Mix proportioning of high-performance concrete by applying the GA and PSO

Alireza rezaee[1], mohamad reza hasani ahangar[2]

[1]Ph.D. of Electrical Engineering, Islamic Azad university hashtgerd branch, hasthtgerd, Alborz, Iran. Email: arrezaee@yahoo.com
[2] Assistant Professor, center of ghadr, Imam hossein university, Tehran, Iran. Email: mrhassani@iust.ac.ir

Abstract

High performance concrete is designed to meets special requirements such as high strength, high flowability, and high durability in large scale concrete construction. To obtain such performance many trial mixes are required to find desired combination of materials and there is no conventional way to achieve proper mix proportioning. Genetic algorithm is a global optimization technique based on mechanics of natural selection and natural genetics and can be used to find a near optimal solution to a problem that may have many solutions. Particle swarm optimization is another evolutionary searching strategy motivated by social behaviors to obtain optimum answer. This paper presents a method whereby the mixture proportion of concrete can be optimized to reduce the number of trial mixtures with desired properties by using the genetic algorithm and particle swarm optimization techniques.

Keywords: High-performance concrete, Genetic algorithm, Particle swarm optimization, Mixture

1. Introduction

1.1 High-performance concrete

The American Concrete Institute (ACI) defines high-performance concrete as concrete meeting special combinations of performance and uniformity requirements that cannot always be achieved routinely by using conventional mix proportioning.

For many years, High-Performance Concrete (HPC) has been used in the column of high-rise buildings. How ever in recent years, there has been increased use of HPC in highway bridges, marine structures, aggressive environments, pavements and nuclear structures tunnels, etc. when this large volume of concrete is used for construction, the safety and durability of concrete become fundamental issues.

The major difference between conventional concrete and HPC is essentially the use of chemical and mineral admixtures. The use of chemical admixtures reduces the water content, thereby at the same time reduces the porosity within the hydrated cement paste. But reduction in the water content to a very low value with high dosage of chemical admixtures is undesirable. Because the effectiveness of chemical admixtures such as superplasticizer principally depends on the temperature, cement chemistry, and fineness. Mineral admixtures, also called as cement replacement materials, makes hardened cement matrix denser and stronger. Therefore the combined use of superplasticizer and cement replacement materials can lead to economical high-performance concrete with enhanced strength, workability, and durability [1].

There have not been any guide on the mix proportion of HPC and therefore mix proportion are obtained by trial and error methods based on existing data and conventional concrete mixture. Such methods needs large number of trail mixes to select the desired combination of materials. Thus a near optimum mix proportion of HPC is very important and useful to minimize the number of trial mixes to achieve economical and satisfactory mixture with desired properties [2].

This paper describes an evolutionary stochastic search technique for HPC mixtures using genetic algorithm and particle swarm optimization to minimize the number of trial mixes and provide appropriate mix proportion.

1.2 Genetic algorithm

The origin of Genetic Algorithm (GA) was found in the studies for simulating the mechanism of the natural evolution and selection by John Holland. By adopting such concepts borrowed from nature, GAs are able to evolve solutions to a large variety of problems. They are not limited by assumptions about search space such as continuity, existence of derivatives etc.

GA starts with an initial set of random solutions called population. Each individual in the population is called a chromosome, representing a solution to the problem. The evolution operation simulates the process of Darwinian evolution to create population from generation to generation by selection, crossover and mutation operations. The success of genetic algorithm is founded in its ability to keep existing parts of solution, which have a positive effect on the outcome [3].

GA, known as a very efficient heuristic algorithm, gives therefore more accurate results than other algorithms in the mix proportioning problem having many local solutions.

1.2.1. Selection

The selection algorithm selects individuals for reproduction on the basis of their relative fitness. Many selection techniques employ a “roulette wheel” mechanism to select individuals by determining the survival probability for each chromosome proportional to the fitness value. For example in Fig.1 the circumference of the roulette wheel is the sum of all six individual’s fitness values. Individual 5 is the fit individual and occupies the largest interval, whereas individuals 6 and 4 are the least fit and have correspondingly smaller intervals within the roulette wheel. To select an individual, a random number is generated in the interval and the individual whose segment spans the random number is selected. This process is repeated until the desired numbers of individuals have been selected [4].

