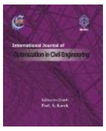
**INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING**
*Int. J. Optim. Civil Eng., 2013; 3(3):389-408*



# USING LATIN HYPERCUBE SAMPLING BASED ON THE ANN-HPSOGA MODEL FOR ESTIMATION OF THE CREATION PROBABILITY OF DAMAGED ZONE AROUND UNDERGROUND SPACES

H. Fattahi[1], S. Shojaee[*, †, 2], M A. Ebrahimi Farsangi[1] and H. Mansouri[1]
[1]Department of Mining Engineering, Shahid Bahonar University of Kerman, Iran
[2]Department of Civil Engineering, Shahid Bahonar University of Kerman, Iran

## ABSTRACT

The excavation damaged zone (EDZ) can be defined as a rock zone where the rock properties and conditions have been changed due to the processes related to an excavation. This zone affects the behavior of rock mass surrounding the construction that reduces the stability and safety factor and increase probability of failure of the structure. In this paper, a methodology was examined for computing the creation probability of damaged zone by Latin hypercube sampling based on a feed-forward artificial neural network (ANN) optimized by hybrid particle swarm optimization and genetic algorithm (HPSOGA). The HPSOGA was carried out to decide the initial weights of the neural network. A case study in a test gallery of the Gotvand dam, Iran was carried out and creation probabilities of 0.191 for highly damaged zone (HDZ) and 0.502 for EDZ were obtained.

Received: 10 February 2013 ; Accepted: 15 July 2013

KEY WORDS: latin hypercube sampling, artificial neural network, particle swarm optimization, genetic algorithm, The creation probability of damaged zone

## 1. INTRODUCTION

The most cost effective method for excavating underground spaces in massive hard rocks, where the uniaxial compressive strength very often exceeds 200 MPa, is drilling and blasting [1]. A very important concern often arises with this method: unwanted damage induced by blasting beyond the desired perimeter of the underground space. The significance and

[*]Corresponding author: S. Shojaee, Department of Civil Engineering, Shahid Bahonar University of Kerman, Iran

[†]E-mail address: saeed.shojaee@uk.ac.ir

importance of this damage have been addressed by various researches [1-4]. Perimeter blasting techniques such as smooth blasting [5] are commonly used to minimize this damage, complemented by theoretical blast damage tables and charts. Although these precautionary measures are taken, blast damage is still inevitable and the conceived consequences are evidenced in the form of increased support cost and requirements, reduction in tunnel life, unforeseen stability problems originating from blast damage, slow tunnel advance, and conduit for water flow.

In the last decades, in different engineering fields, various methods for reliability analysis have been developed to include uncertainties associated with material properties and geometry, loading and boundary conditions. There are three major methods to uncertainty analysis: interval analysis, fuzzy logic and probabilistic analysis (the most well developed methods) [6]. Methods using probability density functions are generally referred to as probabilistic methods. These methods produce the nominal value of the objective functions and constraints as well as their probability density functions. Monte Carlo simulation is the most basic, simplest approach among all probabilistic design methods, but on the other hand it is a very time consuming method [7]. To solve the problem of excessive number of samples required to perform a Monte Carlo simulation, there have been several other simulation techniques developed over the years. In general these techniques can be categorized under the general heading of "Variance Reduction Techniques" and divided into three basic classes: stratified sampling, importance or adaptive sampling and quasi- Monte Carlo simulation [7]. In general, these techniques can often reduce the number of simulations required by several orders of magnitude as compared to basic Monte Carlo simulation. Of the stratified sampling techniques, latin hypercube sampling (LHS) is arguably the most popular version [7]. The LHS is a technique for reducing the number of simulations needed to obtain reasonable results. Essentially, the LHS is the same as the Monte Carlo simulation except that the sampling process is more effective and leading to a better coverage of the sampling space with a smaller number of iterations [8]. In this paper, the LHS method is used to compute reliability of constraints in a reasonable time. By using the LHS, an estimate of the creation probability of damaged zone can be obtained.

Finding the creation probability of damaged zone can be led to a better understanding the risks of a project, a more efficient of establishing geotechnical zoning and the costs can be estimated with more reliability. Furthermore, it can be utilized for optimal designs of support pattern, blast pattern, and excavation method of underground spaces.

Moreover, over the years, the application of artificial neural network (ANN) in geotechnical engineering has been growing. In recent years, there is a growing interest of using ANNs to assist building a reasonable model structure for physical nonlinear systems [9]. ANNs have a special capacity to approximate the dynamics of nonlinear systems in many applications in a black box manner [10]. Given sufficient input-output data, ANN is able to approximate any continuous function to arbitrary accuracy [11]. In addition, several different attempts have been proposed by various researchers to propitiate this training problem. These include imposing constraints on the search space, adjusting training parameters, restarting training at many random points, and restructuring the ANN structure [9]. One of the most promising techniques is by introducing adaptation of network training using hybrid particle swarm optimization and genetic algorithms (HPSOGA). Montana and Davis [12] reported the successful application of a GA to a relatively large ANN problem. They proved that GA produce results superior than back propagation (BP).

In this paper, a new methodology is introduced to determine the creation probability of damaged zone an underground structure by the LHS based on the ANN-HPSOGA model. Based upon the results of plate loading test and using the LHS based on the ANN-HPSOGA model, samples of data were generated. Using the data generated, the creation probability of damaged zone can be estimated. To show the ability of the methodology proposed, field data from a test gallery of the Gotvand dam, Iran were used. According to the authors' knowledge, using the LHS based on the ANN-HPSOGA model for estimation of the creation probability of damaged zone around underground spaces is a unique research.

## 2. A BRIEF REVIEW OF METHODS USED IN THIS STUDY

### *2.1 Latin hypercube sampling*

latin hypercube sampling (LHS) was first proposed by McKay et al. [13] and has been further developed for different purposes by several researchers [14, 15]. The LHS provides a constrained sampling scheme instead of random sampling according to the direct Monte Carlo simulation. A comparison of random sampling with LHS for two variables is shown in Figure 1. In the LHS, the region is uniformly divided into *N* non-overlapping intervals for each random variable; where *N* is the number of random numbers, which need to be generated for each random variable. The *N* non-overlapping intervals are selected to be of the same probability of occurrence. Then, *N* different values in the *N* non-overlapping intervals are randomly selected for each random variable. This can be accomplished by initially generating *N* random numbers. These values represent the percentage position of each generated value of a variable within an interval [16]. Therefore, these values are linearly transformed to the random numbers in the non-overlapping intervals for each random variable using the following equation:

$$u_i = \frac{u}{N} + \frac{(i-1)}{N} \tag{1}$$

where, $i$=1,2,..,n; $u$=a random number; and $u_i$ = random number in the $i^{th}$ interval. From Eq. (1), it is quite obvious that there is only one generated value that is randomly selected within each of the *N* intervals for each random variable. This is due to the following relationship:

$$\frac{(i-1)}{N} \langle u_i \langle \frac{1}{N} \tag{2}$$

where, $(i - 1)/N$ and $i/N$ are lower and upper bounds for the $i^{th}$ interval. Then $n$ values obtained for $x_1$ (the first random variable) are paired in a random manner (equally likely combination) with the $n$ values of $x_2$ (the second random variable). These $n$ pairs of $(x_1,x_2)$ are combined in a random manner with the $n$ values of $x_3$ (the third random variable) to form the first $n$-triplets $(x_1,x_2, x_3)$, and so on, until the $(k\text{-}2)$th $n$-triplets $(x_1,x_2, x_{n-2}, x_{n-1}, x_n)$ ($k$ = number of random variables) are formed. Thus, an $N \times k$ matrix is formed [16].

