Seventh International Conference on Computing in Civil and Building Engineering
19-21 August 1997, Seoul, Korea

The Proportioning of Concrete Mixture Using Artificial Neural Networks

*Ju-Won Oh[1), Gyu-Won Lee[2) and In-Won Lee[3)

Dept. of Civil and Environmental Engineering, Hannam Univ., 306-791, Taejon, Korea[1)
Dept. of Civil Engineering, Chonbuk National Univ., Chonbuk, 561-756, Korea[2)
Dept. of Civil Engineering, KAIST, Taejon, 305-701, Korea[3)

ABSTRACT

In determining proportioning of concrete mixtures we meet the uncertainties of materials, temperature, site environmental situations, personal skillfulness, and errors in calculations and testing process. Then the adjustments must be made for a proper proportioning. This kind of concrete mix design and adjustments are somewhat complicated, time-consuming, and uncertain tasks. In this paper, as a tool to minimize the uncertainties and errors, the neural network is applied to the proportioning of concrete mixtures. Not only the numerical required compressive strengths but also the actual compressive strengths with variations obtained from the final compressive strength test is used to train and test the networks. The results show that neural networks have a strong potential as a tool for concrete mix design.

INTRODUCTION

Many obscure factors influence the concrete mix design and their mutual relationship are so complicated that it is impossible to formulate mathematical models to express their mutual actions and reactions. Adjustments are always performed by taking into account the information from the concrete quality control tests and other available researches, and the expert's advice and experience. But the final mix proportions through the adjustments is still a statistic containing the uncertainties and the various errors.

Since a reasonable concrete mix design and adjustment is somewhat complicated, time-consuming, and tedious tasks so that is easy to be neglected and also it is not always possible to be helped by the experts, there are some efforts to develop an expert system for concrete mix design, including the adjustment, as a valuable tool. In this expert system they tried to help the user in concrete mix design and adjustments (Bay 1994; Celik et al 1989; Clifton et al 1988).

---

1), 2), 3) Professors

But it is still impossible to avoid the personal, physical, and mechanical errors and uncertainties which are encountered in materials, testing en- vironments, constructing environments, and other factors such as transporting, delay in placing, and weather conditions. In this paper the neural network is applied to the concrete mix design as a tool to minimize the uncertainties and the errors which are unavoidable in concrete mix design and adjustments.

NEURAL NETWORKS

An artificial neural network, in short a neural network, inspired by a neuronal structure and operation of the biological brain appeared for the first time in the 40s (McCulloch and Pitts 1943). But only in the past few years neural networks have emerged as a new practical alternative to do mainly with pattern recognition in many fields such as biological, business, environmental, financial, manu- facturing, medical, and military fields (Nelson et al. 1991)

In civil engineering neural networks are applied to the detection of structural damage (X. Wu et al. 1992), structural system identification (Chen et al. 1992; Yokosuka et al. 1994), modeling of material behavior (Ghaboussi et al. 1991), structural opti-mization (Adeli et al. 1995), structural control (Chen et al. 1995), groundwater monitoring (Ranjithan et al. 1993), and now concrete mix design in the present paper.

Learning of neural networks usually means modifying connection weights by means of a learning rule (Rosenblatt 1962). In other words, neural networks learn from examples and exhibit some capability for generalization beyond the training data. Then another testing data are used for checking the generalization. there are many learning methods for neural networks by now.

Many of these learning methods are closely connected with a certain network topology. Well-known examples in unsupervised learning are; the Hopfield network (Hopfield 1982) and the competitive learning (Grossberg 1976; Rumelhart and Zipser 1985) for feed-back nets, the fuzzy associative memory and the counterpropagation for feed forward-only nets. In supervised learning; the Boltzmann Machine and fuzzy cognitive map for feed-back nets and, the back-propagation and the perceptron for feed forward-only nets. In the present paper the back-propagation network (Rumelhart et al, 1986) is used for a concrete mix design.

