Energy Cost Minimization of a Compressor Station by Modified Genetic Algorithms

X. Zhang, C. Wu

***Abstract*—Energy cost minimization of a compressor station is an integral part of operation optimization for gas pipelines. Given the suction flow rate, the suction pressure and temperature, and the required discharge pressure of a compressor station, the operator needs to figure out the optimal compressor combination and load distribution of the station. To investigate the feasibility of genetic algorithms for solving this problem and to examine how the coding sequence of a genetic algorithm influences its performance, four genetic algorithms which are different in aspect of coding method and coding sequence were devised for this problem. These four algorithms are tested on multiple case problems of two in-service compressor stations. Comparison of the four algorithms shows that the coding sequence of a genetic algorithm influences its ability to find a feasible solution. The weaker this ability is, the more severely the algorithm is impacted. However, once any feasible solution is found, the coding sequence just impacts slightly on how steady a genetic algorithm performs in solving a problem multiple times, and no obvious bias is observed. According to the comparison, one of the four genetic algorithms was chosen to compare with two global optimization approaches, and the results show that the genetic algorithm is comparable with these global optimization methods.**

***Index Terms*—coding sequence, compressor station, genetic algorithm, power optimization**

I. Introduction

PIPELINES are the most widely used and economical way to transport natural gas on land. When gas flows in a pipeline, its pressure decreases gradually due to friction. Compressor stations are located along the pipeline to compensate this pressure drop. Typically, these compressors consume 2% to 3% of the natural gas transported by the pipeline. Thus, even minor fuel reduction will lead to considerable profit, and minimizing the fuel consumption of a pipeline has attracted intense interest [1]-[6].

In this paper, the problem of how to minimize the energy cost of a compressor station was addressed. Usually, serval compressors are equipped in a station. These compressors may be the same or not in type, and are often arranged in parallel, as illustrated in Fig. 1 [7]. Given the suction pressure, P^s, the suction temperature, T^s, the total volumetric flow rate, Q_{iso}^{total}, and the required discharge pressure, P^d, of the station, the operation scheme of minimum energy cost is of interest. An operation scheme includes which compressors to run, i.e., the compressor combination, and how to distribute load among the running compressors.

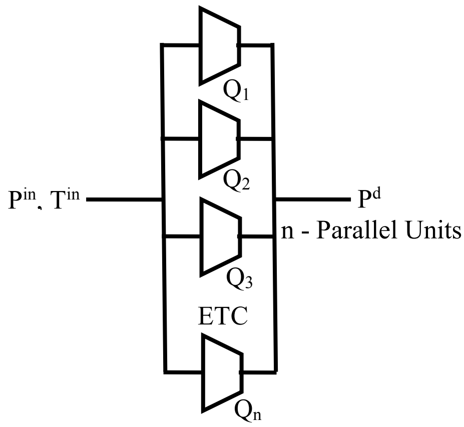

Fig. 1. Topology of a compressor station.

Much research has been done about this subject [8]-[10], [19], [20]. These studies often assume that the compressor combination has been prefixed, and only the load distribution problem is addressed [9], [10], [13], [14], [16], [17]. A simulation based optimization method was presented in [9], [10] to compute the optimal speed of each running compressor at transient state. In [13], a hybrid algorithm composed of generalized reduced gradient method and generalized projection gradient method was proposed to optimize the compressor speeds at steady state. Reference [14] adopted data-driven compressor models to compute the optimal load distribution. These models took the cooling water system of a multi-stage centrifugal compressor into consideration. However, they are accurate in predicting the power of a compressor within limited operating region [15]. A framework in which the optimal compressor combination and load distribution of a station were decided in real time was developed in [16]. However, only the load distribution problem was studied in detail. Similar problem was addressed in [17] with adaptive data-driven compressor models [18].

Research optimizing the compressor combination and load distribution of a station simultaneously is rare [7], [8], [11], [39]. Heuristics are often adopted to decide the compressor combination due to their low computational labor and robustness [7], [11], [12]. However, only local optimal should be expected for heuristic-based methods.

Manuscript received November 21, 2014; revised July 6, 2015. This work was supported in part by the Australian and Western Australian Governments and the North West Shelf Joint Venture Partners, as well as the Western Australian Energy Research Alliance under the Australia-China Natural Gas Technology Partnership Fund Top Up Scholarship.

X. Zhang and C. Wu are with the National Engineering Laboratory for Pipeline Safety, China University of Petroleum (Beijing), Changping, Beijing, 102200, China (phone: +86-10-8973-4398; e-mail: zxri8700@gmail.com, wucc@cup.edu.cn).

Sometimes, the fuel of a compressor unit is approximated by a linear or quadratic function. Then, the optimization problem is formulated as a mixed integer programing problem or a quadratic programming problem [7], and global optimal solution can be expected. However, the fuel of a compressor unit can be highly nonlinear, and approximation by a linear or quadratic function is probably very coarse. Consequently, the optimization results are less reliable.

Global optimization methods have also been adopted to minimize the energy cost of a compressor station. Reference [8] provided a detailed discussion of the Simulated Annealing algorithm as a solution method for determining the optimum combination and power settings for multiple compressors where the number of compressors is large and arranged in serial or parallel. The main shortage of the method is its slow convergence rate. A dynamic programming approach was reported in [39]. The approach can yield the optimal compressor combination and load distribution of a station simultaneously and robustly. However, its calculation labor rises dramatically as the step size discretizing the feasible flow rate region of a compressor decreases.

Genetic algorithms (GAs) mimic the natural evolution process, and are a kind of intelligence algorithm. Due to their ease of implementation, robustness and high probability yielding a global optimal solution, genetic algorithms are widely used in solving various optimization problems [21]-[27]. For example, the energy variance of a production schedule was minimized by a genetic algorithm in [28], and reference [29] proposed a dual objective genetic algorithm to maximize the security offered to a task with minimum security overhead in the security critical grid scheduling.

Genetic algorithms have also been adopted to minimize the fuel of a compressor station, including single-objective approaches [30], [31] and multi-objective ones [33], [38]. A comparison among a genetic algorithm, a heuristic method and an exhaustive enumeration method was reported in [32]. A brief comparison between a dynamic programming approach and a genetic algorithm was also discussed in [39]. However, to the best knowledge of the authors, no detailed comparison between the genetic algorithms and other global optimization methods about minimizing the energy cost of a compressor station has been reported. In addition, influences of the coding sequence of a genetic algorithm have also not been studied in solving the same problem.

In this paper, the mathematical model of the energy cost minimization problem is first introduced in section 2. Four different genetic algorithms which are different in aspect of coding method and coding sequence are formulated in section 3. In section 4, these four algorithms are adopted to solve multiple case problems of two in-service compressor stations. The results are analyzed to examine how the coding sequence of a genetic algorithm influences its performance. In addition, one of the four algorithms is compared with two global optimization approaches to investigate its feasibility for solving this problem. Finally, the conclusion section closes this paper.

II. Mathematical Model

Given the suction pressure, P^s, the suction temperature, T^s, the total volumetric flow rate, Q_{iso}^{total}, and the required discharge pressure, P^d, of a compressor station, the problem of minimizing its energy cost is addressed here. This problem was formulated as follows [7], [32], [39]:

$$\min \Sigma_{i=1}^{N} f_i\left(P^s, T^s, Q_{iso,i}^d, P^d\right) \quad (1)$$

$$s.t.\ y_i Q_{iso,i}^{min} \le Q_{iso,i}^d \le y_i Q_{iso,i}^{max}, i = 1,2,...,N \quad (2)$$

$$y_i = 0,1, i = 1,2,...,N \quad (3)$$

$$\Sigma_{i=1}^{N}\left(Q_{iso,i}^d + Q_{iso,i}^{consum}\right) = Q_{iso}^{total} \quad (4)$$

The objective of the problem is to minimize the total energy cost of the compressors in a station, illustrated as (1). Here, N is the number of compressors in the station. The energy cost of a compressor, f, is influenced by its suction pressure, P^s, suction temperature, T^s, discharge pressure, P^d and flow rate Q_{iso}^d.

