Research Article

Back Analysis of Rock Hydraulic Fracturing by Coupling Numerical Model and Computational Intelligent Technology

Shaojun Li, Hongbo Zhao, Xiating Feng, Quan Jiang, and Qiancheng Sun

1 State Key Laboratory of Geomechanics and Geotechnical Engineering, Institute of Rock and Soil Mechanics, Chinese Academy of Sciences, Wuhan, Hubei 430071, China
2 School of Civil Engineering, Henan Polytechnic University, Jiaozuo 454003, China

Correspondence should be addressed to Shaojun Li; sjli@whrsm.ac.cn

Received 17 June 2017; Revised 23 August 2017; Accepted 7 September 2017; Published 18 October 2017

Academic Editor: Zhongwei Chen

Copyright © 2017 Shaojun Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Hydraulic fracturing is widely used to determine in situ stress of rock engineering. In this paper we propose a new method for simultaneously determining the in situ stress and elastic parameters of rock. The method utilizing the hydraulic fracturing numerical model and a computational intelligent method is proposed and verified. The hydraulic fracturing numerical model provides the samples which include borehole pressure, in situ stress, and elastic parameters. A computational intelligent method is applied in back analysis. A multioutput support vector machine is used to map the complex, nonlinear relationship between the in situ stress, elastic parameters, and borehole pressure. The artificial bee colony algorithm is applied in back analysis to find the optimal in situ stress and elastic parameters. The in situ stress is determined using the proposed method and the results are compared with those of the classic breakdown formula. The proposed method provides a good estimate of the relationship between the in situ stress and borehole pressure and predicts the maximum horizontal in situ stress with high precision while considering the influence of pore pressure without the need to estimate Biot’s coefficient and other parameters.

1. Introduction

Hydraulic fracturing is widely used in the recovery of oil, gas, geothermal, and mineral resources [1]. In petroleum engineering it is important to determine the in situ stresses and elastic parameters of the rock mass when using hydraulic fracturing in fracturing operations, wellbore stability analysis, and reservoir simulation [2]. While high accuracy is required for the values of the in situ stress and mechanical parameters of the rock mass, determination of these parameters is still one of the most challenging tasks in hydraulic fracturing.

Hydraulic fracturing tests are considered the most effective method for determining the in situ stress and mechanical parameters of rock mass [3–9]. The Hubbert and Willis hydraulic fracturing criterion and Haimson and Fairhurst’s hydraulic fracturing criterion are the two classic formulae for hydrofracture breakdown ([10]; Hubbert et al., 1953). However, the pore pressure term, which is a significant factor in deep boreholes, is ignored in Hubbert and Willis’s hydraulic fracturing criterion. Modifications of the original equations were proposed to account for the pore pressure (Detournay et al., 1988; [1, 11–13]), but they have not been used in practice because it involves the Biot poroelastic parameters and Poisson’s ratio which are difficult to obtain. Schmitt and Zoback built a more useful generalized form of the hydrofracture breakdown equation by considering the poroelastic effects [1]. It can be used to provide upper and lower bound to the maximum horizontal in situ stress because it depends on the specific pore and microcrack structure. However, this method requires the poroelastic coefficients which are difficult to determine in practice.

Owing to the limitations of the classic breakdown formulae and the complexity of hydraulic fracturing tests, laboratory and field tests have been commonly used to determine the in situ stress and mechanical parameters of rock mass (Algorithm 1). However, these tests may not always produce the poroelastic parameters or may provide inaccurate results.
Sub MSVM()

Dim N As Integer ' The number of training samples
Dim Dim_x As Integer ' The dimension of input variables
Dim Dim_y As Integer ' The number of output variables
Dim C As Double ' Penalty factor of SVM
Dim epsilon As Double
Dim sigma As Double

Dim X, input() As Double ' The input of training samples
Dim Y, output() As Double ' The output of training samples

' The one sample
Dim x1() As Double
Dim x2() As Double
'The weights of MSVM
Dim W() As Double
Dim b() As Double
Dim W_k() As Double
Dim b_k() As Double
Dim W_s() As Double
Dim b_s() As Double