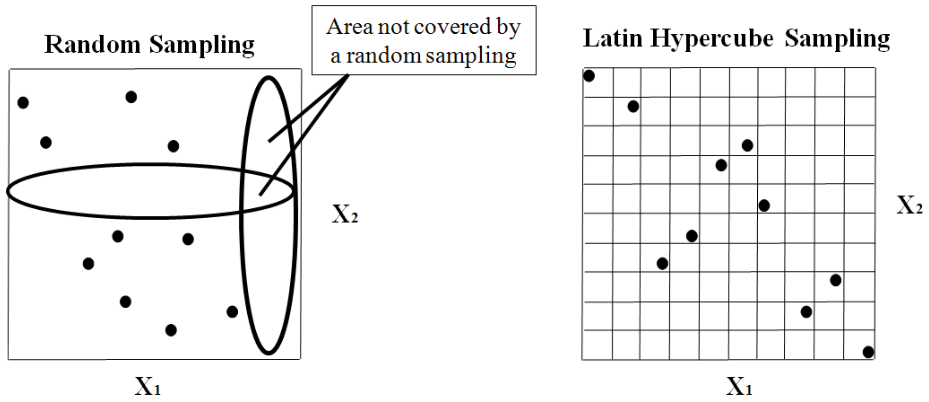


Figure 1. A Comparison of random sampling with the LHS for two variables

About the LHS, it can be said that a reliability problem is normally formulated, using a failure function, $g(x_1, x_2, ..., x_n)$ where, $x_1, x_2, ..., x_n$ are random variables. Violation of the limit state is defined by the condition $g(x_1, x_2, ..., x_n) \leq 0$ and the probability of damage, $p_f$ , is expressed by the following expression [17]:

$$\begin{aligned} p_f &= P\left[g(x_1, x_2, ..., x_n) \leq 0\right] \\ &= \iint\limits_{g(x_1, x_2, ..., x_n) \leq 0} \cdots \int f_{x_1, x_2, ..., x_n}(x_1, x_2, ..., x_n) dx_1 dx_2 .. dx_n \end{aligned} \tag{3}$$

where, $(x_1, x_2, ..., x_n)$ are values of the random variables and $f_{x_1, x_2, ..., x_n}(x_1, x_2, ..., x_n)$ is the joint probability density function. The limit state function, also called performance function, define the boundary between the safe and failure regions in the design parameter space. This function plays an important role in the development of reliability analysis methods. Figure 2 shows the concept of limit state function.

The LHS allows the determination of an estimate of the probability of damage, given by:

$$\bar{p}_f = \frac{1}{N} \sum_{i=1}^{N} I(x_1, x_2, ..., x_n) \tag{4}$$

where, $I(x_1, x_2, ..., x_n)$ is a function defined by:

$$I(x_1, x_2, ..., x_n) = \begin{cases} 1 & if \;\; g(x_1, x_2, ..., x_n) \leq 0 \\ 0 & if \;\; g(x_1, x_2, ..., x_n) \rangle \; 0 \end{cases} \tag{5}$$

According to Eq. (4), $N$ independent sets of values $x_1, x_2, ..., x_n$ are obtained based on the probability distribution for each random variable and the failure function is computed for each sample. Using the LHS, an estimate of the probability of structural failure is obtained by:

$$\bar{p}_f = \frac{N_f}{N} \tag{6}$$

where, $N$ is the total number of samples, and $N_f$ is the number of samples locating at the failure region where, $g(x_1, x_2, ..., x_n) \le 0$.

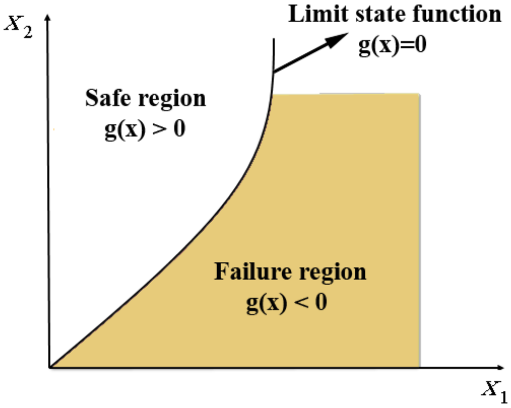


Figure 2. The concept of limit state function

### *2.2 Artificial neural network*

Artificial neural networks (ANNs) are parallel information processing methods that can express complex and nonlinear relationship use, number of input-output training patterns from the experimental data. ANNs provide a nonlinear mapping between inputs and outputs by its intrinsic ability [18]. The success in obtaining a reliable and robust network strongly depends on the correct data preprocessing, correct network training choice, and correct architecture selection [19]. The most common neural network architecture is the feed-forward neural network. Feed-forward neural network is the network structure in which the information or signals will propagates only in one direction, from input to output [18, 20]. The network is trained by performing optimization of weights for each node interconnection and bias terms, until the values output at the output layer neurons are as close as possible to the actual outputs [21].

The data are split into two sets, a training data set and a validating data set. The model is produced using only the training data. The validating data are used to estimate the accuracy of the model performance. In training a network, the objective is to find an optimum set of weights [21]. When the number of weights is higher than the number of available data, the error in-fitting the non trained data initially decreases but then increases as the network becomes over-trained. In contrast, when the number of weights is smaller than the number of data, the over-fitting problem is not crucial [21].

In the last years, ANN technology, a sub-field of artificial intelligence, are being used to solve a wide variety of problems [22-26].

### *2.3 Genetic algorithm*

The genetic algorithm (GA) is a frequently and well-known used evolutionary computation technique. This method was originally developed by John Holland [27] and Hassan et al. [28]. The idea was inspired from Darwin's natural selection theorem that is based on the idea of the survival of the fittest. It uses the principles of genetics and evolution and mimics the reproduction behavior observed in biological populations. In the GA, a candidate solution for a particular problem is called an individual or a chromosome and consists of a linear list of genes. The search in the GA begins from a randomly generated population of designs that evolve over successive generations (iterations), eliminating the need for a user supplied starting point [28]. To perform its optimization like process, the GA employs three operators to propagate its population from one generation to another. The first operator is the "selection" operator in which the GA takes into account the principal of "survival of the fittest" to select and generate individuals (design solutions) that are adapted to their environment. The second operator is the "crossover" operator, which mimics mating in biological populations. The crossover operator propagates features of good surviving designs from the current population into the future population, which will have a better fitness value on average. The last operator is "mutation", which promotes diversity in population characteristics. The mutation operator allows for global search of the design space and prevents the algorithm from getting trapped in local minima [28].

### *2.4 Particle swarm optimization*

The particle swarm optimization (PSO) is one of the recent evolutionary optimization methods. This technique was originally presented by Kennedy and Eberhart [29] in order to solve problems with continuous search space. The PSO is based on the metaphor of communication and social interaction, such as fish schooling and bird flocking. The PSO is similar to the GA in many common points. It performs the search using a population of particles that correspond to individuals in the GA. Both algorithms start with a randomly generated population. The PSO does not have a direct recombination operator. However, the stochastic acceleration of a particle toward its previous best position, as well as toward the best particle of the swarm, resembles the recombination procedure in evolutionary computation [30]. In comparison to the GA, the PSO has some attractive characteristics. It has memory, thus the knowledge of good solutions is retained by all particles, whereas in the GA, previous knowledge of the problem is destroyed once the population changes. The PSO does not use the filtering operation (such as selection in the GAs), and all the members of the population are maintained through the search procedure to share their information effectively. The PSO uses social rules to search in the design space by controlling the trajectories of a set of independent particles. The position of each particle, $x_i$, representing a particular solution of the problem, is used to compute the value of the fitness function to be optimized. Each particle may change its position and consequently may explore the solution space, simply varying its associated velocity. In fact, the main the PSO operator is the velocity update, which considers the best position, in terms of fitness value reached by all the particles during their paths, $P_g^t$, and the best position that the agent itself has reached during its search, $P_i^t$, resulting in a migration of the entire swarm toward the global optimum [31].