In the constraints, equation (2) defines the feasible flow rate region of each compressor, and (3) refers to the compressor states: 0 for stopped, 1 for running. In addition, equation (4) describes the flow rate balance, in which Q_{iso}^{consum} is the fuel consumption of a compressor unit.

It should be noted that only centrifugal compressors were considered in this paper. This is due to their wide applications in gas pipelines, whereas reciprocating compressors are rarely used. In the following, how to calculate the energy cost of a centrifugal compressor unit is described first. Then, a robust method computing the feasible flow rate region of a compressor is reported.

A. Energy Cost of a Compressor Unit

This section describes how to compute the energy cost of a compressor given its suction pressure, suction temperature, discharge pressure and flow rate. This includes two process: simulation of a compressor and simulation of its driver.

Simulating a compressor is basically solving an equation system composed of (5) to (7) and an equation of state. Among these equations, (5) and (6) are adopted to regress the performance map of the compressor. The relation of the compressor head, H, with its speed, S, and the volumetric flow rate under its suction conditions, Q_{ac}, is described by (5), and (6) shows the relation of the compressor efficiency, η_c, with its speed and flow rate. The coefficients in the two equations, $a_{0,0}$, $a_{0,1}$, ..., $a_{2,2}$, $b_{0,0}$, $b_{0,1}$, ..., $b_{2,2}$, are compressor specific, and they are computed by regressing its performance map.

Compression in a compressor was considered as a poly-tropic process in this paper to minimize the deviation between the reality and the optimal solution. The poly-tropic process is described by (7). Besides, (8) and (9) are adopted to compute the head and efficiency of the compression process respectively. In (8), m_v is the poly-tropic exponent, Z^s is the compressibility factor of the gas compressed by the compressor under suction conditions, R is the gas constant and M_{GAS} is the molar mass of the gas. The item k^{ave} in (9) is the average heat capacity ratio, which is calculated by (10) with the heat capacity ratio under suction conditions, k^s, and that under discharge conditions, k^d.

$$\begin{aligned} H/S^2 &= \left(a_{0,0} + a_{0,1}S + a_{0,2}S^2\right) \\ &+ \left(a_{1,0} + a_{1,1}S + a_{1,2}S^2\right)\left(Q_{ac}/S\right) \\ &+ \left(a_{2,0} + a_{2,1}S + a_{2,2}S^2\right)\left(Q_{ac}/S\right)^2 \end{aligned} \quad (5)$$

$$\eta_c = b_0 + b_1\left(Q_{ac}/S\right) + b_2\left(Q_{ac}/S\right)^2 + b_3\left(Q_{ac}/S\right)^3 \quad (6)$$

$$T^d/T^s = Z^s/Z^d\left(P^d/P^s\right)^{(m_V-1)/m_V} \quad (7)$$

$$H = \frac{m_V}{m_V-1} Z^s \frac{R}{M_{GAS}} T^s\left[\left(P^d/P^s\right)^{(m_V-1)/m_V} - 1\right] \quad (8)$$

$$\eta_c = \left(k^{ave}-1\right)\lg\left(P^d / P^s\right)/\lg\left(T^d / T^s\right)/k^{ave} \quad (9)$$

$$k^{ave} = \left(k^s + k^d\right)/2 \quad (10)$$

Once the previous equations system is solved, the energy cost of the compressor is calculated by (11) to (14). Among these equations, the compressor input power, P_{shaft}, is first calculated by (11) according to its mass flow rate, $\dot{m}$, and the mechanical efficiency, η_m, which was regarded as a constant. Then, if the compressor is driven by a gas turbine, (12) gives its fuel according to the low heat value of its fuel, LHV, and its driver efficiency, η_{driver}. However, if the compressor is driven by an electric motor, its fuel equals zero and (13) gives the driver input power. Finally, (14) computes its energy cost according the fuel unit price, c_{fuel}, or that of electricity, c_{ele}.

$$P_{shaft} = \left(H\dot{m} \right. \quad (11)$$

$$Q_{iso}^{consum} = P_{shaft}/LHV/\eta_{driver} \quad (12)$$

$$P_{driver} = P_{shaft}/\eta_{driver} \quad (13)$$

$$f = c_{fuel}Q_{iso}^{consum} + c_{ele}P_{driver} \quad (14)$$

If a compressor is driven by a gas turbine, its efficiency is calculated by (15) to (19). In (15), T_a and P_a are the atmospheric temperature and pressure on site, whereas $T_{a,0}$ and $P_{a,0}$ are the atmospheric temperature and pressure of design. In addition, the coefficients, e_0, e_1, ..., e_5, in it are gas turbine specific, and they are calculated by regressing the efficiency performance map of the gas turbine. However, if a compressor is driven by an electric motor, constant driver efficiency is adopted.

$$\begin{aligned} \eta_{gasturbine} &= e_0 + e_1\bar{n}_{PT} + e_2\bar{N}_{PT} + e_3\left(\bar{n}_{PT}\right)^2 \\ &+ e_4\left(\bar{N}_{PT}\right)^2 + e_5\bar{n}_{PT}\bar{N}_{PT} \end{aligned} \quad (15)$$

$$\bar{n}_{PT} = S/\sqrt{\theta} \quad (16)$$

$$\theta = T_a/T_{a,0} \quad (17)$$

$$\bar{N}_{PT} = P_{shaft}/\delta\sqrt{\theta} \quad (18)$$

$$\delta = P_a/P_{a,0} \quad (19)$$

B. Feasible Flow Rate Region of a Compressor

In minimizing the energy cost of a compressor station, its suction pressure, the suction temperature and discharge pressure are given. Consequently, these variables are also fixed for each compressor in the station. In this section, the feasible flow rate region of a compressor under these fixed conditions is computed.

The case in which a compressor is running was considered first. If the suction pressure, the suction temperature, and the discharge pressure of the running compressor are fixed, its head can be considered constant [32]. Consequently, its feasible flow rate region is a horizontal line segment on its performance map, illustrated as the solid line in Fig. 2. Besides, this feasible region is also influenced by the maximum available power of its driver, such as the dash line in Fig. 2. On the other hand, if a compressor is stopped, no flow rate is allowed to pass it.

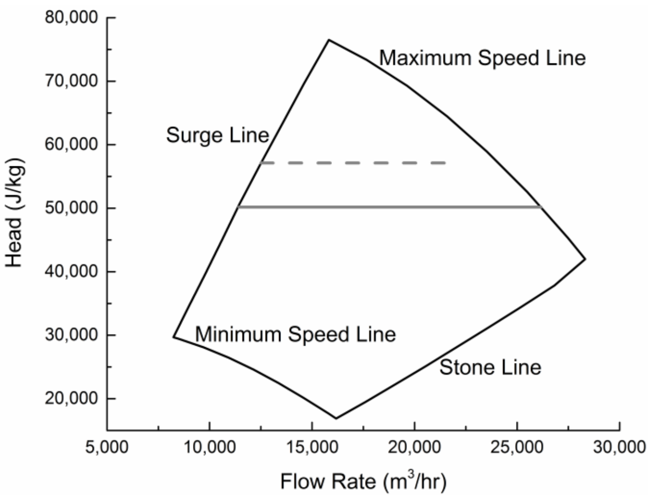

Fig. 2. Feasible flow rate region of a running compressor.

However, even if the suction pressure, the suction temperature, and the discharge pressure of a compressor are all fixed, its head varies slightly along with its flow rate fluctuation. Consequently, a simple but more reliable approach was devised to compute the feasible flow rate region of the compressor. In the following, the method calculating the minimum feasible flow rate is described. The method computing the maximum one is similar.

1. Calculate the flow rate of the point where the surge line and the minimum speed line meet, and that where the stone line and the maximum speed line meet. Mark them as Q_1 and Q_2 respectively.
2. Check whether the compressor is able to operate with the flow rate $(Q_1+Q_2)/2$ and the suction pressure, the suction temperature, and the discharge pressure specified in the energy cost minimization problem. This means checking whether the operating point lies within the operating envelope. The operating enveloped of a compressor is bounded by its surge line, stone line, minimum speed and maximum speed, illustrated as Fig. 2. If it is, go to 3; otherwise, go to 4.
3. Let $Q_2 = (Q_1+Q_2)/2$, go to 5.
4. Let $Q_1 = (Q_1+Q_2)/2$, go to 5.
5. Check whether $|Q_2-Q_1| <= \varepsilon_{frbound}$. If the inequality is fulfilled, go to 6; otherwise, go to 2.
6. Convert Q_2 to the volumetric flow rate under standard conditions, and this is the minimum feasible flow rate.