' the error of each sample
Dim u() As Double
Dim a() As Double
Dim u_new() As Double

'The coefficient of matrix for computing Ws and bs
Dim A() As Double
Dim B() As Double
Dim A_last_row() As Double
Dim B_last_row() As Double

'The Descending direction
Dim P_W() As Double
Dim P_b() As Double

Dim k() As Double
Dim D_a() As Double
Dim D_a_L() As Double
Dim k_f() As Double

Dim I As Integer
Dim j As Integer
Dim l As Integer

'The step size eta_k
Dim eta As Double

'The control parameters of algorithm convergence
Dim delta_u As Double
Dim t As Integer

Algorithm I: Continued.
The value of the numbers of parameters of training samples
N = Range("N").Cells.Value
Dim_x = Range("Dim_x").Cells.Value
Dim_y = Range("Dim_y").Cells.Value

The parameters of SVM
C = Range("C").Cells.Value
epsilon = Range("epsilon").Cells.Value
sigma = Range("sigma").Cells.Value

ReDim X_input(1 To N, 1 To Dim_x) As Double
ReDim Y_output(1 To N, 1 To Dim_y) As Double
ReDim xi(1 To Dim_x) As Double
ReDim xj(1 To Dim_x) As Double
ReDim W(1 To Dim_y, 1 To N) As Double
ReDim b(1 To Dim_y) As Double
ReDim W_k(1 To Dim_y, 1 To N) As Double
ReDim b_k(1 To Dim_y) As Double
ReDim W_s(1 To Dim_y, 1 To N) As Double
ReDim b_s(1 To Dim_y) As Double
ReDim u(1 To N) As Double
ReDim a_i(1 To N) As Double
ReDim u(1 To Dim_y) As Double
ReDim u_new(1 To N) As Double
ReDim A(1 To N + 1, 1 To N + 1) As Double
ReDim X(1 To N + 1, 1 To Dim_y) As Double
ReDim B(1 To N + 1, 1 To Dim_y) As Double
ReDim A_last_row(1 To N) As Double
ReDim BLast_row(1 To Dim_y) As Double
ReDim P_W(1 To Dim_y, 1 To N) As Double
ReDim P_b(1 To Dim_y) As Double
ReDim k(1 To N, 1 To N) As Double
ReDim D_a(1 To N, 1 To N) As Double
ReDim D_a_j(1 To N, 1 To N) As Double
ReDim k(1 To N) As Double

' Read the input of training samples
For I = 1 To N
    For j = 1 To Dim_x
        X_input(I, j) = Range("xi").Cells(I, j)
    Next
Next

' Read the output of training samples
For I = 1 To N
    For j = 1 To Dim_y
        Y_output(I, j) = Range("yi").Cells(I, j).Value
    Next
Next

' The initial value of Wk and bk
For I = 1 To Dim_y
    b(I) = 0
    For j = 1 To N
        W(I, j) = 0
    Next
Next

delta_u = 1

Algorithm 1: Continued.
t = 0

The iteration process of algorithm
While (delta_u > 0.001 And t < 100)

' Replace the value of Wk and bk by the New w and b
For I = 1 To Dim_y
    For j = 1 To N
        W_k(I, j) = W(I, j)
    Next
    b_k(I) = b(I)
Next

' Compute the value of ui and ai
For I = 1 To N
    u(I) = 0
    For j = 1 To Dim_x
        xi(j) = X_input(I, j)
    Next
    For ii = 1 To N
        For j = 1 To Dim_x
            xj(j) = X_input(ii, j)
        Next
        kf(ii) = kernel_fun(xi, xj, sigma)
    Next
    For ii = 1 To Dim_y
        ui(ii) = b_k(ii)
        For j = 1 To N
            ui(ii) = ui(ii) + W_k(ii, j) * kf(j)
        Next
    Next
    For j = 1 To Dim_y
        u(I) = u(I) + (Y_output(I, j) - ui(j))^2
    Next
    u(I) = Sqr(u(I))
    If (u(I) < epsilon) Then ai(I) = 0
    If (u(I) >= epsilon) Then ai(I) = 2 * C * (u(I) - epsilon)/u(I)
    ai(i) = 2 * C * (u(i) - epsilon)/u(i)
Next