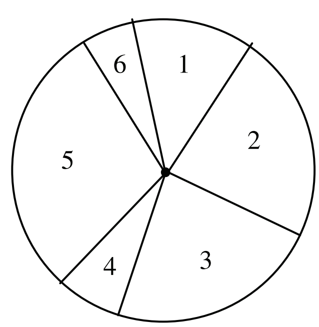

Fig. 1. Roulette wheel selection

1.2.2. Crossover

The basic operator for producing new chromosomes in the GA is crossover. Like its equivalent in nature, crossover produces new individuals that have some parts of both parent's genetic material. The simplest form of crossover is that of single-point crossover. Consider the two parent binary strings:

P1 = 1 0 0 1 0 1 1 0

P2 = 1 0 1 1 1 0 0 0

If an integer position, i, is selected uniformly at random between 1 and the string length, l, minus one [1, l-1], and the genetic information exchanged between the individuals about this point, then two new offspring strings are produced. The two offspring below are produced when the crossover point i = 5 is selected,

O1 = 1 0 0 1 0 0 0 0

O2 = 1 0 1 1 1 1 1 0

For multi-point crossover, m crossover positions, $k_i \in \{1, 2, ..., l-1\}$ where k_i are the crossover points and l is the length of the chromosome, are chosen at random with no duplicates and sorted into ascending order. Then, the bits between successive crossover points are exchanged between the two parents to produce two new offspring. This process is illustrated in Fig. 2.

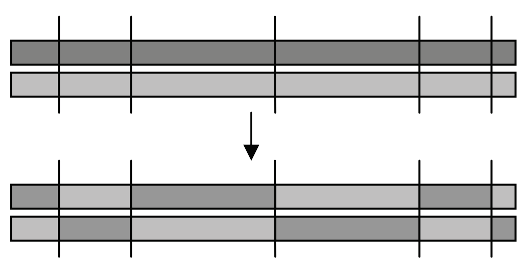

Fig. 2. Multi-point crossover (m=5)

1.2.3. Mutation

In natural evolution, mutation is a random process where one allele of a gene is replaced by another to produce a new genetic structure. In GAs, mutation is randomly applied with low probability, typically in the range 0.001 and 0.01, and modifies elements in the chromosomes. Usually considered as a background operator, the role of mutation is often seen as providing a guarantee that the probability of searching any given string will never be zero and acting as a safety net to recover good genetic material that may be lost through the action of selection and crossover. The effect of mutation on a binary string is illustrated in Fig. 3 for a 10-bit chromosome. Given that mutation is generally applied uniformly to an entire population of strings, it is possible that a given binary string may be mutated at more than one point [4].

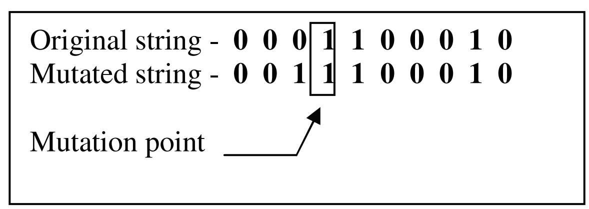

Fig. 3 Binary mutation

After creating initial population composed of strings substituting for mix proportions, fitness increases through the repeating process of selection, crossover and mutation. When fitness is satisfied, the repeating process is terminated and optimal solution is approached. This process is illustrated in Fig. 4.

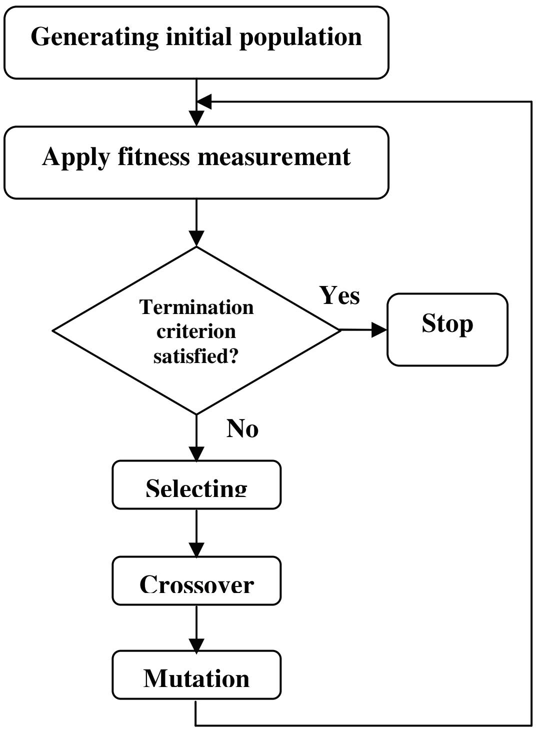

Fig. 4. Outline of genetic algorithm

1.3.Particle swarm optimization

Particle swarm optimization (PSO) is a population based stochastic optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995, inspired by social behaviour of bird flocking or fish schooling.