As stated previously, the maximal feasible flow rate of the compressor can be computed by a similar procedure. Thus, the feasible flow rate region of a compressor is $0 \cup \left[Q_{iso}^{\min}, Q_{iso}^{\max} \right]$.

III. Genetic Algorithm Design

To investigate the influences of the coding sequence of a genetic algorithm on solving the energy cost minimization problem, four different genetic algorithms are formulated in this section. A genetic algorithm is basically an iterative process, illustrated as Fig. 3. It starts from initializing a set of solution candidates, which form a population. And a solution candidate should be coded in proper form to be handled by the genetic operators. Then, each solution candidate, or individual, in the population is evaluated about how optimal it is. In this evaluation process, if the problem addressed is constrained, proper constraints handling method is necessary. Then, some genetic operators are adopted to handle the population. Commonly used operators include selection, crossover, and mutation. The algorithm continues until some stop criterion is satisfied. In the following, different aspects including solution coding, population initialization, constraint handling and genetic operators are discussed.

A. Solution Coding

To solve an optimization problem with genetic algorithms, solution candidates should be coded first. There are two kinds of coding method: binary coding and real coding. If a solution is coded in binary form, it is coded into a string of bits, 0 or 1. However, if real coding method is adopted, it is coded into a string of real numbers. And some of them are rounded to integers if these numbers correspond to the integer variables in the solution. Both of these two coding methods were adopted in this paper.

In addition to coding method, coding sequence is another problem need to be addressed. For a mixed integer nonlinear programming problem, a commonly used coding sequence is $(x_1, x_2, \ldots, x_m, y_1, y_2, \ldots, y_n)$, in which $x_1, x_2, \ldots, x_m$ are the real variables, whereas $y_1, y_2, \ldots, y_n$ denote the integer variables [21], [23], [34]. Besides, if the number of the real variables is equal to that of the integer ones, a specific coding sequence, $(x_1, y_1, x_2, y_2, \ldots, x_m, y_m)$, can be adopted [22]. These two coding sequences are named as the *common coding sequence* and the *specific coding sequence* respectively, *CCS* and *SCS* in short.

Thus, by combing different coding methods and coding sequences, four different genetic algorithms were formulated: two binary-coded GAs and two real-coded GAs. It should be noted the only difference is their coding sequence between the two binary-coded GAs. The same is true for the two real-coded GAs. More details about these two kinds of algorithms are stated in the following.

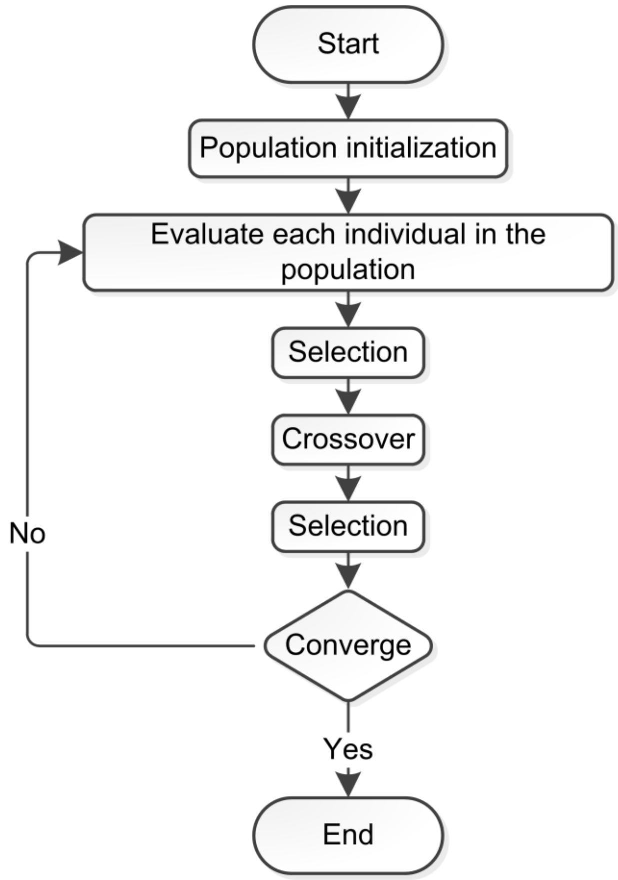

Fig. 3. Basic procedure of a genetic algorithm.

B. Population Initialization

To start a genetic algorithm, an initial population is needed. In this paper, the initial population was generated by random sampling. For a compressor, the sampling region of y is {0, 1} and that of Q is [$Q_{iso}^{\min}$, $Q_{iso}^{\max}$]. And its flow rate equals $y*Q$. Thus, if the compressor is stopped, no flow passes it. Otherwise, its flow rate is equal to Q, which has been guaranteed to be feasible.

C. Constraint Handling

During an iteration of a genetic algorithm, each individual in a population is evaluated about how optimal it is. This normally includes evaluating the objective function. Besides, if the problem addressed is constrained, checking whether any constraint is violated and evaluating how severely a constraint is violated are necessary.

In current problem, the constraints (2) and (3) were fulfilled by forcing a solution candidate to fall into these bounds. Only the flow balance constraint was addressed. Penalty function methods are the most popular ones to handle the constraints in an optimization problem [22], [24], [27], [35], [36]. However, fine parameter tuning is inevitable to obtain satisfactory algorithm performance. This can be time-consuming. To overcome this drawback, the method proposed in [21] was adopted. For current problem, it evaluates an individual according to (20) and (21). Illustrated as (20), if the flow balance error of an individual lies within the tolerance, it is considered feasible and evaluated by its objective function. Otherwise, it is regarded infeasible and is evaluated in another way. Here, the item, f_{max}, is computed according to (21). If there is no feasible solution in current population, f_{max} equals 0. Otherwise, f_{max} equals the maximum objective function value of all the feasible solutions in current population. Thus, the feasible solutions are evaluated according to their objective function values, whereas the infeasible solutions are evaluated based on their violations of the flow balance constraint. And feasible solutions are always in favor than infeasible solutions.

$$F(x)=\begin{cases} f(x), if \dfrac{\left|\Sigma_{i=1}^{N}\left(Q_{iso,i}^{d}+Q_{iso,i}^{consum}\right)-Q_{iso}^{total}\right|}{Q_{iso}^{total}}<\varepsilon_{eq} \\ f_{max}+\left|\Sigma_{i=1}^{N}\left(Q_{iso,i}^{d}+Q_{iso,i}^{consum}\right)-Q_{iso}^{total}\right|, otherwise \end{cases} \quad (20)$$

$$f_{max}=\begin{cases} 0, no\ feasible\ solution\ exists\ in\ the\ population \\ \max\left(f(x)\right), f(x)\in\{f(x)\,|\,x\ is\ feasible\} \end{cases} \quad (21)$$

D. Genetic Operators

After each individual in a population is evaluated, several genetic operators are adopted to handle the population and to generate a new population. Basic genetic operators include selection, crossover and mutation. For the binary-coded GAs, roulette-wheel selection, uniform crossover, and uniform mutation were adopted [27], [37], whereas tournament with niching, simulated binary crossover, and polynomial mutation were utilized in the real-coded genetic algorithms [21].

To prevent losing the best solution found during the algorithm process, elitism was also adopted in the four GAs. In the elitism operator, some optimal individuals of the previous population are directly copied to the current population. Care should be taken about how many individuals are directly copied. It the number is high, fast convergence rate will be achieved. However, the risk of premature rises at the same time. This means that the genetic algorithm is probably trapped at some local optimal solution.

IV. ALGORITHM TEST AND ANALYSIS

To investigate how the coding sequence of a genetic algorithm influences it on minimizing the energy cost of a compressor station, the four previously designed GAs were tested on two in-service compressor stations of different scale. One of the two in-service stations is equipped with four identical compressors. Some key details are listed in Table I, and Table II tabulates the coefficient values of the functions describing the characteristics of a compressor set. These equations include (5), (6), (15), (22), (23), among which (22) and (23) describe the surge line and the stone line of a compressor respectively.