'compute the Matrix K and Da
For I = 1 To N
    For j = 1 To N
        For i = 1 To Dim_x
            xi(i) = X_input(I, i)
        Next
        xj(j) = X_input(j, i)
        For l = 1 To Dim_x
            xi(l) = X_input(I, l)
            xj(l) = X_input(I, l)
        Next
        k(I, j) = kernel_fun(xi, xj, sigma)
        If (I = j) Then D_a(I, j) = ai(I) Else D_a(I, j) = 0
    Next
Next

For I = 1 To N
    For j = 1 To N
        If (I = j) Then A(I, j) = k(I, j) + 1/D_a(I, j) Else A(I, j) = k(I, j)
Next

Algorithm 1: Continued.
Next
Next

' Compute Transpose(a)*K
For I = 1 To N
    A_last_row(I) = 0
    For j = 1 To N
        A_last_row(I) = A_last_row(I) + ai(j) * k(j, I)
    Next
Next

A(N + 1, N + 1) = 0
For I = 1 To N
    A(N + 1, I) = A_last_row(I)
    A(N + 1, N + 1) = A(N + 1, N + 1) + ai(I)
    A(I, N + 1) = 1
Next

For I = 1 To Dim_y
    B_last_row(I) = 0
    For j = 1 To N
        B_last_row(I) = B_last_row(I) + ai(j) * Y_output(j, I)
    Next
Next

For I = 1 To N
    For j = 1 To Dim_y
        BB(I, j) = Y_output(I, j)
    Next
Next

' Compute Ws and bs
With Application.WorksheetFunction
    x1 = .MMult(.MInverse(A), BB)
End With

For I = 1 To Dim_y
    For j = 1 To N
        Ws(I, j) = x1(j, I)
    Next
    bs(I) = x1(N + 1, I)
Next

' Compute the descending direction
For I = 1 To Dim_y
    For j = 1 To N
        PW(I, j) = Ws(I, j) - Wk(I, j)
    Next
    Pb(I) = bs(I) - b_k(I)
Next

eta = 1
Dim delta_Lp As Double
Dim Lp.k,1 As Double  
Dim Lp.k As Double  

delta_Lp = 1  

'Update the solution of W and b  
While (delta_Lp > 0.0001)  

    For I = 1 To Dim_y  
        For j = 1 To N  
            W(I, j) = (1 - eta) * W.k(I, j) + eta * P.W(I, j)  
            Next  
            b(I) = (1 - eta) * b.k(I) + eta * P.b(I)  
            Next  
    For I = 1 To N  
        u_new(I) = 0  
    For j = 1 To Dim_x  
        xi(j) = X_input(I, j)  
        Next  
    For ii = 1 To N  
        For j = 1 To Dim_x  
            xj(j) = X_input(ii, j)  
            Next  
            kf(ii) = kernel_fun(xi, xj, sigma)  
            Next  
    For ii = 1 To Dim_y  
        ui(ii) = b(ii)  
        For j = 1 To N  
            ui(ii) = ui(ii) + W(ii, j) * kf(j)  
            Next  
        Next  
    For j = 1 To Dim_y  
        u_new(I) = u_new(I) + (Y_output(I, j) - ui(j))^2  
        Next  
        u_new(I) = Sqr(u_new(I))  
        If (u_new(I) < epsilon) Then a(I) = 0  
        If (u_new(I) >= epsilon) Then a(I) = 2 * C * (u_new(I) - epsilon)/u_new(I)  
        delta_u = delta_u + u_new(I)  
    Next  
    Lp.k,1 = 0  
    Lp.k = 0  
    For I = 1 To Dim_y  
        For j = 1 To N  
            Lp.k,1 = Lp.k,1 + W(I, j)^2/2  
            Lp.k = Lp.k + W(I, j)^2/2  
            Next  
        Next  
    For j = 1 To N  
        If u_new(j) >= epsilon Then Lp.k = Lp.k + C * (u_new(j)^2 - 2 * u(j) * epsilon + epsilon^2)  
        If u(j) >= epsilon Then Lp.k = Lp.k + C * (u(j)^2 - 2 * u(j) * epsilon + epsilon^2)  
        Next  