PSO shares many similarities with evolutionary computation techniques such as GA. The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. In PSO, the potential solutions, called particles, fly through the problem space by following the current optimum particles. Compared to GA, the advantages of PSO are that PSO is easy to implement and there are few parameters to adjust. PSO has been successfully applied in many areas: function optimization, artificial neural network training, fuzzy system control, and other areas where GA can be applied.

1.3.1. The algorithm

As stated before, PSO simulates the behaviors of bird flocking. Suppose the following scenario: a group of birds are randomly searching food in an area. There is only one piece of food in the area being searched. All the birds do not know where the food is. But they know how far the food is in each iteration. So the best strategy to find the food is to follow the bird which is nearest to the food.

In PSO, each single solution is a "bird" in the search space. It’s called "particle". All of particles have fitness values which are evaluated by the fitness function to be optimized, and have velocities which direct the flying of the particles. The particles fly through the problem space by following the current optimum particles.

PSO is initialized with a group of random particles (solutions) and then searches for optima by updating generations. In every iteration, each particle is updated by following two "best" values. The first one is the best solution (fitness) it has achieved so far. (The fitness value is also stored.) This value is called Pbest. Another "best" value that is tracked by the particle swarm optimizer is the best value, obtained so far by any particle in the population. This best value is a global best and called Gbest. When a particle takes part of the population as its topological neighbours, the best value is a local best and is called Lbest. After finding the two best values, the particle updates its velocity and positions with following equations [5].

$$V[i+1] = V[i] + C1 * rand\,(i) * (Pbest[i] - present[i]) + C2 * rand\,(i) * (Gbest[i] - present[i]) \quad (1)$$

$$present[i+1] = persent[i] + V[i] \quad (2)$$

V[] is the particle velocity, present[] is the current particle (solution). Rand() is a random number between (0,1). C1, C2 are learning factors. Usually C1 = C2 = 2.

The code of the procedure is as follows.

For *each particle*
  Initialize particle
End

Do
  For *each particle*
    Calculate fitness value
    If *the fitness value is better than the best Fitness value (Pbest) in history set current value as the new Pbest.*
    End

Choose the particle with the best fitness value of all the particles as the Gbest
  For *each particle*
    Calculate particle velocity according to equation **(I)**
    Update particle position according to equation **(II)**
  End
While *maximum iterations or minimum error criteria is not attained*

After creating initial particles for mix proportion, fitness increases through updating particles position. The process is stopped when fitness value is satisfied and then the optimal answer is approached.

2. **Experimental data**

The 108 sets of experimental mixture test for compressive strength and slump were used. The factors influencing compressive strength tests are W/B: water to blinder ratio (%), W: water content (kg/m3), s/a: fine aggregate ratio (%), FA: fly ash replacement ratio (%), SF: silica fume replacement ratio (%), AE: air-entraining agent content (kg/m3). The factors affecting slump are W/B: water to blinder ratio (%), W: water content (kg/m3), s/a: fine aggregate ratio (%), FA: fly ash replacement ratio (%), SF: silica fume replacement ratio (%), AE: air-entraining agent content (kg/m3) and the content of superplasticizer (SP, kg/m3).

The 104 sets of mixtures for compressive strength between 40-80 MPa, are listed in Table.1. The four sets of mixture were used for verification and validation of procedure.

Table.1
Sets of mixtures (40-80 MPa)