TABLE I
DETAILS OF A COMPRESSOR

Item	Value
Driver	Gas Turbine
Maximum Driver Power(kW)	30,680
Minimum Speed	3,965
Maximum Speed	6,405

TABLE II
COEFFICIENT VALUES OF FUNCTIONS DESCRIBING A COMPRESSOR SET

Item	Value	Item	Value
$a_{0,0}$	2.42E-03	c_0	5.83E+03
$a_{0,1}$	-1.26E-07	c_1	-5.00E-01
$a_{0,2}$	7.12E-12	c_2	3.13E-04
$a_{1,0}$	-1.62E-04	d_0	3.58E+01
$a_{1,1}$	9.09E-08	d_1	3.81E+00
$a_{1,2}$	-5.25E-12	d_2	1.20E-04
$a_{2,0}$	-4.56E-05	e_0	6.98E-01
$a_{2,1}$	-1.40E-08	e_1	7.57E-03
$a_{2,2}$	1.04E-12	e_2	4.03E-04
b_0	6.76E-01	e_3	1.33E-07
b_1	9.09E-02	e_4	-7.30E-07
b_2	1.16E-02	e_5	-1.66E-08
b_3	-6.44E-03		

$$Q_{ac}^{surge}=c_0+c_1S+c_2S^2 \quad (22)$$

$$Q_{ac}^{stone}=d_0+d_1S+d_2S^2 \quad (23)$$

The other compressor station is equipped with five compressors, and some details of these compressors are listed in Table III. Notice that two different types of compressor are utilized in this station. Table IV shows the coefficient values of functions describing the characteristics of the two types of compressor. And Table V offers these that describe the efficiency characteristic of the gas turbine which drives the type-B compressor.

The composition of the natural gas boosted by the two compressor stations is tabulated in Table VI. Besides, the fuel unit price, c_{fuel}, is 1.6 Yuan/Sm^3, and that of the electricity, c_{ele}, is 0.56 Yuan/(kW*h).

To make the four genetic algorithms complete, some vital algorithm parameters were set as Table VII. In addition, the step size adopted to code a real variable into binary form was $10^{-4}*Q_{total}$, which is the typical measurement accuracy of a flow rate meter equipped on a gas pipeline. The mutation rate of the binary-coded genetic algorithms was 0.1. The tournament size and mutation rate of the real-coded genetic algorithms were identical to that adopted in [21]. Finally, in the elitism operation, the optimal individual of the former population was directly copied to current generation.

TABLE III
SOME DETAILS OF TWO COMPRESSORS

Item	Type-A	Type-B
Amount	3	2
Driver	Electric Motor	Gas Turbine
Driver Efficiency	0.98	--
Maximum Driver Power (kW)	20,000	30,000
Minimum Speed	6,000	2,400
Maximum Speed	10,500	5,040

TABLE IV
COEFFICIENT VALUES OF FUNCTIONS DESCRIBING A COMPRESSOR

Item	Type-A	Type-B
$a_{0,0}$	-1.77E-05	-5.13E-03
$a_{0,1}$	4.87E-07	4.57E-06
$a_{0,2}$	-4.39E-11	-9.14E-10
$a_{1,0}$	2.35E-03	8.34E-03
$a_{1,1}$	-5.58E-07	-2.70E-06
$a_{1,2}$	5.71E-11	5.37E-10
$a_{2,0}$	-1.40E-03	-1.88E-03
$a_{2,1}$	1.78E-07	4.54E-07
$a_{2,2}$	-1.72E-11	-7.81E-11
b_0	6.71E-01	1.90E+00
b_1	3.00E-02	-1.16E+00
b_2	3.23E-01	4.28E-01
b_3	-1.95E-01	-5.42E-02
c_0	3.58E+03	3.19E+03
c_1	-2.57E-01	1.56E-01
c_2	7.69E-05	3.16E-04
d_0	-1.74E+03	-1.59E+01
d_1	1.56E+00	2.89E+00
d_2	3.19E-05	2.38E-04

TABLE V
COEFFICIENT VALUES DESCRIBING EFFICIENCY CHARACTERISTIC OF A GAS TURBINE

Item	Value
e_0	-2.4083E+01
e_1	1.7077E-02
e_2	7.7093E-04
e_3	6.6169E-08
e_4	-1.6391E-06
e_5	-1.6373E-08

TABLE VI
NATURAL GAS COMPOSITION

Content	Molar Percentage (%)
C1	92.545
C2H6	2.41
C3H8	0.37
IC4	0.05
NC4	0.08
IC5	0.02
NC5	0.02
C6	0.06
N2	1.53
CO2	0.92
H2S	1.995

TABLE VII
ALGORITHM PARAMETERS

Item	Value
Population Size	20
Maximum Generation	100
Crossover Probability	0.8
$\varepsilon_{frbound}$	$10^{-4}*Q_{total}$
ε_{eq}	10^{-4}

To make the statement more concise, the four algorithms were named as GAbn, GAbs, GArn, and GArs respectively. In the name, “b” and “r” refer to the two coding methods, i.e., the binary-coding method and the real-coding method. And the character “n” and “s” refer to the two coding sequences, that is, the *common coding sequence* and the *specific coding sequence.*

In the following, the four genetic algorithms were utilized to minimize the energy cost of the two in-service compressor stations under different operation conditions. Each problem was solved 50 times independently due to the random nature of genetic algorithms. Based on the results, the algorithms were compared with each other from aspects of feasibility rate and optimal objective function values to investigate the influence of the coding sequences. The feasibility rate of an algorithm is defined as the proportion of the runs in which it finds any feasible solution out of its total runs. Finally, one of these four algorithms was chosen based on the comparison to compare with two other global optimization approaches.

A. Test on Station Equipped with Identical Compressors

59 case problems in total were computed. The parameters describing a case problem vary among the ranges listed in Table VIII. And each problem was solved by each algorithm 50 times independently, stated as before.

TABLE VIII
PROBLEM PARAMETERS

Item	Value
Flow Rate ($\times 10^4$ Sm^3/hr)	100~400
Inlet Pressure (kPa)	6,000~7,000
Inlet Temperature (℃)	5~20
Discharge Pressure (kPa)	8,000~9,800

It should be noted that minimizing the energy cost of a station equipped with identical compressors can be considered as an easy problem. This is because distributing the load among the running compressors is optimal or near optimal [31], [39]. Consequently, the only problem left is to decide how many compressors to run. Although this heuristic was not coded into the genetic algorithms, minimizing the energy cost of this station should be considered easier than the other one on account of the identical compressors equipped in the station.

According to the results, the feasibility rates of the four algorithms were calculated, and are depicted in Fig. 4. It shows that the real coded algorithms can always offer a feasible solution except for several problems, whereas the binary coded algorithms are inferior. In addition, the coding sequence of the real coded algorithm barely influences its feasibility rate, whereas the binary coded one is severely influenced.

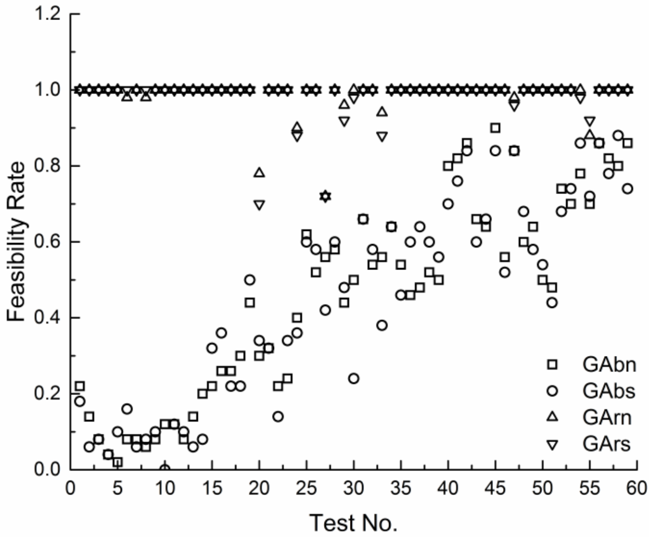

Fig. 4. Feasibility rate.