Algorithm 1: Continued.
Next

delta_Lp = Lp_k - Lp_k-1
eta = 0.5 * eta
Wend

delta_u = 0
For I = 1 To N
   delta_u = delta_u + u_new(I)
Next

delta_u = delta_u/N

t = t + 1

Wend

For I = 1 To N
   For j = 1 To Dim_y
      Range("wi").Cells(I, j) = W(j, I)
   Next
Next

For j = 1 To Dim_y
   Range("bi").Cells(j) = b(j)
Next

End Sub

' Kernel function of RBF
Function kernel_fun(xx, yy, sigma2) As Double
   Dim temp As Double
   Dim temp1 As Double
   Dim Dim_x As Integer

   Dim_x = Range("Dim_x").Cells.Value
   temp = 0
   For I = 1 To Dim_x
      temp = temp + (xx(I) - yy(I))^2
   Next

   temp1 = Sqr(temp)/(2 * sigma2^2)

   kernel_fun = Exp(-temp1)
   'Kf = (temp + 1)^sigma2
End Function

' Compute the performance function value using the MSVM
Sub Perfunc()
   Dim l As Integer
   Dim ii As Integer
   Dim I As Integer
   Dim j As Integer
   Dim N As Integer
   Dim Np As Integer
   Dim sigma As Double

   Dim xi() As Double
   Dim xi() As Double

   Algorithm 1: Continued.
Algorithm 1: The code of MSVM.

```
Dim xj() As Double
Dim W() As Double
Dim b() As Double
Dim Yy_p() As Double
Dim kf() As Double

N = Range("N").Cells.Value
Np = Range("Np").Cells.Value
Dim_x = Range("Dim_x").Cells.Value
Dim_y = Range("Dim_y").Cells.Value
sigma = Range("sigma").Cells.Value

ReDim xi(1 To Dim_x) As Double
ReDim xj(1 To Dim_x) As Double
ReDim W(1 To Dim_y, 1 To N) As Double
ReDim b(1 To Dim_y) As Double
ReDim Yy_p(1 To Np, 1 To Dim_y) As Double
ReDim kf(1 To N) As Double

For I = 1 To Dim_y
    For j = 1 To N
        W(I, j) = Range("w").Cells(j, I)
    Next
    b(I) = Range("b").Cells(I)
Next

For I = 1 To Np
    For j = 1 To Dim_x
        xi(j) = Range("x_p_input").Cells(I, j)
    Next
    For l = 1 To N
        For j = 1 To Dim_x
            xj(j) = Range("x").Cells(l, j)
        Next
        kf(l) = kernel_fun(xi, xj, sigma)
    Next
    For ii = 1 To Dim_y
        Yy_p(I, ii) = b(ii)
        For j = 1 To N
            Yy_p(I, ii) = Yy_p(I, ii) + W(ii, j) * kf(j)
        Next
    Next
Next

For I = 1 To Np
    For j = 1 To Dim_y
        Range("y_p_output").Cells(I, j) = Yy_p(I, j)
    Next
Next

End Sub
```
because of low-quality core samples [14]. Alternatively, back analysis, associated with the “in situ” approach, has been widely used to determine the mechanical parameters of rock mass in rock engineering [15–20]. Zhang and Yin proposed a back analysis method which combined a neural network and a genetic algorithm to simultaneously identify the in situ stresses and elastic parameters [2]; however, this method did not consider the poroelastic effect. To overcome this difficulty, in this paper we extend our proposed displacement back analysis method to determine the in situ stress and mechanical parameters of a rock mass based on measured borehole pressure. The borehole pressure can be easily measured in the field with a pressure gauge installed inside the borehole [21]. Back analysis is implemented following an optimization strategy based on the multioutput support vector machine (MSVM) and artificial bee colony algorithm (ABC) model, which is effective in multiple parameter identification [22].