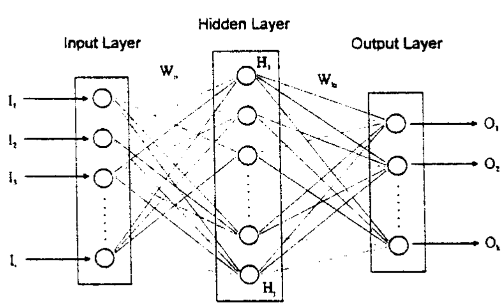

Fig. 1. Simple Neural Network

Fig. 1 shows a simple architecture of a back-propagation network which consists of an input layer, a hidden layer, an output layer, and connections between them. The learning mechanism of this back propagation network is a generalized delta rule which performs a gradient-descent on the error space to minimize fast the total error between the calculated values and the designed ones of the output layer during modification of connection strengths like exercises in nonlinear optimization

Though the training data contain the various errors and uncertainties, it is no great matter to train and to generalize the examples. As a matter of fact, it is reasonableto contain the errors and the uncertainties in training data.Training is accomplished in an iterative process. Each iteration cycle, called an epoch, involved a feed-forward computation followed by an error-backward propagation to modify their connection weights.

Convergence depends on the number of hidden layer neurons, size of the learning rate parameters, and amount of data necessary to create the proper results. The back-propagation neural network is not guaranteed to get a global optimum, but only a local error minimum.

CONCRETE MIX DESIGN

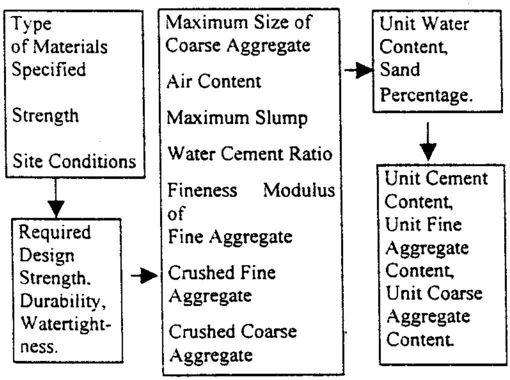

Fig. 2. Process of Concrete Mix

For the application of the neural networks to the concrete mix design, finally we have to use the data tested on practical sites. But for this, much time and many experiments are needed. In this paper the analytical data are used instead of the practical and experimental data, because there are no differences in results for applicability. The process of concrete mix design is shown in Fig. 2. In the present paper it is assumed that normal portlant cement is used, gravity of cement is 3.16, gravity of fine aggregate 2.60, gravity of coarse aggregate 2.65, and air-entrained 4%.

Here computer program was written to obtain the unit weights of components of concrete. The water-cement ratio, sand percentage, unit water 2, 180 data sets out of 192 data sets previously obtained are used for training and 12 data sets for testing. Only error goal is different each other in order to compare the tendency of computation and convergence. Required compressive strengths for input are the same as one assumed in analytical design. When we use the data sets obtained from the practical sites the compressive strengths in 28-days are different from the required compressive strengths and distribute to make a normal curve as a statistic. content, unit cement content, unit fine aggregate content, and unit coarse aggregate content are calculated from the program on the assumption that the various values of specified compressive strengths, maximum size of aggregate, slump, and fineness modules are given as following.

· compressive strength(kg/cm^2) : 180, 210, 240, 270, 300, 330
·max. size of aggregate(mm) : 15, 20, 25, 40
· slump(cm) : 5, 7.5, 10, 12.5
· fineness modules : 2.5, 3.0

As the result of combination of the given data shown above, it was possible to get 192 data sets of which some are shown in table 1. From these 192 data sets, 12 data sets shown in table 2. are selected aside for testing, and 180 data sets are used for training in the back-propagation neural networks.

DESIGN, TRAINING, AND TESTING OF NEURAL NETWORKS.

The purpose of the neural networks applied tothe concrete mix design is to decide the unit contents of the ingredients of concrete such as cement, water, fine aggregate, and coarse aggregate when the various design parameters are known. The various design parameters would be the input neurons, and the unit contents of the ingredients of concrete would be the output neurons. The architecture of neural networks was designed as back-propagation networks with four output neurons for cement contents, water contents, fine aggregate contents and coarse aggregate contents; and10 neurons in a hidden layer; and six input neurons for required compressive strength, maximum size of coarse aggregate, maximum slump, fineness modules, water-cement ratio and sand percentage. The number of hidden neurons was chosen based on trial and error.