No	f_c (MPa)	Slump (mm)	W/B (%)	W Kg/m^3	s/a (%)	FA (%)	AE Kg/m^3	SP Kg/m^3
1	74	215	30	160	48	10	0.069	8
2	74	245	30	160	48	20	0.069	8
3	71	200	30	160	46	0	0.069	8
4	72	210	30	160	45	10	0.069	8
5	69	205	30	160	44	20	0.069	8
6	69	240	30	160	42	0	0.069	8
7	68	210	30	160	42	10	0.069	8
8	65	225	30	160	41	20	0.069	8
9	66	210	30	170	47	0	0.074	8.5
10	66	260	30	170	46	20	0.074	8.5
11	65	225	30	170	44	0	0.074	8.5
12	65	205	30	170	43	10	0.074	8.5
13	63	200	30	170	42	20	0.074	8.5
14	64	245	30	170	41	0	0.074	8.5
15	63	225	30	170	40	10	0.074	8.5
16	63	260	30	170	39	20	0.074	8.5
17	61	220	30	180	45	0	0.078	7.5
18	62	195	30	180	44	10	0.078	7.5
19	62	250	30	180	44	20	0.078	7.5
20	62	210	30	180	42	0	0.078	7.5
21	61	210	30	180	41	10	0.078	7.5
22	58	200	30	180	40	20	0.078	7.5
23	61	225	30	180	38	0	0.078	7.5
24	61	210	30	180	38	10	0.078	7.5
25	61	240	30	180	37	20	0.078	7.5
26	63	145	35	160	51	0	0.059	5.71
27	63	250	35	160	50	10	0.059	5.71
28	62	240	35	160	50	20	0.059	5.71
29	63	175	35	160	48	0	0.059	5.71
30	63	195	35	160	47	10	0.059	5.71
31	59	245	35	160	47	20	0.059	5.71
32	63	185	35	160	45	0	0.059	5.71
33	62	230	35	160	44	10	0.059	5.71
34	59	240	35	160	43	20	0.059	5.71
35	60	195	35	170	49	0	0.063	4.86
36	58	225	35	170	49	10	0.063	4.86
37	56	200	35	170	48	20	0.063	4.86
38	59	195	35	170	46	0	0.063	4.86
39	58	240	35	170	45	10	0.063	4.86
40	58	225	35	170	45	20	0.063	4.86
41	57	220	35	170	43	0	0.063	4.86
42	55	225	35	170	42	20	0.063	4.86
43	55	195	35	180	48	0	0.067	3.86
44	54	195	35	180	47	10	0.067	3.86
45	52	200	35	180	46	20	0.067	3.86
46	56	150	35	180	44	0	0.067	3.86
47	51	190	35	180	44	10	0.067	3.86
48	48	170	35	180	43	20	0.067	3.86
49	53	190	35	180	41	0	0.067	3.86
50	46	220	35	180	40	10	0.067	5.14
51	48	210	35	180	40	20	0.067	5.14
52	51	170	40	160	52	0	0.04	4
53	49	95	40	160	52	10	0.04	2.57
54	49	220	40	160	51	20	0.04	4
55	50	210	40	160	49	0	0.04	4

56	49	205	40	160	49	10	0.04	4
57	49	220	40	160	48	20	0.04	4
58	50	230	40	160	46	0	0.04	4
59	49	195	40	160	46	10	0.04	4
60	47	210	40	160	45	20	0.04	4
61	49	205	40	170	51	0	0.043	2.13
62	48	195	40	170	50	10	0.043	2.13
63	46	175	40	170	50	20	0.043	2.13
64	47	190	40	170	48	0	0.043	2.13
65	47	195	40	170	47	10	0.043	2.13
66	46	195	40	170	47	20	0.043	2.13
67	47	170	40	170	45	0	0.043	2.13
68	46	200	40	170	44	10	0.043	2.13
69	44	180	40	170	44	20	0.043	2.13
70	45	210	40	180	49	0	0.045	2.25
71	44	205	40	180	49	10	0.045	2.25
72	43	205	40	180	48	20	0.045	2.25
73	45	210	40	180	46	0	0.045	2.25
74	44	200	40	180	46	10	0.045	2.25
75	44	210	40	180	45	20	0.045	2.25
76	44	220	40	180	43	0	0.045	2.25
77	42	195	40	180	42	10	0.045	2.25
78	43	220	40	180	42	20	0.045	2.25
79	47	180	45	160	53	0	0.036	3.56
80	46	140	45	160	53	10	0.036	3.56
81	45	130	45	160	52	20	0.036	3.56
82	45	160	45	160	50	0	0.036	3.56
83	43	160	45	160	50	10	0.036	3.56
84	45	170	45	160	49	20	0.036	3.56
85	44	120	45	160	47	0	0.036	3.56
86	43	160	45	160	47	10	0.036	3.56
87	44	200	45	160	46	20	0.036	3.56
88	46	175	45	170	52	0	0.038	1.89
89	42	130	45	170	51	10	0.038	1.89
90	42	100	45	170	51	20	0.038	1.89
91	43	190	45	170	49	0	0.038	1.89
92	42	165	45	170	48	10	0.038	1.89
93	42	190	45	170	48	20	0.038	1.89
94	43	200	45	170	46	0	0.038	1.89
95	42	185	45	170	45	10	0.038	1.89
96	42	180	45	170	45	20	0.038	1.89
97	42	230	45	180	51	0	0.04	2
98	42	210	45	180	50	10	0.04	2
99	41	175	45	180	50	20	0.04	2
100	42	170	45	180	47	0	0.04	2
101	41	185	45	180	47	20	0.04	2
102	43	175	45	180	44	0	0.04	2
103	40	220	45	180	44	10	0.04	2
104	38	170	45	180	43	20	0.04	2