The rest of this paper is organized as follows. The classic breakdown formula is presented in detail in Section 2. The formulation and procedure of back analysis based on borehole pressure are presented in detail in Section 3. In Section 4, a numerical example is used to verify the proposed method, and our conclusions are presented in Section 5.

2. Hydraulic Breakdown Equations

Hydraulic fracturing is a widely accepted technology used for determining in situ stress magnitude and direction. The principal stress $\sigma$ has a magnitude equal to the overburden pressure in the vertical direction. The smallest horizontal principal stress $\sigma_{H_{\text{min}}}$ is usually determined directly in the experiment from the shut-in pressure. The greatest horizontal principal stress $\sigma_{H_{\text{max}}}$ must be calculated using a breakdown formula derived from an appropriate hydraulic fracturing model. Hubbert and Willis proposed a classic breakdown formula (1) to calculate $\sigma_{H_{\text{max}}}$ for hydraulic fracturing in nonporous impermeable rocks [23], ignoring the pore pressure term.

$$ P_b = 3\sigma_{H_{\text{min}}} - \sigma_{H_{\text{max}}} + T, $$ (1)

where $T$ is the rock tensile strength.

Equations (2) and (3) are the breakdown formulae of porous impermeable rocks and porous permeable rocks, respectively, including the pore pressure [24]:

$$ P_b = 3\sigma_{H_{\text{min}}} - \sigma_{H_{\text{max}}} + T - P_p $$ (2)

$$ P_b = \frac{3\sigma_{H_{\text{min}}} - \sigma_{H_{\text{max}}} + T - \alpha ((1 - 2\nu) / (1 - \nu)) P_p}{1 + \beta - \alpha ((1 - 2\nu) / (1 - \nu))}, $$(3)

where $P_b$ is the breakdown pressure, $P_p$ is the pore pressure, $\alpha$ is the Biot poroelastic parameter, and $\nu$ is Poisson’s ratio. Although (3) may best describe the conditions under which hydraulic fracturing is conducted from an open borehole, (2) is used in practice because of the difficulty of determining $\alpha$ and $\nu$. Schmitt and Zoback proposed a more generalized form for the equations of hydrofracture breakdown for porous impermeable rocks and porous permeable rocks [1]:

$$ P_b = 3\sigma_{H_{\text{min}}} - \sigma_{H_{\text{max}}} + T - \beta P_p $$ (4)

$$ P_b = \frac{3\sigma_{H_{\text{min}}} - \sigma_{H_{\text{max}}} + T - \alpha ((1 - 2\nu) / (1 - \nu)) P_p}{1 + \beta - \alpha ((1 - 2\nu) / (1 - \nu))}, $$ (5)

where $\beta$ is the poroelastic effect parameter.

3. Back Analysis Model Based on Borehole Pressure

The in situ stress can be estimated based on borehole pressure using the hydraulic breakdown equations (Section 2). However, these equations present some limitations in practice. Therefore, we propose a back analysis method that combines a numerical method and an intelligent computational method. A multioutput support vector machine (MSVM) is used to map the complex, nonlinear relationship between the in situ stress, elastic parameters, and borehole pressure. The numerical method provides the training samples for the MSVM. It is important to use an optimization method in back analysis. Here, we use the ABC algorithm to find the best-fit in situ stress and elastic parameters by comparing the measured pressure data and the MSVM predicted pressure.