At each iteration the particle moves around according to its velocity and position; the cost function to be optimized is evaluated for each particle in order to rank the current location. The position of each particle is updated using its velocity vector as shown in Eq. (8) and depicted in Figure 3.

$$V_i^{t+1} = \omega V_i^t + C_1 r_1^t (P_i^t - X_i^t) + C_2 r_2^t (P_g^t - X_i^t) \tag{7}$$

$$X_i^{t+1} = X_i^t + V_i^{t+1} \tag{8}$$

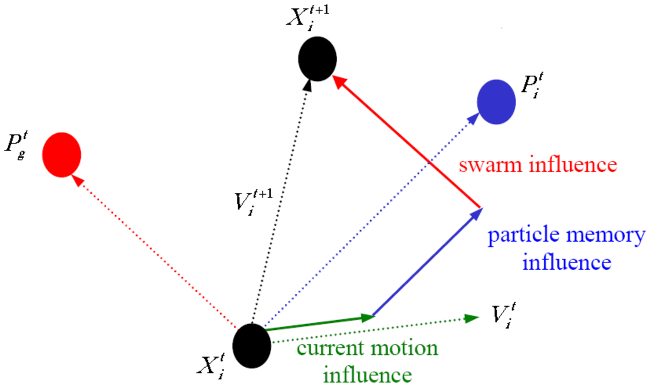


Figure 3. Depiction of the velocity and position updates in PSO (after [28])

where, $V_i^t$ is the velocity vector at iteration *t*, $r_1$ and $r_2$ represents random numbers in the range [0,1]; $P_g^t$ denotes the best ever particle position of particle *i*, and $P_i^t$ corresponds to the global best position in the swarm up to iteration *t* [30]. The remaining terms are problem-dependent parameters; for example, $C_1$ and $C_2$ represent "trust" parameters indicating how much confidence the current particle has in itself ($C_1$: cognitive parameter) and how much confidence it has in the swarm ($C_2$: social parameter), and ω is the inertia weight. The latter term plays an important role in the PSO convergence behavior since it is employed to control the exploration abilities of the swarm. It directly influences the current velocity, which in turn is based on the previous history of velocities. Large inertia weights allow for wide velocity updates providing the global exploration of the search space, while small inertia values concentrate the velocity updates to nearby regions of the design space [32].

### *2.5 Hybrid genetic algorithm and particle swarm optimization*

Although the GAs have been successfully applied to a wide range of problems, using the GAs for large-scale optimization could be very expensive due to its requirement of a large number of function evaluations for convergence. This would result in a prohibitive cost for computation of function evaluations even with the best computational facilities available today [33]. Considering the efficiency of the PSO and the compensatory property of the GA and the PSO, combining the searching abilities of both methods in one algorithm seems to be a logical approach. In this paper, the hybrid of the GA and the PSO named the HPSOGA, originally presented by Juang [34], was used. The flowchart of the HPSOGA is shown in Figure 4.

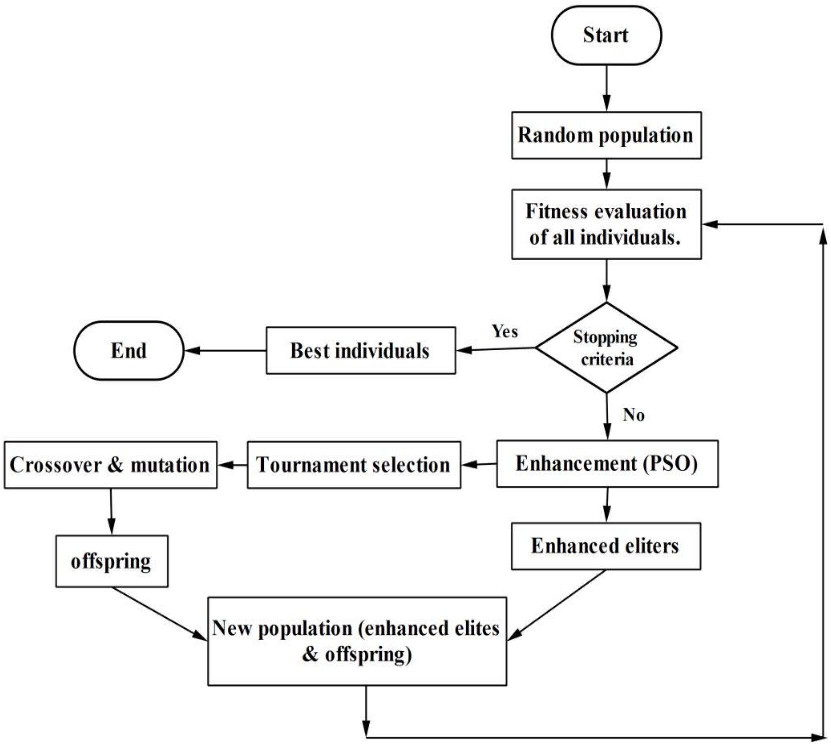


Figure 4. Flowchart of the HPSOGA

## 3. EXCAVATION DAMAGE ZONE

Excavation of an underground construction by drilling and blasting creates a zone of damaged rock around the structure. This zone affects the behavior of rock mass surrounding the construction (Figure 5), that reduces the stability and safety factor and increase probability of failure of the structure. Different definitions for the damaged or disturbed zone have been used. In this paper, the definitions of Tsang et al. [35] for excavation disturbed zone (EdZ), excavation damaged zone (EDZ) and highly damaged zone (HDZ) are adopted (Figure 6).

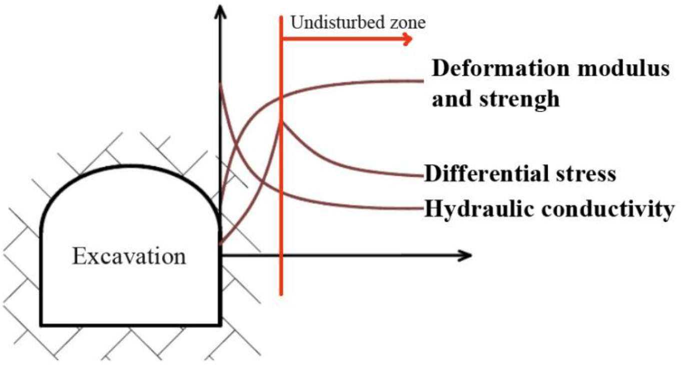


Figure 5. Behavior of rock mass surrounding an underground construction

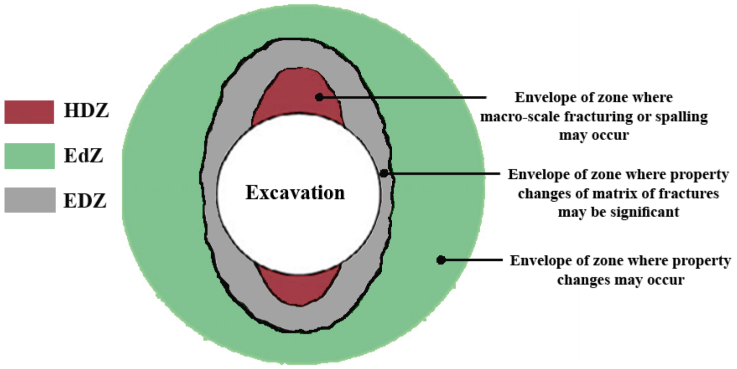

HDZ
EdZ
EDZ
Excavation
Envelope of zone where macro-scale fracturing or spalling may occur
Envelope of zone where property changes of matrix of fractures may be significant
Envelope of zone where property changes may occur

Figure 6. Zones around an underground construction

## 4. SITE DESCRIPTIONS AND GEOLOGY OF CASE STUDY

The Gotvand dam is located on the Karun river in the Khuzestan province, south west of Iran (Figure 7). This dam with 178 m height and 730 m length of embankment, regulates the water of the Karun river, also serves power generation, flood control and irrigation needs.