2.1 Mix proportions ranges

According to the Table.1 mix proportion ranges to obtain compressive strength between 40 and 80 MPa are, the W/B varies between 30% and 40%, the water content between 160-180 kg/m3, the fine aggregate ratio is 38-53%, the amount of fly ash used varies from 0% to 20% and the content of superplasticizer and air-entraining agent are 1.89-8 kg/m3 and 0.036-0.078 kg/m3 respectively.

2.2. Compressive strength test

Specimens for this test were made in 100×100 mm cylinder molds. After curing the specimens in water at 20±3 oC for 28 days, the tests were executed accordance with ASTM C 684-95 standard.

2.3. Slump test

The slump tests were determined immediately after finishing of mixing according to the ASTM C 143-90a.

3. Fitness function

To determine the fitness function for compressive strength and slump with 104 experimental tests, multiple regression modeling was applied. For a function with n independent variables as inputs and one dependent variable as output, the least square problem is used to find out the unknown parameters of linear model as shown in Eq.3 [6].

$$f = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + ... + \alpha_n x_n \qquad (3)$$

In this study MATLAB version 6.5 was used to determine the unknown parameters α of model. Table.2 shows the fitness function of compressive strength and slump, together with short definition of each independent variable.

Table.2

Fitness function

Compressive strength	f=121.65-0.819 W/B-0.416 W +0.33 s/a-0.081 FA+336.87 AE
Slump	Slump=-452-3.12W/B+5.11W+ 0.53s/a+0.9FA+6423.3AE+39.5SP
f: compressive strength (MPa),W/B: water to binder ratio (%),W: water content (kg/m3), s/a: fine aggregate ratio (%), FA: fly ash replacement ratio (%), SF: silica fume replacement, AE: air-entraining agent content (kg/m3), SP: superplasticizer content (kg/m3).	

To verify compatibility of regression model, coefficient of determination (R2), must be over 70% and the change of (R2), to the change of the number of data needs to be observed. As shown in Table.3 it was concluded the compatibility of model.

3. Table

The coefficient of determination (R2)

Number of data	104	95	85	75	65
Compressive Strength R2 (%)	95.44	95.32	95.3	95.1	94.7
Slump R2 (%)	75.6	75.6	75.5	75.4	75.4

4. Application of genetic algorithm

The genetic algorithm program was developed to find the high-performance concrete proportion mixture by using MATLAB version 6.5. In this program the inputs and outputs are compressive strength and slump values. Through selection, crossover and mutation operations on population composed of W/B, W, s/a, FA, SF, the fitness function increases. When the error between input and output become minimize the fitness is satisfied, the program is terminated and optimal solution is approached. By using the optimal mixture, SP for specific slump is determined.

In this search linear ranking was used. Selection was performed based on stochastic universal sampling. Crossover was applied based on single-point and multi-point method with probability of 0.7 and the mutation probability value was 0.7/Lind, where Lind is the length of an individual. The number of initial individuals was 10.

To verify the accuracy and usefulness of these procedures four sets of mixture tests listed in Table.4 were compared to the results from genetic algorithm program. Table.5 shows the results from genetic algorithm and error. The convergence of output to the specific fitness value of input is shown in Fig.5.

Table.4
Tested Mixtures for verification

No	f (MPa)	Slump (mm)	W/B	W	s/a	FA	AE	SP
1	48	210	35	180	40	20	0.067	5.14
2	57	230	35	170	42	10	0.063	4.86
3	66	195	30	170	46	10	0.074	8.50
4	75	205	30	160	49	0	0.069	8.00