3.1. Nonlinear Relationships between Pressure and Geomechanical Parameters. The relationship between the borehole pressure and geomechanical parameters can be derived by the MSVM. The basic idea of MSVM is to extend the single-output support vector machine to a multidimensional output case. Given a set of training samples $\{(X_1,Y_1),(X_2,Y_2),\ldots,(X_N,Y_N)\}$, $X_i \in \mathbb{R}^q$, $Y_i \in \mathbb{R}^q$, the MSVM model can be built by solving the following optimization problem based on an iterative reweighted least-square algorithm [25].

$$ L_p(W,b) = \frac{1}{2} \sum_{j=1}^{Q} \|W_j\|^2 + \frac{1}{2} \sum_{j=1}^{N} a_i u_i^j + \tau $$ (6)

where $N$ is the number of input, $Q$ is the number of output, $W$ is the weight, $b$ and $C$ are constants, $\tau$ is the sum of constant terms that do not depend on either $W$ or $b$, $\varepsilon$ is the tolerant error, and $t$ denotes the $t$th iteration. A brief description and the MSVM algorithm can be found in the literature [22]. The MSVM model can be expressed as

$$ Y(X) = \sum_{k=1}^{n} W_k(X, X_k) + b. $$ (7)
Based on the above MSVM model, the nonlinear relationship between the borehole pressure and geomechanical parameters can be described as

\[
\text{MSVM}(X) : R^n \rightarrow R^Q
\]

\[Y = \text{MSVM}(X),\]

where \(X = (x_1, x_2, \ldots, x_n)\) is the \(n\)-dimensional vector of the identified parameter, for example, the in situ stress, Young's modulus, or Poisson's ratio. \(Y = (y_1, y_2, \ldots, y_Q)\) is the \(Q\)-dimensional vector of the borehole pressure.

To build the MSVM model, the necessary training or learning samples are constructed and the MSVM parameters are determined. The samples are constructed by numerical analysis which computes the corresponding borehole pressure for a given set of tentative determined parameters. The MSVM parameters have a strong influence on the performance of the MSVM. In this study, we determined these parameters using the formulation presented by Meza et al. [26].

3.2. Optimization Method. The back analysis ABC algorithm, developed by Karaboga [27], was adopted to search for the optimal geomechanical parameters of the rock mass. In the algorithm, the colony of artificial bees consists of three groups: employed bees, onlookers, and scouts. The ABC algorithm involves a cycle of four phases: the initialization phase, employed bees phase, onlooker bee phase, and scout bee phase.

In the initialization phase, the ABC generates a randomly distributed initial population of SN solutions and calculates the fitness of each solution.

\[x(i, j) = x_{ij}^{\text{min}} + \text{rand}(0, 1) \left( x_{ij}^{\text{max}} - x_{ij}^{\text{min}} \right), \quad (9)\]

where \(x(i, j)\) is the candidate solution of the problem; \(i = 1, 2, \ldots, \text{SN}/2\) and \(\text{SN}/2\) denotes the size of the population; \(j = 1, 2, \ldots, D\) and \(D\) is the dimension number of each solution; \(\text{rand}(0, 1)\) is a random number between \([0, 1]\); \(x_{ij}^{\text{min}}\) and \(x_{ij}^{\text{max}}\) are the upper and lower bounds of each solution.

Once initialization is completed, the employed bees search for a solution and calculate the fitness value (see Section 3.3) in the employed bees phase. A candidate solution is produced according to the following equation:

\[v(i, j) = x(i, j) + \varphi_{ij}(x(i, j) - x(k, j)), \quad (10)\]

where \(k\) is different from \(i\) and is a randomly chosen index from \([1, 2, \ldots, \text{SN}/2]\), \(j\) is also an index randomly chosen from \([1, 2, \ldots, D]\), and \(\varphi_{ij}\) is a random number in the range \([-1, 1]\) that controls the generation of food sources around \(x(i, j)\) and represents the comparison of two food positions seen by a bee.

In the onlooker bee phase, the onlooker bees choose a solution based on the fitness value, determine which solution will be abandoned, and allocate its employed bees as scout bees. The probability of being selected for each fitness value can be expressed as

\[p_i = \frac{\text{fitness}_{s_i}}{\sum_{i=1}^{\text{SN}} \text{fitness}_{s_i}}, \quad (11)\]

where \(\text{fitness}_{s_i}\) is the fitness value of the solution.

Finally, in the scout bee phase the scout bees randomly search for a new solution in the determined ranges. A solution that cannot be improved further through a predetermined number of cycles is assumed to be abandoned by the onlookers.