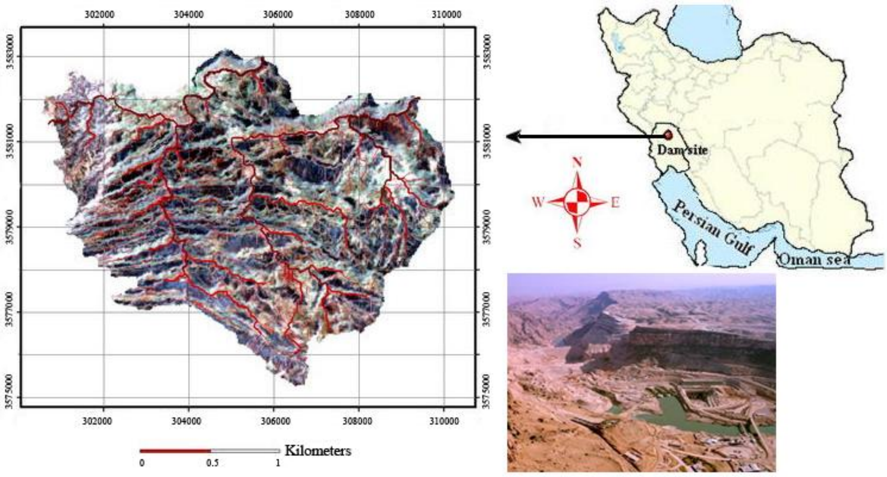


Figure 7. Location of Gotvand dam

The geology of area is mainly including two formations; Bakhtiary (BK) and Aghajari (AJ). The BK formation consists of conglomerate, cherty limestone and inter bedded mudstones and sandstone. The AJ formation contains 2 to 5 m thick layers of gray and greenish gray sandstones, inter bedded claystone, siltstone and brow reddish marlstone.

### *4.1 Determination of deformation modulus by plate loading test*

The creation of EDZ due to a blasting impact and stress redistribution after excavation causes significant changes on the mechanical and physical properties and hydraulic conductivity around an underground excavation. The modulus of deformation is an important parameter among geomechanical parameters that represents the behavior of rock mass after excavation, which can be used for the assessment of EDZ.

The plate loading test (PLT) is the most familiar in situ experiment in rock mass studying. It is generally performed in special test galleries or underground spaces excavated by conventional drill and blast, having a span of 2 m and a height of 2.5 m [36]. In the PLT, load is directly imposed on the wall of gallery, and the resultant displacement is measured on the loading point in rock. A cycle of loading and unloading (Figure 8) provides the load-displacement curve, which is necessary to determine deformation modulus.

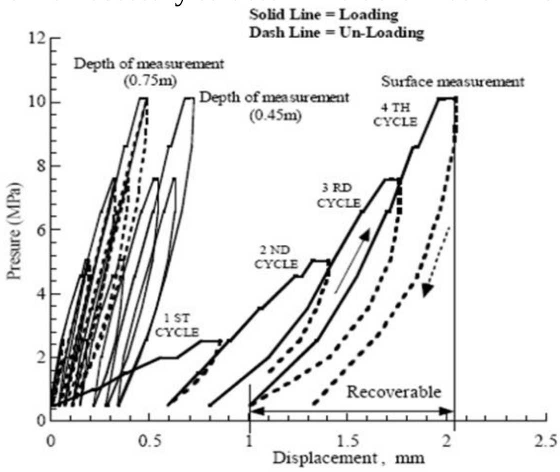


Figure 8. Pressure-displacement curves obtained from the PLT [37]

The recoverable displacement is used to evaluate the deformation modulus based on the theory of elasticity. Depending on the loading condition, the PLT can be classified into a flexible type and a rigid type. In this paper, the flexible PLT procedure suggested by the ISRM 1981 in which Boussinesq's equation is applied in the interpretation of the PLT results is used. An illustration of a PLT site is shown in Figure 9. The PLT was carried out in a test gallery, excavated by drill and blast, at the Gotvand dam to determine deformation modulus for the assessment of EDZ.

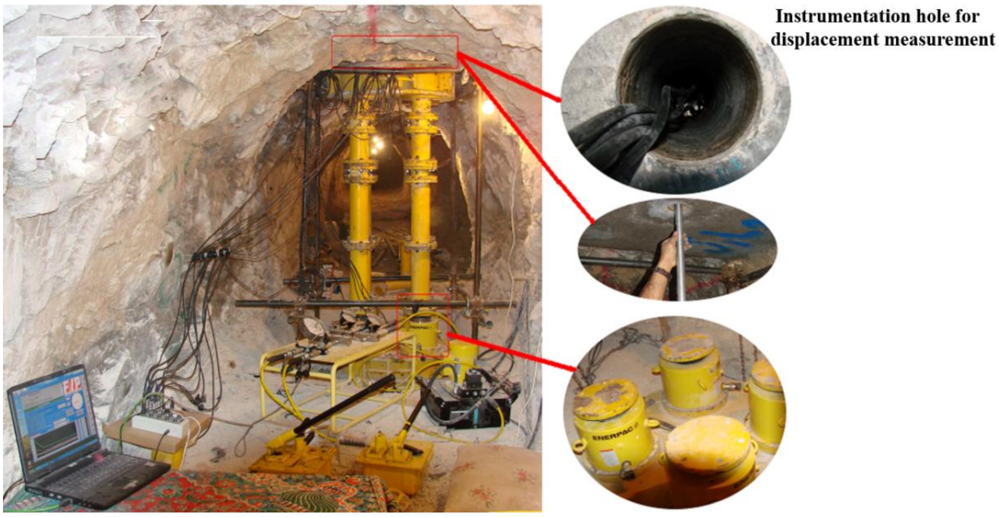


Figure 9. The set-up of PLT

## 5. RESEARCH METHODOLOGY

In this paper, a new methodology is introduced to determine the creation probability of excavation damaged zone around an underground structure. A primary issue the creation probability of excavation damaged zone based on the ANN-HPSOGA model is how to generate the training samples. Since the relation function is not known explicitly and is complicated, to make a relation between input variables and output to generate random value for each variable, the ANN-HPSOGA model can be used. In this paper, based upon the results of the PLT from a test gallery of the Gotvand dam, Iran and using the LHS based on the ANN-HPSOGA model, samples of data were generated (Figure 10). By using the data generated, the creation probability of damaged zone can be estimated.

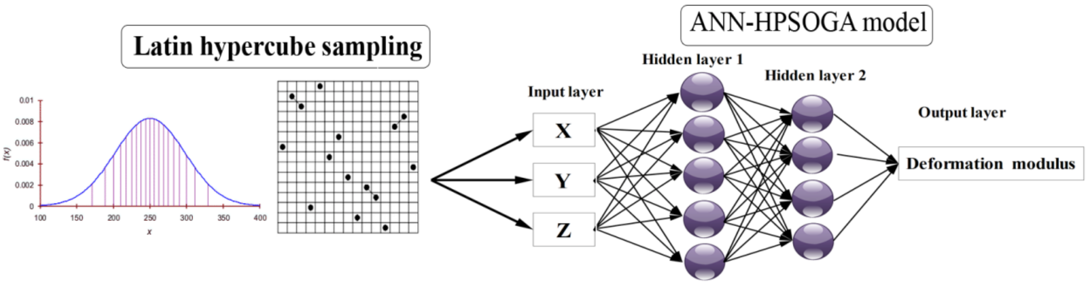


Figure 10. Latin hypercube sampling based on the ANN-HPSOGA model

### *5.1. Prediction of deformation modulus, using the ANN-HPSOGA model*

### *5.1.1 Tuning parameters for the GA and the PSO*

To develop an accurate ANN model, the training, and validation processes are the important steps. In the training process, a set of input-output patterns is repeated to the ANN. From that, weights of all the interconnections between neurons are adjusted until the specified input yields the desired output. Through these activities, the ANN learns the correct input-output response behavior. The model training stage includes choosing a criterion of fit (mean squared error) and an iterative search algorithm to find the network parameters that minimizes the criterion. Hybrid the GA with PSO (HPSOGA) was used in an effort to formalize a systematic approach to training the ANN, and to insure creation of a valid model. It was used to perform global search algorithms to update the weights and biases of neural network. The control parameters used for running the PSO and the GA are shown in Tables 1 and 2 respectively:

Table 1: The control parameters used for running the PSO

| **Parameter** | **Value** |
|---|---|
| Number of population (swarm size) | 50 |
| Number of generations | 1000 |
| Personal learning coefficient | 1.4962 |
| Global learning coefficient | 1.4962 |
| Inertia weights | 0.73 |
| Fitness | Mean squared error |

Table 2: The control parameters used for running the GA

| **Parameter** | **Value** |
|---|---|
| Number of population | 50 |
| Number of generations | 1000 |
| Crossover probability | 0.7 |
| Mutation probability | 0.2 |
| Selection function | Ranking |
| Fitness | Mean squared error |

### *5.1.2 Network architecture*

Architecture of the ANN model includes type of network, number of input and output neurons, transfer function, number of hidden layers as well as number of hidden neurons. Generally, the input neurons and output neurons are problem specific [11]. In this paper, multi-input single-output structure had been utilized; therefore, there will be only one output neuron. The architecture of the network is given in Table 3.

Also, it is important that the transfer function possesses the properties of differentiability and continuity. Generally, log sigmoid function is utilized in the hidden layer and the output generated has a value between 0 and 1 however, the linear transfer function is more suitable in output [11]. The equations for the log and linear transfer functions used in this work are shown in Eqs. (9) and (10):

$$f(x) = \frac{1}{1+\exp(-x)} \tag{9}$$

$$f(x) = x \tag{10}$$

Table 3: The architecture of the network

| **Parameter** | **Value** |
|---|---|
| No. of input neurons | 3 |
| No. of output neurons | 1 |
| No. of hidden layers | 2 |
| No. of neurons in first hidden layer | 5 |
| No. of neurons in second hidden layer | 4 |
| No. of training data sets | 79 |
| No. of testing data sets | 20 |

### *5.1.3 Training and validation results*

In this paper, the ANN-HPSOGA model was used to predict deformation modulus, using MATLAB environment. Figure 11 shows the architecture of the ANN-HPSOGA model used. As it can be seen in Figure 11, X, Y and Z coordinates (location of installation extensometers from the portal of test gallery that in these points, displacements and modulus of deformations were obtained) were defined as input parameters into the ANN-HPSOGA model and the deformation modulus as output. The model proposed was trained with 79 data sets collected from a test gallery in the Gotvand dam for training phase. A few samples of the training and testing data sets are shown in Tables 4 and 5 respectively.

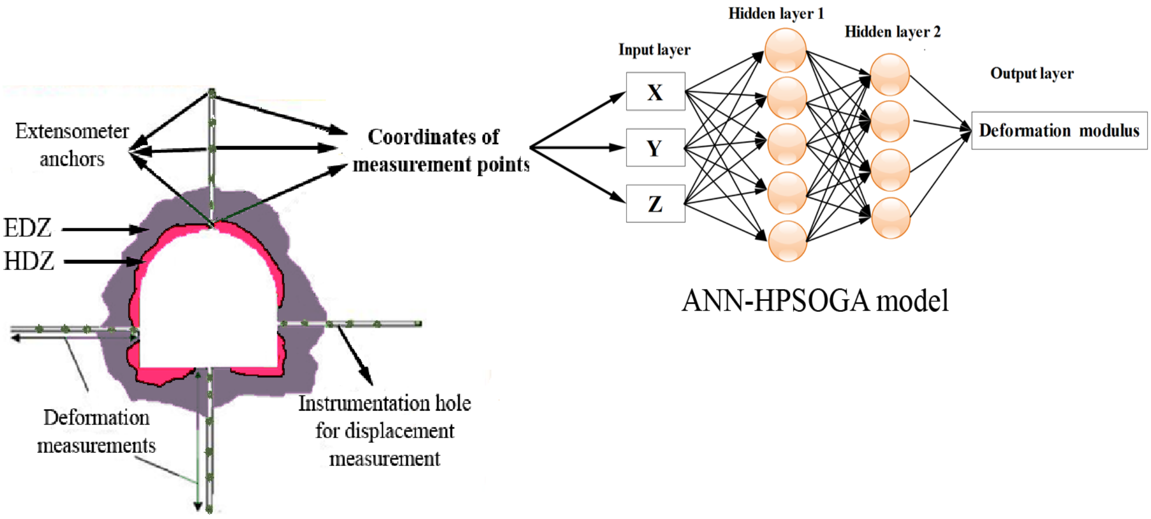
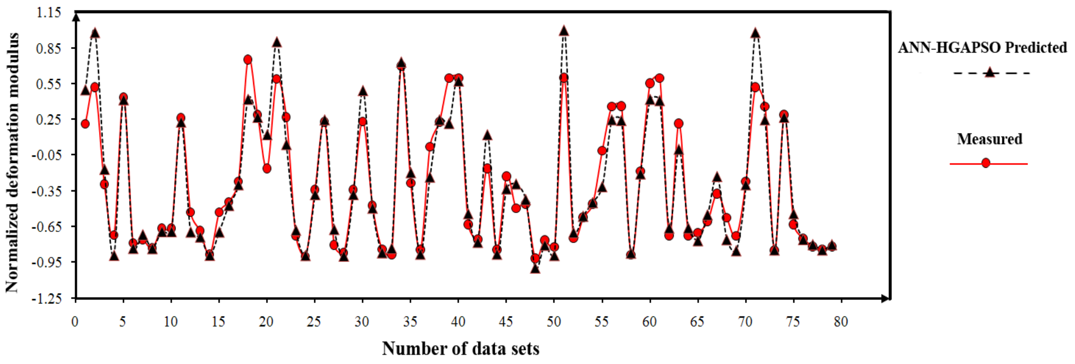


Figure 11. Architecture of the ANN-HPSOGA model

A comparison between predicted values of deformation modulus by the ANN-HPSOGA model and measured values for 79 data sets at training and testing phases is shown in Figure 12. As shown in Figure 12, the results of the ANN-HPSOGA model in comparison with actual data show a good precision of the ANN-HPSOGA model (see Table 7). It should be noted that the predicted and measured deformation modulus (Figure 12) represent normalized values that was calculated using the following equation:

$$Normalized\ deformation\ \text{modulus} = \frac{2(E - E_{\min})}{(E_{\max} - E_{\min})} - 1 \tag{11}$$

where, $E_{min}$ and $E_{max}$ are the minimum and maximum deformation modulus of the data used in this study, respectively.