The training and testing of this neural networks was performed for four cases as shown in table 3. During the training, the network updates its connection weights and biases until the summation of squared errors(SSE) is less than the specified error goal, and then the training is finished. In cases 1 and 2, 180 data sets out of 192 data sets previously obtained are used for training and 12 data sets for testing. Only error goal is different each other in order to compare the tendency of computation and convergence. Required compressive strengths for input are the same as one assumed in analytical design. When we use the data sets obtained from the practical sites the compressive strengths in 28-days are different from the required compressive strengths and distribute to make a normal curve as a statistic.

Table 1. Some Samples of Concrete Mix Design

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
180	15	5.0	4.0	2.5	63.	49.	193	302	864	898
210	20	5.0	4.0	2.5	57.	44.	179	315	786	100
240	25	5.0	4.0	2.5	51.	40.	167	323	729	108
270	40	5.0	4.0	2.5	46.	36.	157	338	655	117
300	15	7.5	4.0	2.5	42.	45.	192	456	732	906
330	20	7.5	4.0	2.5	38.	40.	179	466	670	995
180	40	7.5	4.0	2.5	63.	39.	167	262	731	113
210	15	10.	4.0	2.5	57.	48.	203	355	808	886
240	20	10.	4.0	2.5	51.	43.	189	366	738	984
270	25	10.	4.0	2.5	46.	39.	176	377	684	106
300	40	10.	4.0	2.5	42.	35.	166	394	614	114
330	14	12.	4.0	2.5	38.	44.	203	527	682	869
180	20	5.0	4.0	3.0	68.	48.	185	290	857	936
210	25	5.0	4.0	3.0	57.	44.	172	302	795	102
240	40	5.0	4.0	3.0	51.	39.	162	315	720	111
270	15	7.5	4.0	3.0	46.	48.	197	424	793	857
300	20	7.5	4.0	3.0	42.	43.	184	437	728	947
330	20	10.	4.0	3.0	38.	43.	189	490	692	927

(7) : sand percentage(%)
(8) : unit water content(kg/m^3)
(9) : unit cement content(kg/m^3)

Table 2. 12 Data Sets for Testing

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)
330	15	5.0	4.0	2.5	38.	44.	185	481	719	916
300	40	5.0	4.0	2.5	42.	35.	156	370	631	117
270	25	7.5	4.0	2.5	46.	39.	170	366	693	107
240	20	10.	4.0	2.5	51.	43.	189	366	738	984
210	15	12.	4.0	2.5	57.	48.	209	366	796	873
180	40	12.	4.0	2.5	63.	39.	177	278	716	111
330	20	5.0	4.0	3.0	38.	43.	178	461	715	958
300	15	7.5	4.0	3.0	42.	47.	196	465	765	855
270	25	7.5	4.0	3.0	46.	42.	174	374	730	102
240	20	10.	4.0	3.0	51.	45.	193	373	773	932
210	15	12.	4.0	3.0	57.	50.	212	372	830	824
180	40	12.	4.0	3.0	63.	42.	181	284	755	105

(1) : required strengh(kg/m^3)
(2) : maximum size of coarse aggregate(mm)
(3) : maximum slump(cm)
(4) : air content(%)
(5) : fineness modulus of fine aggregate
(6) : water-cement ratio(%)
(10) : unit fine aggregate content(kg/m^3)
(11) : unit coarse aggregate content(kg/m^3)

Table 3. Cases of Learning and Testing

ases	28-days compressive strength	error goals	No. of learning data set	No. of testing data set
1	Required	0.01	180	12
2	Required	0.005	180	12
3	Actual	0.01	180	12
4	Actual	0.01	372	12

Table 4. Actual Compressive Strengths (kg/cm^2)

Case1,2	180	210	240	270	300	330
Case3,4	170-188	198-219	226-250	254-281	282-312	310-343

Even if all characteristics considered in this network are same, the actual compressive strength varies, because there are other influences not considered in this networks. For this reason, in cases 3 and 4 the actual compressive strengths, the 28-day compressive strengths, are used for training. In fact the actual compressive strength should be obtained from the real compressive tests in practical sites. But in this paper the actual compressive strengths are chosen randomly in the range of less than 10% of the required compressive strengths as shown in table 4. In case 4 another 192 data sets are added and total 372 data sets are used for training to find the influences of increasing the number of training data sets. In all cases the testing data sets are the same as shown in table 2.