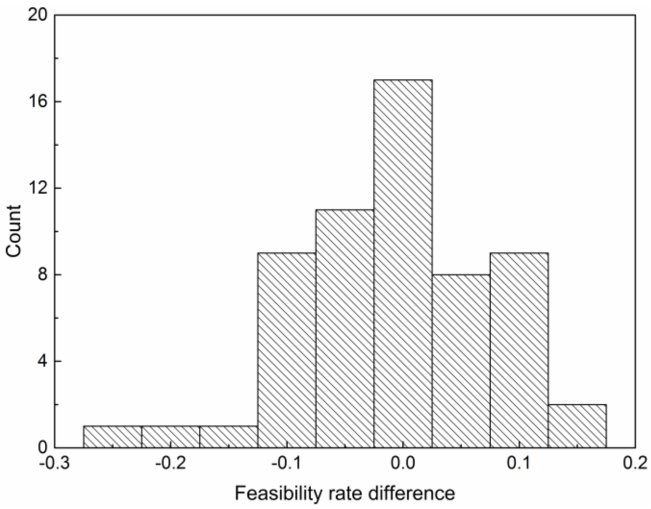

Fig. 5. Influence of coding sequence on feasibility rate (Binary Coding).

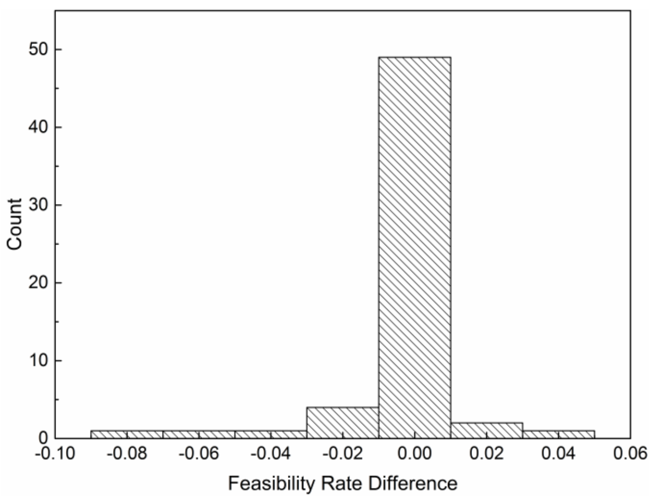

Fig. 6. Influence of coding sequence on feasibility rate (Real Coding).

For the two binary coded algorithms, the histogram of feasibility rate difference is plotted in Fig. 5. The difference is equal to the feasibility rate of GAbs minus that of GAbn. Fig. 5 shows that the feasibility rate of GAbs is a little lower than that of GAbn, which means that the *specific coding sequence* makes the algorithm slightly worse. The same but lighter influence is observed in the two real coded algorithms, illustrated as Fig. 6.

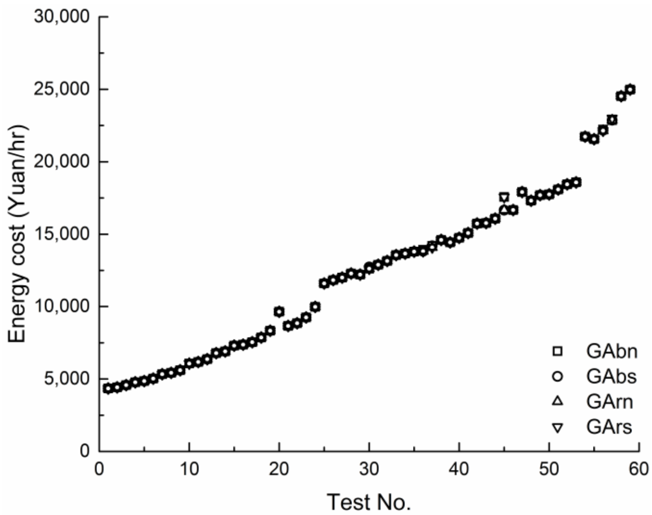

Fig. 7. Best solution of each case problem.

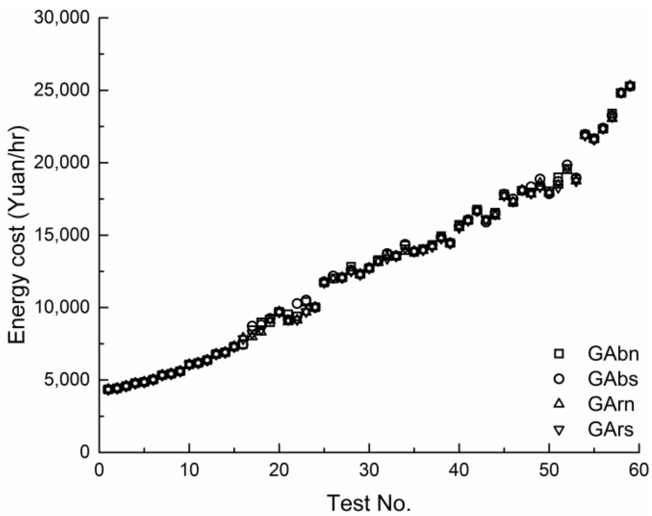

Fig. 8. Worst solution of each case problem.

For the feasible solutions resulted from multiple runs for the same problem, the best, the worst, and the standard error of these solutions were calculated for each algorithm. The best solution and the worst solution of each case problem are plotted in Fig. 7 and Fig. 8 respectively. It can be seen that the coding sequence barely influences.

For the two binary coded algorithms, the histogram of the standard error difference is plotted in Fig. 9. The difference is equal to the standard error of GAbs minus that of GAbn. No obvious bias can be observed in Fig. 9, whereas slightly higher standard error is found for GArs, as illstrated in Fig. 10.

In summary, the tests show that the coding sequence of a genetic algorithm influences its ability to find a feasible solution. And the weaker this ability is, the more severely the algorithm is influenced. In addition, adopting the *specific coding sequence* makes a genetic algorithm performs a little worse in aspect of finding a feasible solution.

Comparison of the feasible solutions resulted from multiple runs for the same problem reveals that the coding sequence influences the standard error of these solutions slightly. And the *specific coding sequence* makes that of the real coded algorithm a little bigger, whereas no influence bias is observed for the binary coded algorithm.

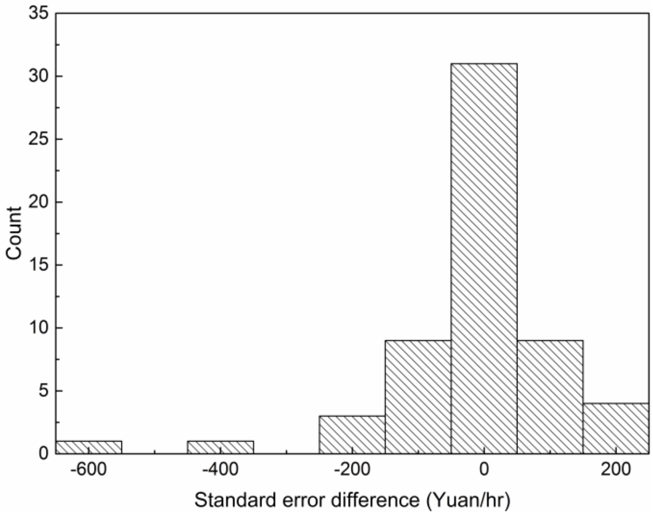

Fig. 9. Influence of coding sequence on standard error (Binary coding).

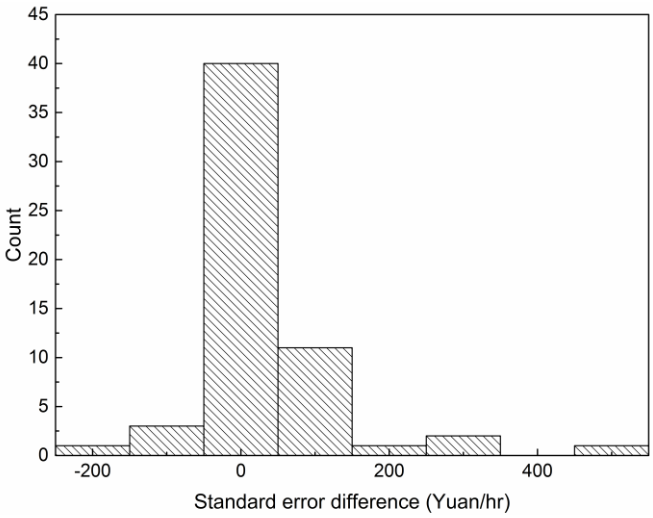

Fig. 10. Influence of coding sequence on standard error (Real coding).