(a)

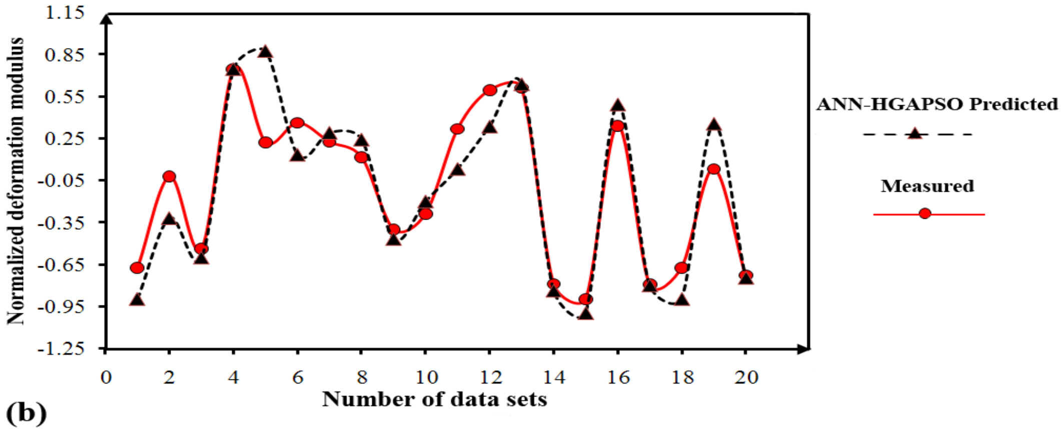


Figure 12. Comparison between measured and predicted deformation modulus **a** Training, **b** Testing

Table 4: A few samples of the training data sets, the ANN-HPSOGA model

| No. | Depth of extensometer in instrumentation hole (m) | Input |  |  | Output |
|---|---|---|---|---|---|
|  |  | X (m) | Y (m) | Z (m) | Deformation modulus (GPa) |
| 1 | 0.5 | 6 | 7.75 | 3 | 3.18 |
| 2 | 1 | 6 | 8.25 | 3 | 6.48 |
| 3 | 0.4 | 7.65 | 6 | 9 | 7.75 |
| 4 | 1.2 | 8.45 | 6 | 9 | 14.9 |
| 5 | 0.4 | 7.65 | 6 | 27 | 4.3 |
| 6 | 0 | 6 | 7.25 | 9 | 2.11 |
| 7 | 0.9 | 6 | 3.85 | 27 | 9.42 |
| 8 | 2.3 | 9.55 | 6 | 3 | 24.95 |
| 9 | 0.4 | 7.65 | 6 | 27 | 4.32 |
| 10 | 0.5 | 6 | 4.25 | 9 | 3.33 |

Table 5: A few samples for testing the ANN-HPSOGA model

| No. | Depth of extensometer in instrumentation hole (m) | Input |  |  | Output |
|---|---|---|---|---|---|
|  |  | X (m) | Y (m) | Z (m) | Deformation modulus (GPa) |
| 1 | 0 | 6 | 7.25 | 3 | 0.955 |
| 2 | 0.6 | 4.15 | 6 | 3 | 9.274 |
| 3 | 0.9 | 6 | 3.85 | 3 | 2.55 |
| 4 | 0.5 | 6 | 7.75 | 27 | 3.829 |
| 5 | 1.2 | 3.55 | 6 | 27 | 9.66 |

Also, performance prediction of the predictive model proposed was evaluated, using coefficient of determination ($R^2$), mean squared error (MSE), root mean square error (RMSE), median absolute error (MEDAE) and variance account for (VAF) (Table 6) where, $N$ is the number of samples, *var* denotes the variance, $y$ and $y'$ are the measured and predicted values, respectively.

Table 6: Statistical indicators

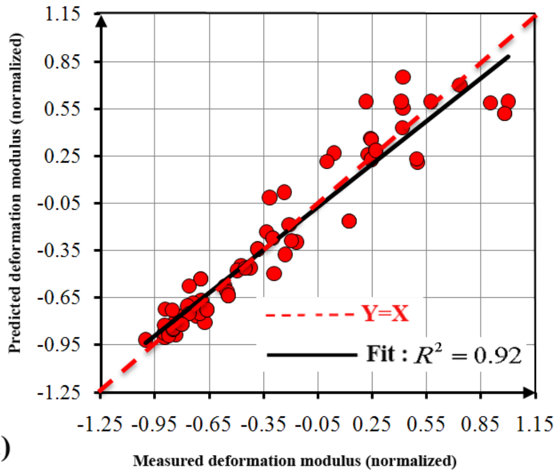
| Statistical indicator | Equation |
|---|---|
| **Mean squared error** | $MSE = \frac{1}{N}\sum_{i=1}^{N}(y - y')^2$ |
| **Root mean square error** | $RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y - y')^2}$ |
| **Median absolute error** | $\text{MEDAE} = \text{median}\,(y - y')$ |
| **Variance account for** | $VAF = \left(1 - \frac{\text{var}(y - y')}{\text{var}(y)}\right)$ |

Performance analysis of the ANN-HPSOGA model for predicting deformation modulus is shown in Table 7.

Table 7: Performance of the model for predicting deformation modulus

| Description | $R^2$ | MSE | RMSE | MEDAE | VAF |
|---|---|---|---|---|---|
| **Training data** | 0.92 | 0.02 | 0.16 | 0.02 | 90.98 |
| **Testing data** | 0.87 | 0.08 | 0.27 | 0.08 | 73.88 |

The performance indices obtained in Table 7 indicate the high performance of the ANN-HPSOGA model that can be used successfully for the prediction of deformation modulus. Furthermore, correlation between measured and predicted values of deformation modulus for training and testing phases are shown in Figure 13.



(a)

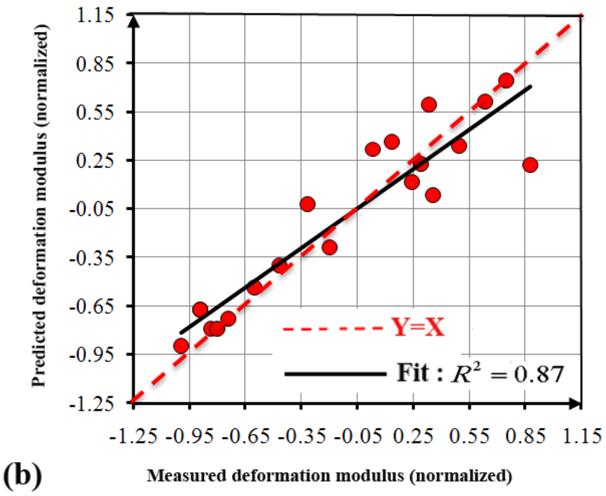


Figure 13. Correlation between measured and predicted values of deformation modulus **a** Training, **b** Testing

### *5.2 Estimation of the creation probability of damaged zone*

The actual modulus of deformation for rock mass in the Gotvand dam is 5.8 GPa [38]. Also, a threshold of less than 2 GPa was chosen to recognize the HDZ, which is a part of EDZ and a threshold of less than 5 GPa was chosen to recognize the EDZ. Therefore, limit state function was defined for EDZ as E<5 GPa (Figure 14) and HDZ as E<2 GPa.

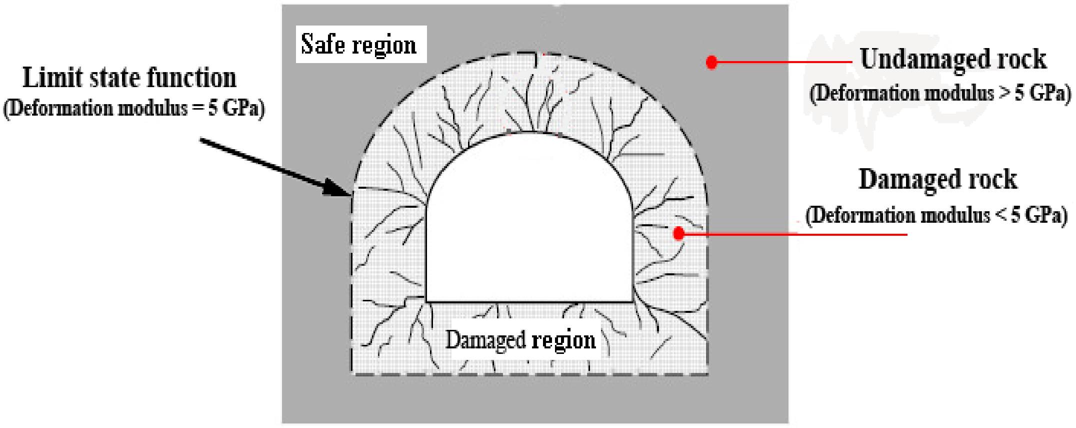

Safe region
Limit state function
(Deformation modulus = 5 GPa)
Undamaged rock
(Deformation modulus > 5 GPa)
Damaged rock
(Deformation modulus < 5 GPa)
Damaged region