Table 5. Convergences in Training

Cases	error goals	sse	epochs
1	0.01	0.0099998	27.264
2	0.005	0.0049999	157.808
3	0.01	0.0099993	234.428
4	0.01	0.0099999	340.375

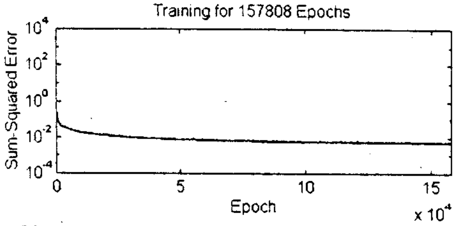

Fig. 3. Convergence curve in training case

Table 5 and Fig. 3 show the results of training. In all cases the neural network applied to concrete mix design was converged well. In general the convergence is very fast in early stage, but gradually becomes slower as the epoch goes ahead. Comparing the cases 1 and 2, the error goal in case 2 is strengthened twice better than the one in case 1, but the epochs increased five times. This means that to get some more precise results, the amount of calculations should be increased with geometric series.

Comparing the cases 1 and 3, training in case 3 using the actual compressive strength obtained from the real compression tests needs many calculations, that is many number of epochs as shown in table 5. Cases 3 and 4 shown in table 5 need more input data and calculations.

The concrete mix design testing results using the trained neural network are listed in tables 6((a)-(d)). In the table 6 the numerical values in ( ) is error percentages(%) which indicates the percentages of differences between the target values and the calculated values with respect to the target values.

Case 1 shows the errors in the range of 0-1.3% within epochs 27,264 with error goals of 0.01, and case 2 shows the errors in the range of 0-1.1% within epochs 157,808 with error goals of 0.005. Comparing this results, any error goals beyond the certain level of error goals are almost of no use to decrease the errors but increase the amount of calculations. So it is necessary to get a proper error goals. In this design the error goal of 0.01 is enough for practical purposes.

Case 3 shows the errors in the range of 0-1.7% and case 4 the errors in the range of 0-1.1%. These results indicate that the neural network trained by the practical data which are distributed with a normal curve can converge well into an allowable error level. Case 3 and 4 also show that the more training data the neural network can be trained by, the more the error level is satisfactory. The allowable error level depends on the error goal, the number of training data, and the architecture of neural networks.

CONCLUSIONS AND REMARKS

The trying and testing results of neural networks applied to concrete mix design look very applicable. The neural networks trained well without any oscillations. And the testing results show that the unit contents of ingredients of concrete can be obtained within an allowable error level which shows maximum error percentage of 1.7% in this paper.

During performing the concrete mix design and adjustments in practical sites, the quality control tests are really necessary. If we make a data base for saving the quality control tests results, we can use it for the concrete mix design using the neural networks. The more the data are available, the better the neural networks will be performed.

REFERENCES

American Concrete Institute (1989). "ACI 211.1-81: standard practice for selecting proportions for normal, heavyweight, and mass concrete," *ACI Manual of Concrete Practice*, Part 1, Detroit, Mich.

Bai, Y., Amirkhanian, S. N. (1994). "Knowledge-Based Expert-System for Concrete Mix Design," *Journal of Construction Engineering and Management*, ASCE, 120(2), 357-373.

Celik, T., Thorpe, A., and McCaffer, R. (1989). "Development of an Expert System," *Concrete Int. Des. and*

Const., 11(8), 37-41.

Clifton, J. R., and Kaetzel. L. J. (1988). "Expert Systems for Concrete Construction," *Concrete Int. Des. and Const.*, 10(11), 19-24.

Elkordy, M. F., Chang, K. C., and Lee, G. C. (1993). "Neural Networks Trained by Analytically Simulated Damage States," *J. of Comput. in Civ. Engrg.*, 7(2), 130-145.

Grossberg, S. (1976). "Adaptive Pattern Classification and Universal Recording: Part 1. Parallel Development and Coding of Natural Features Detectors," *Biolog. Cybernet.*, 23, 121-134.

Ghaboussi, J., Garret, Jr., J. H., and Wu, X. (1991). "Knowledge-Based Modeling of Material Behavior with Neural Networks," *J. Engrg. Mech.*, ASCE, 117(1), 132-153.

Hinton, G. E., Sejnowski, T. J., and Ackley, D. H. (1984). "Boltzman Machines: Constraint Satisfaction Networks That Learn," *Tech. Rep. No. CMU-CS-84-119,* Department of Computer Science, Carnegie Mellon University, Pittsburgh, Pa.