B. Test on Station Equipped with Different Compressors

The four genetic algorithms were adopted to solve 72 case problems in total. And the parameters describing a case problem vary among the ranges listed in Table IX. It should be noted that minimizing the energy cost of a station equipped with compressors of different types is a harder problem compared with the former one.

TABLE IX
PROBLEM PARAMETERS

Item	Value
Flow Rate (×10^4 Sm^3/hr)	50~200
Inlet Pressure (kPa)	4,000~6,000
Inlet Temperature (℃)	5~20
Discharge Pressure (kPa)	8,000~9,800

As stated before, each problem was solved by each algorithm 50 time independently. According to the results, the feasibility rates of the four algorithms were calculated. The results are plotted in Fig. 11. Similar pattern is found as the former case study. The real coded algorithms are superior to the binary coded ones, and they are severely influenced by the coding sequence.

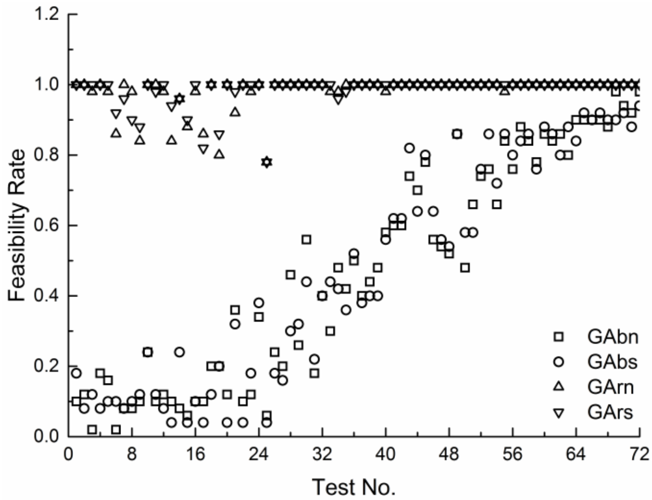

Fig. 11. Feasibility rate.

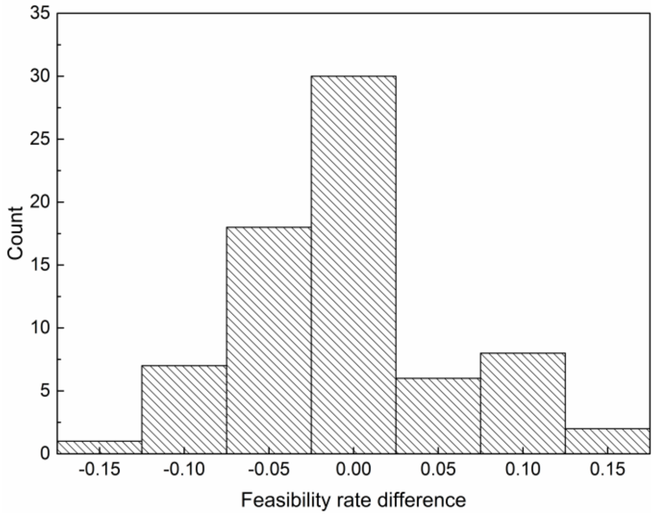

Fig. 12. Influence of coding sequence on feasibility rate (Binary coding).

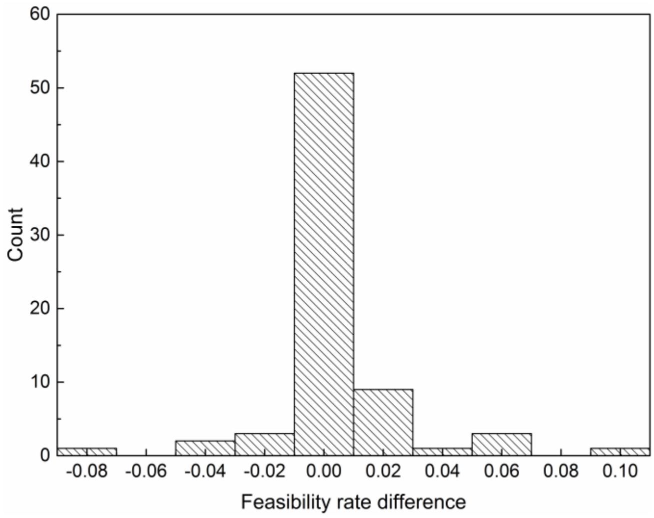

Fig. 13. Influence of coding sequence on feasibility rate (Real coding).

Fig. 12 depicts the histogram of the feasibility rate difference for the two binary coded algorithms. It shows again that the *specific coding sequence* makes the algorithm performs worse in aspect of finding a feasible solution. However, this coding sequence makes the real coded algorithm works slightly better, illustrated as Fig. 13.

For the feasible solutions resulted from multiple runs for the same problem, the best, the worst and the standard error of these solutions were calculated for each algorithm. And the best solution and the worst solution are plotted in Fig. 14 and Fig. 15 respectively. As the former case, almost no difference is found among different algorithms.

For the two binary coded algorithms, the histogram of the standard error difference is plotted in Fig. 16. It can be seen that adopting the *special coding sequence* slightly decreases the standard error. However, no influence bias was found for the real coded algorithms.

In summary, the tests show again that the coding sequence of a genetic algorithm influences its ability to find a feasible solution, and the weaker this ability is, the more severe the influence is. Besides, the *specific coding sequence* also makes the binary coded algorithm performs a little worse in this aspect. On contrast, this coding sequence makes the real coded algorithm performs a bit better.

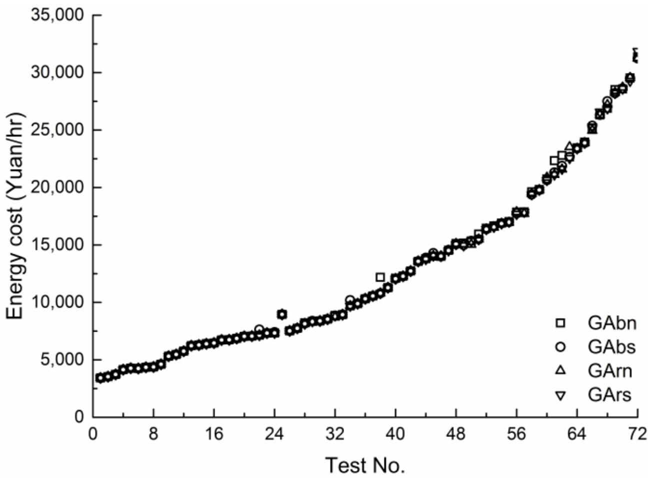

Fig. 14. Best solutions of each case problem.

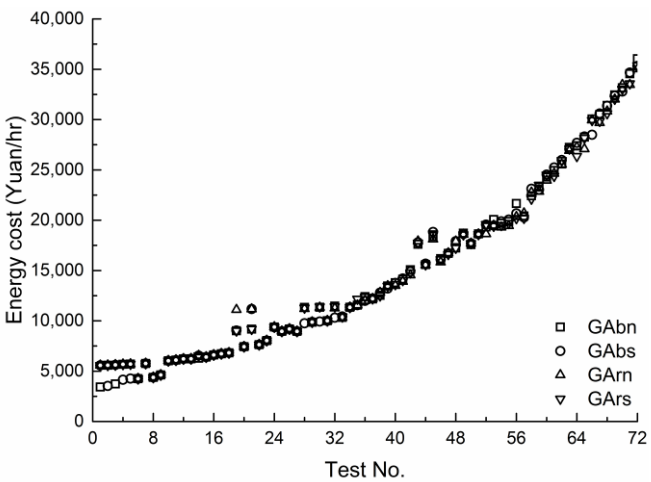

Fig. 15. Worst solution of each case problem.