Figure 14. Limit state function defined for the EDZ

To assess the probability of creation of EDZ and HDZ, the LHS based on ANN-HPSOGA model was used. The creation probability of damaged zone can be estimated as the ratio of the number of samples locating at the damaged region (HDZ, EDZ) ($N_f$) to the total number of samples generated ($N$) [Eq. (6)]. To check the convergence of the LHS, the probability of creation was calculated with 13 different values of $N$. The results obtained, listed in Table 8, indicate that the LHS with 30×$10^4$ samples (the number of sample points in the HDZ and the EDZ is 57182 and 150721 respectively) was converged and the probability of creation achieved for the HDZ and the EDZ are 0.191 and 0.502 respectively. A few samples of data generated, using the LHS based on the ANN-HPSOGA model is listed in Table 9.

Table 8: The creation probability of damaged zone for 13 different values of *N*

| ***N*** | ***N<sub></sub>*** | | | |
|---|---|---|---|---|

| ***N*** | $N_f$<br>**HDZ (*E* < 2)** | $N_f$<br>**EDZ (*E* <5)** | $P_f$<br>**HDZ (*E* < 2)** | $P_f$<br>**EDZ (*E* <5)** |
|---|---|---|---|---|
| **$0.1\times10^4$** | 181 | 496 | 0.181 | 0.496 |
| **$0.5\times10^4$** | 924 | 2535 | 0.185 | 0.507 |
| **$1\times10^4$** | 1880 | 5047 | 0.188 | 0.505 |
| **$5\times10^4$** | 9605 | 25056 | 0.192 | 0.501 |
| **$10\times10^4$** | 19023 | 50179 | 0.190 | 0.502 |
| **$15\times10^4$** | 28801 | 75504 | 0.192 | 0.503 |
| **$20\times10^4$** | 38071 | 100483 | 0.190 | 0.502 |
| **$25\times10^4$** | 48227 | 125674 | 0.193 | 0.503 |
| **$30\times10^4$** | 57182 | 150721 | 0.191 | 0.502 |
| **$35\times10^4$** | 66877 | 175652 | 0.191 | 0.502 |
| **$40\times10^4$** | 76248 | 200798 | 0.191 | 0.502 |
| **$45\times10^4$** | 85897 | 225873 | 0.191 | 0.502 |
| **$50\times10^4$** | 95408 | 250902 | 0.191 | 0.502 |

Table 9: A few samples of data generated, using the LHS based on ANN-HPSOGA model

| No. | Generated input by LHS | | | Generated output by ANN-HPSOGA model |
|---|---|---|---|---|
| | X (m) | Y (m) | Z (m) | Deformation modulus (GPa) |
| 1 | 6.1 | 8.5 | 17.8 | 8.6 |
| 2 | 5.5 | 5.2 | 12.5 | 3.3 |
| 3 | 5.5 | 7.1 | 15.1 | 1.8 |
| 4 | 9.1 | 7.4 | 9.7 | 18.3 |
| 5 | 4.4 | 4.0 | 27.5 | 14.6 |
| 6 | 8.5 | 5.8 | 15.6 | 11.4 |
| 7 | 7.4 | 6.3 | 20.2 | 3.8 |
| 8 | 6.6 | 2.9 | 18.8 | 6.6 |
| 9 | 7.7 | 5.0 | 11.5 | 2.9 |
| 10 | 4.1 | 8.1 | 17.9 | 1.8 |

The procedure of estimation of creation probability of damaged zone is summarized in Figure15.

## 6. CONCLUSIONS

The EDZ affects the behavior of rock mass surrounding the construction that reduces the stability and safety factor and increases probability of failure of the structure. In this research a methodology for determining the creation probability of damaged zone was proposed and the following remarks were concluded:

- Implementation hybrid the particle swarm optimization and genetic algorithm (HPSOGA) as an optimizer of connection weights of artificial neural network to predict the deformation modulus was demonstrated in detail.
- The ANN-HPSOGA model as a good tool can model the damage occurred due to blasting around an underground excavation based on the variation in the deformation modulus.
- The HPSOGA has high robustness in optimization issue due to integrating global search and local search abilities of the GA and the PSO. The results obtained showed that it can be used to tune the weights of the ANN model for the assessment of the damaged zone creation.
- Hybrid the LHS and ANN-HPSOGA when the relation function is not known explicitly and is complicated, is a powerful tool to make a relation between input variables and output to generate a value for each variable.
- The methodology suggested can be used effectively to find creation probability of the damaged zone, which can be led to a better understanding the risks of a project, a more efficient of establishing geotechnical zoning and the costs can be estimated with more reliability. Furthermore, it can be utilized for optimal design of blast pattern and support used.

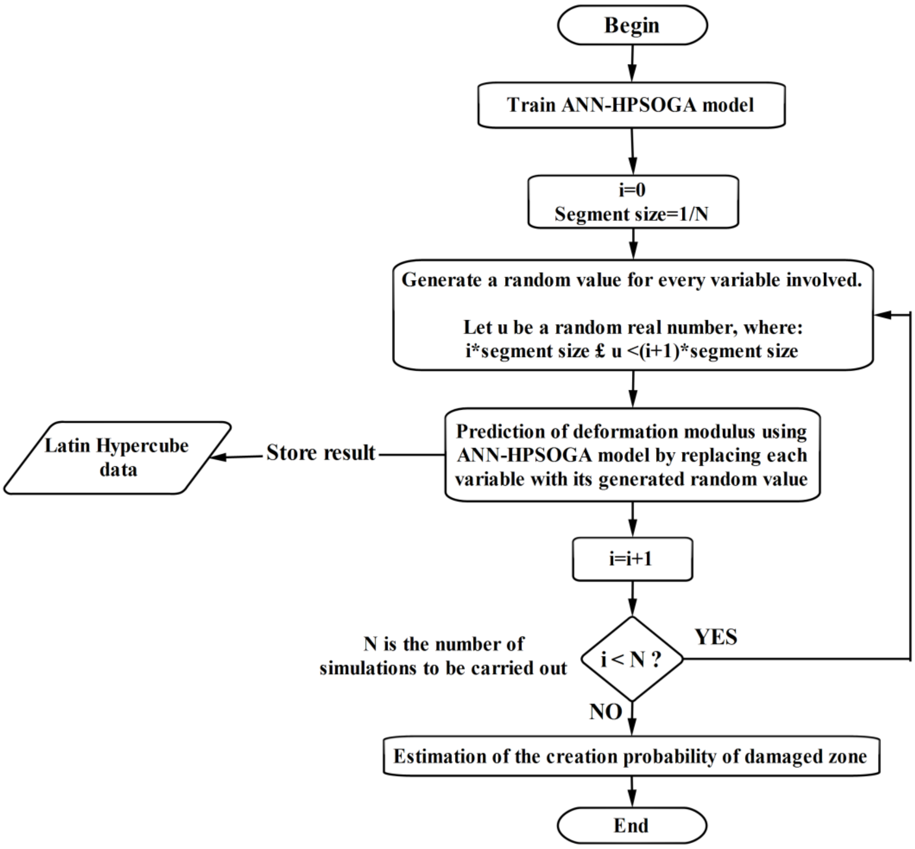

Begin
Train ANN-HPSOGA model
i=0
Segment size=1/N
Generate a random value for every variable involved.
Let u be a random real number, where:
i*segment size £ u <(i+1)*segment size
Latin Hypercube data
Store result
Prediction of deformation modulus using ANN-HPSOGA model by replacing each variable with its generated random value
i=i+1
N is the number of simulations to be carried out
i < N ?
YES
NO
Estimation of the creation probability of damaged zone
End

Figure15. The procedure of estimation of the creation probability of damaged zone

## REFERENCES

1. Saiang D. Stability analysis of the blast-induced damage zone by continuum and coupled continuum–discontinuum methods, *Eng Geol*, 2010; **116**(1): 1-11.

