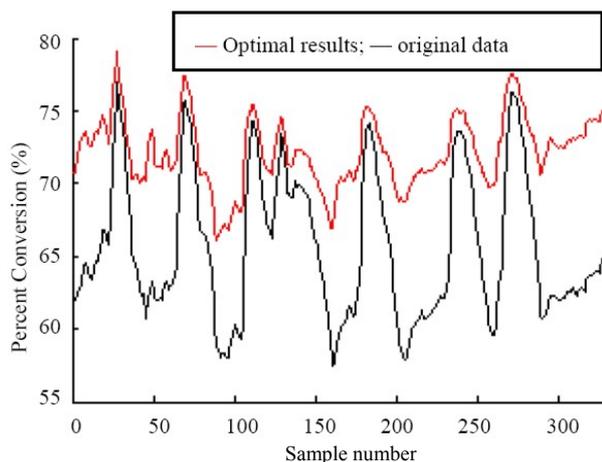


Fig. 6 The results after  $D_{rv}$  and  $F_2$  been optimized by PSO.

TABLE IV The Comparison of Conversion Rate and Computing Time after  $D_{rv}$  and  $F_2$  been Optimized

Algorithm	Average Conversion Rate (%)		Average Computing Time(s)
	Model Data	Optimization Result	
PSO	65.56	72.15	0.64
GA	65.56	72.20	18.88

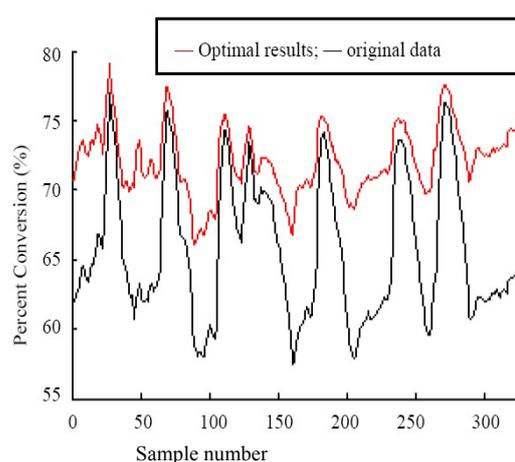


Fig. 7 The results after  $D_{rv}$  and  $F_2$  been optimized by GA.

#### b) Multiple-Objective Optimization

The optimization of  $D_{rv}$  and  $F_2$  are implemented simultaneously and the results are listed in Fig. 6, Fig. 7 and Table IV. The similar conclusion to the single-objective optimization is obtained from the results. Actually, the optimization performance of the PSO is very similar to the GA in this test. But in terms of the computational speed, the PSO has absolute advantage.

### IV. ANALYSES

#### A. Effectivity

Based on the above results, the optimization results of PSO and GA seem to have little differences, and they are both effective in single-objective as well as multiple-objective optimizations to some extent.

It is well known that accurate mathematical model is very important for any optimizations. All the optimization results above are compared with the output of the RBF neural network. It means the mathematical model used in the tests of this paper is completely accurate. Thus, errors from mathematical model are avoided entirely in the results of the algorithms discussed, which help to clarify the effectivity of the algorithms directly. In practical applications, errors brought by model must be taken into account. In other words, the final optimization results should be compared with the original sample data rather than the output of neural network.

#### B. Optimization time

Online optimization puts forward very strict requirements on the optimization rate of algorithm. Data collecting, selecting and modeling of chemical process are time-consuming, so the optimization algorithm can not be used in the online process if its optimization time is too long. According to the above comparison results, the PSO obviously costs much less time than the GA. In both single-object and multiple-object optimizations, the GA takes at least 20 times as much time as the PSO. Optimization rate and algorithm complexity are closely correlated. Crossover and mutation in GA are the two most time-consuming steps, whereas PSO can rapidly obtain the global optimal point through simple and efficient random searching.

#### C. Applicability

The structure of PSO is simple and it is easy to implement in various complicated problems. On the other hand, PSO needs relatively less parameters which simply include the population size and the termination condition. Generally, population size of 20, precision accuracy of  $1 \times 10^{-5}$  and about  $2.0 \times 10^3$  FEs can come to a fairly good optimization result. GA is relatively more complex and it needs more parameters. Moreover, its crossover and mutation operations cost a lot of time, and the specific structure parameters of these operations are hard to be determined. All these make the application of GA relatively more difficult.

## V. CONCLUSION

After the comparison tests of optimization implemented on industrial FCC unit, this paper comes to the conclusion that the performance of PSO in optimization is as excellent as that of GA. The specific numerical values of the conversion increased by PSO and GA in three of the experiments are 12.37% and 12.40%, 3.68% and 3.75%, 6.59% and 6.64%, respectively. However, as the simple structure and fast running, the PSO performs better than GA in terms of optimization time and applicability. So PSO is relatively better than GA in online optimization of chemical processes.

## REFERENCES

- [1] C. M. Castell, R. Lakshmanan and J. M. Skilling, "Optimization of process plant layout using genetic algorithms," *Comput. & Chem. Eng.*, 1998(22S), pp. 993-996.
- [2] T. Y. Park and G. F. Forment, "A hybrid genetic algorithm for the estimation of parameters in detailed kinetic models," *Comput. & Chem. Eng.*, 1998(22S), pp.103-110.
- [3] Y. C. Liang, C. G. Wu, X. H. Shi and H. W. Ge, *The Theory and Application of Swarm Intelligence*, Beijing: Science Press, 2009, pp.250-255.
- [4] L. D. Pan, *The Advanced Control and On-line Optimization*, Beijing: China Machine Press, 2009. pp.182-185.
- [5] J. Rubinovitz, G. Levitin, "Genetic algorithm for assembly line balancing," *Production Economics*, 1995(41), pp.343-354.
- [6] J. Kennedy, R. C. Eberhart and Y. H. Shi, *Swarm Intelligence*, San Mateo, CA: Morgan Kaufmann, 2001.
- [7] R. C. Eberhart and Y. H. Shi, "Particle swarm optimization: Developments, applications and resources," *IEEE Congr. Evol. Comput., Seoul, Korea*, pp. 81-86, 2001.
- [8] J. Chai, Research on Application of AI for Optimal Control in FCCU, Xiamen University, 2002.
- [9] Q. Y. Jiang, Study on Intelligent Optimal Control Technique, University of Science and Technology, 1990.