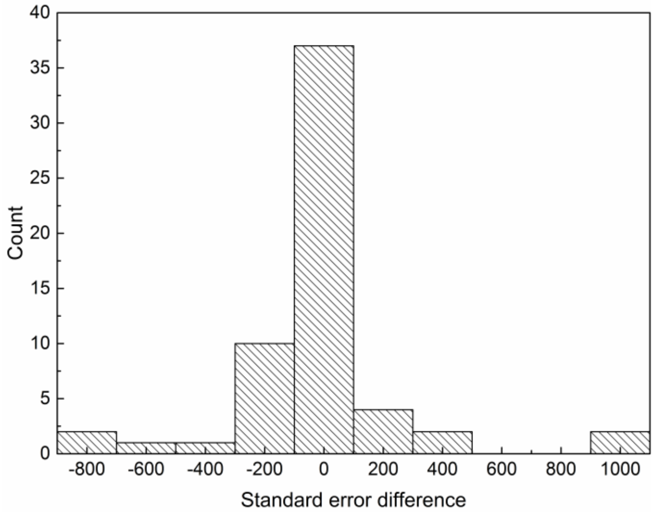

Fig. 16. Influence of coding sequence on standard error (Binary coding).

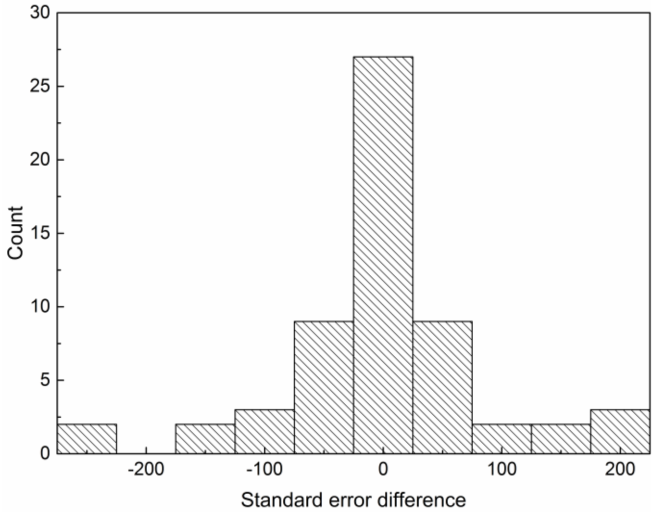

Fig. 17. Influence of coding sequence on standard error (Real coding).

Comparison of the feasible solutions resulted from multiple runs for the same problem reveals again that the coding sequence just impacts the standard error of these solutions slightly. And the *specific coding sequence* makes that of the binary coded algorithm a little smaller, whereas no influence bias is observed for the real coded algorithm.

C. Comparison with Global Optimization Approaches

To evaluate the feasibility of the genetic algorithms for minimizing the energy cost of a compressor station, the results of GArn were compared with two global optimization approaches. On account of the fact that previous study reveals that the four algorithms differ little in aspect of the best solution and the worst solution found in multiple runs, any of the four genetic algorithms can be chosen for the comparison.

As previously stated, if a station is equipped with identical compressors, only how many compressors to run need be decided to minimize its energy cost. And the load is distributed among the online compressors equally. This solution can considered as global optimum.

For the first set of case problems, the optimal amount of online compressors was computed by enumeration. This solution, together with the best, the worst and the average solution of GArn, is plotted in Fig. 18. Although the worst solution of GArn is modestly higher than the result of the equal-distribution approach, the best and the average solution of GArn are comparable with the result of the equal-distribution approach.

For the station equipped with compressors of different types, the dynamic programming approach reported in [39] was adopted to compute the global optimum. This optimum, together with the best, the worst, and the average solution of GArn, is plotted in Fig. 19. Similar pattern can be observed as the former case study. Although the worst solution of GArn is modestly higher than the global optimum, its average and best solution are comparable with the global optimum.

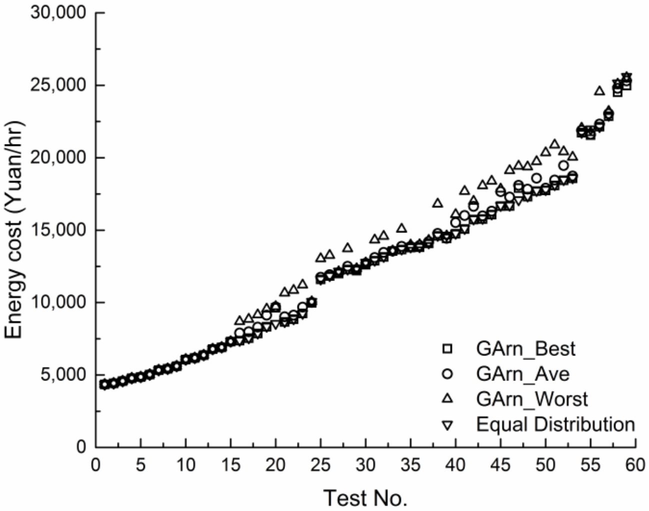

Fig. 18. Comparison of GArn with Equal-Distribution Approach.

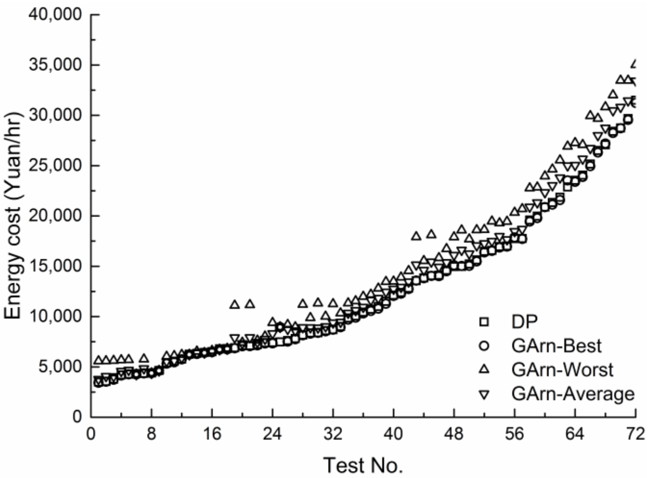

Fig. 19. Comparison of GArn with a dynamic programming approach.

V. Conclusion

By comparing four genetic algorithms which are different in aspect of coding method and coding sequence with each other and with two global optimization approaches, some conclusions can be made. First, the coding sequence of a genetic algorithm impacts its ability to find a feasible solution. And the weaker this ability is, the more severely the algorithm is influenced. In addition, for the algorithms discussed in this paper, the *specific coding sequence* can make the binary coded algorithm performs a little worse in this aspect, whereas no influence bias is observed for the real coded algorithm. Second, for the feasible solutions resulted from multiple runs of the same problem, the standard error of these solutions are just slightly influenced by the coding sequence, and no certain conclusion can be made about the influence pattern. Finally, according to the comparison with two global optimization approach, it can be concluded that genetic algorithms are comparable with these methods in minimizing the energy cost of a station. Take the universal feasibility of the genetic algorithms into account, more applications of genetic algorithms in gas pipeline industries should be expected.

The algorithms discussed in this paper are intended to be modified to minimize the energy cost of a gas pipeline at steady state and transient state. The comparison of the four genetic algorithms carried out in this paper is an important reference for future algorithm modification and design.

References

[1] P.J. Wong and R.E. Larson, “Optimization of natural-gas pipeline systems via dynamic programming,” *IEEE Transactions on Automatic Control*, vol. AC-13, no. 5, pp. 475-481, Oct. 1968.

[2] R.G. Carter, “Pipeline optimization: Dynamic programming after 30 years,” in the *30th Annual Meeting of the Pipeline Simulation Interest Group*, Colorado, 1998.

[3] S. Wu, R.Z. Rios-Mercado, E.A. Boyd, and L.R. Scott, “Model relaxations for the fuel cost minimization of steady-state gas pipeline networks,” *Mathematical and Computer Modelling*, vol. 31, no.2, pp. 197-220, 2000

[4] S. Moritz, “A mixed integer approach for the transient case of gas network optimization,” Ph.D. dissertation, TU Darmstadt, Darmstadt, Germany, 2007.