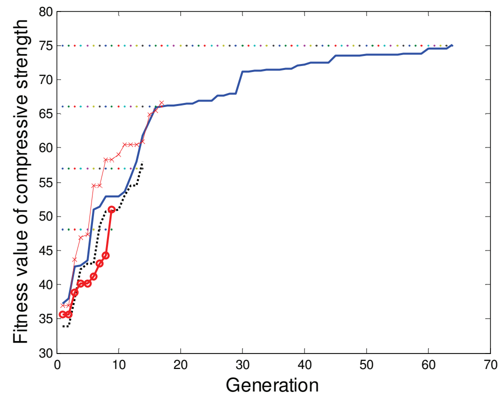

Fig. 5. Fitness values of outputs for 4 sets of mixture tests

Table.5
The results from genetic algorithm and error

No	W/B	W	s/a	FA	AE	SP	Average Error
1	41	179	45	17	0.061	5.72	
Error	17%	0.5%	12.5%	15%	8.9%	11.28%	10.8%
2	38	164	45	11	0.062	5.83	
Error	7.8%	3.5%	7.1%	10%	1.5%	19.9%	8.3%
3	33	175	47	8.6	0.072	7.00	
Error	10%	2.9%	2.17%	14%	2.7%	17.2%	8.1%
4	31	160	52	0	0.077	9.7	
Error	3.3%	0%	6.1%	0%	11.5%	21.2%	7.0%

5. Application of particle swarm optimization

In applying PSO to mix proportion problem of HPC, each particle represents a mixture proportion. The particles including W/B, W, s/a, FA, SF fly through the problem space and fitness function increases. When the fitness function of compressive strength reaches to the value of input, the process is terminated and SP for specific slump is determined.

MATLAB version 6.5 was used to implement this program. Learning factors with values of 1 were used and the number of particles was 10. The convergence of output to the specific fitness value of input is shown in Fig.6.

To verify the accuracy and usefulness of these program four sets of mixture tests listed in Table.4, were compared to the results from PSO. The results and error are shown in Table.6.

Table.6
The results from particle swarm optimization

No	W/B	W	s/a	FA	AE	SP	Average Error
1	39	172	42	15	0.058	6.02	
Error	11.4%	4.4%	5%	25%	13.4%	17%	12.7%
2	34	165	40	13	0.060	5.36	
Error	2.8%	2.9%	5%	30%	4.7%	10.3%	9.26%
3	32	163	49	8.3	0.065	7.3	
Error	6.6%	4.1%	6.52%	10.7%	12.1%	14.1%	8.97%
4	30	160	51	1.6	0.076	9.49	
Error	0%	0%	4.1%	16%	10.1%	18.6%	8.13%

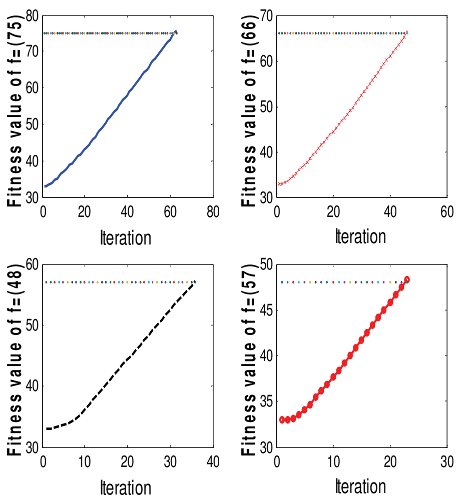

Fig.6. Fitness values of output for 4 sets of mixture tests

6. Conclusions

This research represents the using of genetic algorithm and particle swarm optimization as an evolutionary searching strategy to find the mix proportion of high-performance concrete to minimize number of trials mixes to provide a reasonable mix proportion with specific properties. The experimental data was used to develop fitness function and verification of procedures. The main results from this study can be summarized as follows.

By applying genetic algorithm and particle swarm optimization the number of trail mixtures with desired properties can be reduced.

It is believed that the error of mix proportion, calculated by GA and PSO decreases by increasing the desired value of compressive strength fitness function.

As shown in verification tables some factors such as W/B and s/a show less errors in implementing by PSO in comparison with GA, whereas for W and SF, GA shows better results.

The factors with decimal and low values show large errors. Because changing on a very small scale, the errors tend to have large values. Therefore the errors of these factors are considered to be not so large.

References

[1] A.M. Neville, P.C Aitcin, *High-performance concrete- An overview*, 1998.
[2] I. Maruyama, M. Kanematsu, "Optimization of Mix Proportion of Concrete under Various Severe Conditions by Applying the Genetic Algorithm", University of Tokyo, Japan, 2004.
[3] D. Goldberg, *Genetic Algorithm in search, optimization and Machine Learning*, Addison Welsley publishing company, 1989.
[4] A. Chipperfield, *Genetic algorithm user's guide for use with MATLAB*, Version 1.2.
[5] J. Singh, PSO MATLAB Toolbox, PSOTOOLBOX Open Source.2003.
[6] MATLAB, Version 6.5, The Math Works, 2002.