2. Malmgren L, Saiang D, Töyrä J, Bodare A. The excavation disturbed zone (EDZ) at Kiirunavaara mine, Sweden-by seismic measurements, *J Appl Geophys*, 2007; **61**(1): 1-15.

3. Nicollin F, Gibert D, Bossart P, Nussbaum C, Guervilly C. Seismic tomography of the excavation damaged zone of the gallery 04 in the Mont Terri rock laboratory, *Geophys J Int*, 2008; **172**(1): 226-39.

4. Kwon S, Lee CS, Cho SJ, Jeon SW, Cho WJ. An investigation of the excavation damaged zone at the KAERI underground research tunnel, *Tunn Undergr Sp Tech*, 2009; **24**(1): 1-13.

5. Holmberg R, Persson PA. Design of tunnel perimeter blasthole patterns to prevent rock damage. transactions of the institution of mining and metallurgy1980. p. 37-40.

6. Zang TA, Hemsch MJ, Hilburger MW, Kenny SP, Luckring JM, Maghami P, et al. Needs and opportunities for uncertainty-based multidisciplinary design methods for aerospace vehicles, National Aeronautics and Space Administration, Langley Research Center, 2002.

7. Zentner JM. A design space exploration process for large scale, multi-objective computer simulations, PhD thesis, Georgia Institute of Technology, 2006.

8. Qu B, Guo X, Chi H, Pollino M. Probabilistic evaluation of effect of column stiffness on seismic performance of steel plate shear walls, *Eng Struct*, 2012; **43**(169-79.

9. Sexton RS, Dorsey RE, Sikander NA. Simultaneous optimization of neural network function and architecture algorithm, *Decis Support Syst*, 2004; **36**(3): 283-96.

10. Downs JJ, Vogel EF. A plant-wide industrial process control problem, *Comput Chem Eng*, 1993; **17**(3): 245-55.

11. Braik M, Sheta A, Arieqat A. A comparison between GAs and PSO in training ANN to model the TE chemical process reactor, I*n: Proceedings of the AISB 2008 symposium on swarm intelligence algorithms and applications*, 2008, pp. 24-30

12. Montana DJ, Davis L. Training feedforward neural networks using genetic algorithms, *In: Proceedings of the eleventh international joint conference on artificial Intelligence*, 1989, pp. 762-7

13. McKay M, Beckman R, Conover W. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, *Technometrics*, 2000; **42**(1): 55-61.

14. Olsson A, Sandberg G, Dahlblom O. On Latin hypercube sampling for structural reliability analysis, *Struct Saf*, 2003; *25*(1): 47-68.

15. Owen AB. Controlling correlations in Latin hypercube samples, *J Am Stat Assoc*, 1994; **89**(428): 1517-22.

16. Ding K, Zhou Z, Liu C. Latin hypercube sampling used in the calculation of the fracture probability, *Reliab Eng Syst Safe*, 1998; **59**(2): 239-42.

17. Mahadevan S. Monte Carlo simulation. In: Cruse T, editor. Reliability-based mechanical design. New York: Marcel Dekker, 1997. p. 123-46.

18. Hornik K, Stinchcombe M, White H. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks, *Neural Networks*, 1990; **3**(5): 551-60.

19. García-Pedrajas N, Hervás-Martínez C, Muñoz-Pérez J. COVNET: a cooperative coevolutionary model for evolving artificial neural networks, *IEEE T Neural Networ*, 2003; **14**(3): 575-96.

20. Jang J-S, Sun C-T. Neuro-fuzzy modeling and control, *P IEEE*, 1995; **83**(3): 378-406.

21. Ahmadi MH, Aghaj SSG, Nazeri A. Prediction of power in solar stirling heat engine by using neural network based on hybrid genetic algorithm and particle swarm optimization, *Neural Comput Appl*, 2013; **22**(6): 1141-50.

22. Kamyab R, Salajegheh E. Prediction of nonlinear time history deflection of scallop domes by neural networks, I*nt J Optim Civil Eng*, 2011; **3**(419-32.

23. Muthupriyaa P, Subramanianb K, Vishnuramc B. Prediction of compressive strengh and durability of high performance concrete by artificial neural networks, *Int J Optim Civil Eng*, 2011; **1**(189-209.

24. Rofooei F, Kaveh A, Farahani F. Estimating the vulnerability of the concrete moment resisting frame structures using artificial neural networks, *Int J Optim Civil Eng*, 2011; **3**(433-48.

25. Gholizadeh S, Sheidaii M, Farajzadeh S. Seismic design of double layer grids by neural networks, *Int J Optim Civil Eng*, 2012; **2**(1): 29-45.

26. Ghasemi M, Barghi E. Estimation of inverse dynamic behavior of MR dampers using artificial and fuzzy-based neural networks, *Int J Optim Civil Eng*, 2012; **2**(3): 357-68.

27. Holland John H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*, MIT Press Cambridge, MA, USA, 1975.

28. Hassan R, Cohanim B, De Weck O, Venter G. A comparison of particle swarm optimization and the genetic algorithm, *In: Proceedings of the 1st AIAA multidisciplinary design optimization specialist conference*, 2005, pp. 18-21

29. Eberhart R, Kennedy J. A new optimizer using particle swarm theory, In: Micro Machine and Human Science, *1995 MHS'95, Proceedings of the Sixth International Symposium on*, 1995, pp. 39-43

30. Shi Y, Eberhart R. A modified particle swarm optimizer, In: Evolutionary Computation Proceedings, *IEEE World Congress on Computational Intelligence*, 1998, pp. 69-73

31. Chen S-F. Redundant Feature Selection Based on Hybrid GA and BPSO, *In: Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, 2011, pp. 414-8

32. Ali Ahmadi M. Prediction of minimum miscible pressure by using neural network based hybrid genetic algorithm and particle swarm optimization, *J Petrol Sci Eng*, 2012; Withdrawn Article in Press(

33. Kaveh A, Malakouti Rad S. Hybrid genetic algorithm and particle swarm optimization for the force method-based simultaneous analysis and design, *Iranian Journal of Science and Technology, Transaction B: Engineering*, 2010; **34**(B1): 15-34.

34. Juang C-F. A hybrid of genetic algorithm and particle swarm optimization for recurrent network design, *IEEE Trans Syst Man Cybern B Cybern*, 2004; **34**(2): 997-1006.

35. Tsang CF, Bernier F, Davies C. Geohydromechanical processes in the Excavation Damaged Zone in crystalline rock, rock salt, and indurated and plastic clays-in the context of radioactive waste disposal, *Int J Rock Mech Min Sci*, 2005; **42**(1): 109-25.

36. Palmström A, Singh R. The deformation modulus of rock masses - comparisons between in situ tests and indirect estimates, *Tunn Undergr Sp Tech*, 2001; **16**(2): 115-31.

37. Faramarzi L, Fattahi H. Interpretation of plate loading test results on major project, *In: 5th Asian Rock Mechanics Symposium (ARMS5)*, Tehran, Iran, 2008, pp. 255-60

38. Bashari A, Beiki M, Talebinejad A. Estimation of deformation modulus of rock masses by using fuzzy clustering-based modeling, *Int J Rock Mech Min Sci*, 2011; **48**(8): 1224-34.