[5] B.T. Baumrucker and L.T. Biegler, “MPEC strategies for cost optimization of pipeline operations,” *Computers & chemical engineering*, vol. 34, no. 6, pp. 900-913, Jun. 2010.

[6] S. Alfred, J. Fasullo, J. Pfister and A. Daniels, “Capacity determination using state finding and gas transient optimization,” in the *44th Annual Meeting of the Pipeline Simulation Interest Group*, Prague, Czech Republic, 2013

[7] R.G. Carter, “Compressor station optimization: Computational accuracy and speed,” in the *28th Annual Meeting of the Pipeline Simulation Interest Group,* California, 1996.

[8] S. Wright, M. Somani and C. Ditzel, “Compressor station optimization,” in the *30th Annual Meeting of the Pipeline Simulation Interest Group*, Colorado, 1998.

[9] P. Krishnaswami, K.S. Chapman and M. Abbaspour, “Fuel-efficient operation of compressor stations using simulation-based optimization,” in *Preceedings of IPC 2004*, Alberta, Canada, 2004.

[10] P. Krishnaswami, K.S. Chapman and M. Abbaspour, “Compressor station optimization for linepack maintenance,” in the *36th Annual Meeting of the Pipeline Simulation Interest Group,* California, 2004.

[11] A. Osiadacz, “Nonlinear programming applied to the optimum control of a gas compressor station,” *International Journal for Numerical Methods in Engineering*, vol. 15, no. 9, pp.1287-1301, Sep. 1980.

[12] A. Osiadacz and D.J. Bell, “A local optimization procedure for a gas compressor station,” *Optimal Control Applications and Methods*, vol. 2, no. 3, pp. 239-250, Sep. 1981.

[13] T. Jenicek and J. Kralik, “Optimized control of generalized compressor station,” in the *27th Annual Meeting of the Pipeline Simulation Interest Group*, New Mexico, 1995.

[14] D.P. Xenos, M. Cicciotti, A.E.F. Bouaswaig, N.F. Thornhill and R. Martinez-Botas, “Modeling and optimization of industrial centrifugal compressor stations employing data-driven methods,” In *Proceeding of ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*, Düsseldorf, Germany, Jun. 2014.

[15] D.P. Xenos, N.F. Thornhill, M. Cicciotti and A.E. Bouaswaig, “Preprocessing of raw data for developing steady-state data-driven models for optimizing compressor stations,” In *2014 UKACC International Conference on Control*, Loughborough, U.K, Jul. 2014.

[16] D.P. Xenos, M. Cicciotti, G.M. Kopanos, A.E.F. Bouaswaig, O. Kahrs, *et al.*, "Optimization of a network of compressors in parallel: Real Time Optimization (RTO) of compressors in chemical plants–An industrial case study," *Applied Energy*, vol. 144, pp. 51-63, Apr. 2015.
[17] F. Paparella, L. Domínguez, A. Cortinovis, M. Mercangoz, D. Pareschi, *et al.*, "Load sharing optimization of parallel compressors," In *2013 European Control Conference*, Zürich, Switzerland, Jul. 2013.
[18] A. Cortinovis, M. Zovadelli, M. Mercangoz, D. Pareschi, A. De Marco, *et al.*, "Online adaptation of performance maps for centrifugal gas compressors," In *2014 European Control Conference*, Strasbourg, France, Jun. 2014.
[19] V.S. Norstebo, L.E. Bakken and H.J. Dahl, "Optimum operation of export compressors," In *Proceeding of ASME Turbo Expo 2005: Power for Land, Sea and Air*, Montreal, Canada, May 2007.
[20] M. Pinelli, A. Mazzi and G. Russo, "Arrangement and optimisation turbocompressors in an off-shore natural gas extraction station," In *Proceeding of ASME Turbo Expo 2005: Power for Land, Sea and Air*, Nevada, 2005.
[21] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer methods in applied mechanics and engineering*, vol. 186, no. 2, pp. 311-338, Jun. 2000.
[22] T. Yokota, M. Gen and Y.X. Li, "Genetic algorithm for non-linear mixed integer programming problems and its applications," *Computers & industrial engineering*, vol. 30, no. 4, pp. 905-917, 1996.
[23] M. Danish, S. Kumar, A. Qamareen and S. Kumar, "Optimal solution of MINLP problems using modified genetic algorithm," *Chemical Product and Process Modeling*, vol. 1, no. 1, Oct. 2006.
[24] Y.X. Li and M. Gen, "Nonlinear mixed integer programming problems using genetic algorithm and penalty function," In *IEEE International Conference on Systems Man, and Cybernetics*, Beijing, Oct. 1996.
[25] B.K.S. Cheung, A. Langevin and H. Delmaire, "Coupling genetic algorithm with a grid search method to solve mixed integer nonlinear programming problems," *Computers & Mathematics with Applications*, vol. 34, no. 12, pp. 13-23, Dec. 1997.
[26] Y. Tan, G. Tan, Q. Tan, B. Yang and Y. Ye, "Hybrid genetic algorithm with chaotic local search for mixed integer programming problems," *Journal of Computational Information Systems*, vol. 8, no. 24, pp. 10223-10230, 2012.
[27] E.G. Shopova and N.G. Vaklieva-Bancheva, "BASIC—A genetic algorithm for engineering problems solution," *Computers & chemical engineering,* vol. 30, no. 8, pp. 1293-1309, Jun. 2006.
[28] C. Duerden, L.K. Shark, G. Hall and J. Howe, "Minimisation of energy consumption variance for multi-process manufacturing lines through genetic algorithm manipulation of production schedule," *Engineering Letters* vol. 23, no. 1, pp. 40-48, 2015.
[29] R. Kashyap and D.P. Vidyarthi, "Dual objective security driven scheduling model for computational grid using GA," *IAENG International Journal of Computer Science*, vol. 39, no. 1, pp. 71-79, 2007.
[30] B. Fahimnia, R. Molaei and M. Ebrahimi, "Genetic algorithm optimization of fuel consumption in compressor stations," *WSEAS Transactions on Systems and Control*, vol. 3, no. 1, pp. 1-10, 2008.
[31] T.M. Elshiekh, "Optimization of fuel consumption in compressor stations," *Oil and Gas Facilities*, vol. 4, no. 1, pp. 59-65, Feb. 2014.
[32] J. Mahmoudimehr and S. Sanaye, "Minimization of fuel consumption of natural gas compressor stations with similar and dissimilar turbo-compressor units," *Journal of Energy Engineering*, vol. 140, no. 1, Apr. 2013.
[33] A. Hawryluk, K.K. Botros, H. Golshan and B. Huynh, "Multi-objective optimization of natural gas compression power train with genetic algorithms," In *2010 8th International Pipeline Conference*, Alberta, Canada, 2010.
[34] K. Deep, K.P. Singh, M.L. Kansal and C. Mohan, "A real coded genetic algorithm for solving integer and mixed integer optimization problems," *Applied Mathematics and Computation*, vol. 212, no. 2, pp. 505-518, Jun. 2009.
[35] M. Schluter and M. Gerdts, "The oracle penalty method," *Journal of Global Optimization,* vol. 47, no. 2, pp. 293-325, Oct. 2009.
[36] C.A.C. Coello, "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," *Computer methods in applied mechanics and engineering*, vol. 191, no. 11, pp. 1245-1287, Jan. 2002.
[37] D. Whitley, "A genetic algorithm tutorial," *Statistics and computing*, vol. 4, no. 2, pp. 65-85, Jun, 1994.
[38] K.K. Botros, H. Golshan, A. Hawryluk and B. Sloof, "Optimization of power train involving gas turbine driven compressors and aerial coolers," In *ASME 2011 Turbo Expo: Turbine Technical Conference and Exposition*, British Columbia, Canada, 2011.
[39] X. Zhang, C. Wu, L. Zuo and X. Meng, "Dynamic programming based algorithm for compressor station optimization," In *ASME 2014 International Mechanical Engineering Congress and Exposition*, Montreal, Canada, Nov. 2014.