Hopfield, J. J. (1982). "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proc. Nat. Acad. Sci.*, 79, 2554-2558.

Table 6. Calculated Values and Error Percentages

(a) case 1

Testing sets	calculated values			
	water	cement	sand	Gravel
1	187(1.1)	486(1.0)	24(0.7)	904(1.3)
2	156(0.0)	369(0.3)	31(0.0)	1177(0.1)
3	171(0.6)	368(0.5)	85(1.2)	1085(0.6)
4	187(1.1)	364(0.5)	37(0.1)	992(0.8)
5	210(0.5)	370(1.1)	93(0.4)	870(0.3)
6	179(1.1)	278(0.0)	20(0.6)	1102(0.8)
7	176(1.1)	459(0.4)	10(0.7)	969(1.1)
8	197(0.5)	466(0.2)	66(0.1)	852(0.4)
9	174(0.0)	375(0.3)	31(0.1)	1022(0.3)
10	192(0.5)	371(0.5)	79(0.8)	930(0.2)
11	211(0.5)	372(0.0)	23(0.8)	835(1.3)
12	180(0.6)	283(0.4)	51(0.5)	1063(0.7)

(b) case 2

testing sets	calculated values			
	water	cement	sand	Gravel
1	186(0.5)	483(0.4)	717(0.3)	914(0.2)
2	156(0.0)	371(0.3)	629(0.3)	1179(0.1)
3	171(0.6)	368(0.5)	691(0.3)	1078(0.1)
4	188(0.5)	364(0.5)	741(0.4)	985(0.1)
5	208(0.5)	364(0.5)	796(0.0)	875(0.2)
6	177(0.0)	278(0.0)	711(0.7)	1116(0.5)
7	176(1.1)	458(0.7)	714(0.1)	964(0.6)
8	197(0.5)	469(0.9)	761(0.5)	852(0.4)
9	174(0.0)	375(0.3)	729(0.1)	1024(0.1)
10	192(0.5)	371(0.5)	774(0.1)	935(0.3)
11	211(0.5)	372(0.0)	828(0.2)	828(0.5)
12	181(0.0)	284(0.0)	756(0.1)	1055(0.1)

(c) case 3

testing sets	calculated values			
	water	cement	Sand	gravel
1	185(0.0)	486(1.0)	716(0.4)	918(0.2)
2	156(0.0)	365(1.4)	633(0.3)	1180(0.2)
3	172(1.2)	367(0.3)	691(0.3)	1077(0.2)
4	188(0.5)	363(0.8)	736(0.3)	991(0.7)
5	209(0.0)	368(0.5)	800(0.5)	865(0.9)
6	175(1.1)	280(0.7)	715(0.1)	1115(0.4)
7	175(1.7)	456(1.1)	715(0.0)	969(1.1)
8	196(0.0)	469(0.9)	763(0.3)	853(0.2)
9	176(1.1)	376(0.5)	732(0.3)	1017(0.8)
10	191(1.0)	369(1.1)	774(0.1)	938(0.6)
11	212(0.0)	376(1.1)	832(0.2)	819(0.6)
12	178(1.7)	287(1.1)	747(1.1)	1070(1.3)

(d) case 4

Testing	calculated values			
	water	cement	sand	gravel
1	187(1.1)	486(1.0)	718(0.1)	910(0.7)
2	156(0.0)	367(0.8)	629(0.3)	1182(0.3)
3	171(0.6)	368(0.5)	690(0.4)	1080(0.1)
4	188(0.5)	365(0.3)	739(0.1)	987(0.3)
5	209(0.0)	366(0.0)	796(0.0)	872(0.1)
6	179(1.1)	277(0.4)	714(0.3)	1111(0.0)
7	176(1.1)	458(0.7)	713(0.3)	966(0.8)
8	197(0.5)	467(0.4)	764(0.1)	853(0.2)
9	175(0.6)	375(0.3)	730(0.0)	1022(0.3)
10	192(0.5)	370(0.8)	777(0.5)	933(0.1)
11	212(0.0)	372(0.0)	828(0.2)	827(0.4)
12	181(0.0)	285(0.4)	750(0.7)	1061(0.5)