3.3. The Fitness Function. In order to find the optimal solution, it is necessary to build the fitness function for the ABC algorithm; that is,

\[\text{fitness} = \sqrt{\frac{\|\text{MSVM}(X) - Y\|^2}{Q}}, \quad (12)\]

where MSVM\((X)\) is the predicted pressure using the MSVM model, \(Y\) is the vector of the monitored pressure, and \(Q\) is the number of monitored points.

3.4. Procedure of the MSVM-ABC Based Method. If the MSVM model can establish the nonlinear relation between the borehole pressures and determined parameters, the model can be used to predict the borehole pressures. Then, the ABC algorithm is utilized to find the optimal parameters through error minimization between the pressures predicted by the MSVM model and the measured pressures. The back analysis flowchart is shown in Figure 1 and the algorithm is described as follows.

Step 1. Determine the general information and data such as the unknown (determined by back analysis) and known parameters of the numerical model, the MSVM and ABC algorithm parameters, and the range of parameters to be determined.

Step 2. Generate the combination of determined parameters, calculate the borehole pressure for each combination, and then build the learning samples for MSVM.

Step 3. Based on the learning samples of Step 2, construct the MSVM model using the MSVM algorithm and activate the ABC algorithm.

Step 4. Search for the optimal determined parameters based on the monitored pressure.

4. Validation and Application

To verify the proposed method, a numerical example is adopted to determine the in situ stress and elastic mechanical parameters of the elastic rock. The numerical experiment is conducted based on 2D hydraulic fracturing model of water injection into a hypothetical deep formation. Further details of the physical and numerical model can be found in the literature published by Schmitt and Zoback [1].

The parameters to be determined are the maximum and minimum horizontal in situ stress \(\sigma_{H\text{max}}\) and \(\sigma_{H\text{min}}\), respectively, Young's modulus \(E\), and Poisson’s ratio \(v\). The rock mass in all the zones is considered to be elastic. The mechanical parameters of the joints and the permeability...
of the rock mass are known; the parameter values can be seen in the literature published by Schmitt and Zoback [1]. Thirty sets of training samples and ten testing samples derived in previous studies [1, 2] were selected. Based on the MSVM algorithm, the MSVM code was written in Excel and VBA. The MSVM parameters and some of the weight $w_i$ and constant $b_i$ values and samples are shown in Figure 2. Good agreement between the measured data and the pressures estimated by the MSVM is shown in Figure 3, indicating the good performance of the MSVM model. Thus, the proposed model can accurately estimate the borehole pressures, replacing the existing numerical analysis method for calculating borehole pressures. The results also confirm that the MSVM model provides an accurate representation of the nonlinear relationship between the pressures and the determined parameters.

The ABC code is also written in Excel and VBA. The parameters of the ABC algorithm and the calculation results are shown in Figure 4. Based on the proposed method for determining the in situ stress and mechanical parameters of rock mass, the results are shown in Table 1. We obtained the values of $\sigma_{H_{\text{max}}}$, $\sigma_{h_{\text{min}}}$, $E$, and $\nu$ as 24.46 MPa, 14.33 MPa, 44.02 GPa, and 0.25, respectively. The elastic mechanical parameters of the rock agree with the results calculated by the Genetic Algorithm-Neural Network (ANN-GA) [2]. A comparison of in situ stress values calculated using four different formulations is shown in Figure 5. $\sigma_{h_{\text{min}}}$ agrees well with the value estimated by ANN-GA, (1), and (2). The relative error is only 1.07%. $\sigma_{H_{\text{max}}}$ agrees well with the value estimated by (2), which considers the pore pressure, but differs considerably from the values calculated by (1) and ANN-GA which do not consider the poroelastic effects. The relative error is up to 31.8%. Using (4), we obtain the upper and lower limits of the maximum horizontal in situ stress $\sigma_{H_{\text{max}}}$ (14.95–34.55 MPa). The value 24.46 MPa is within this range. Thus, the proposed method can be used in back analysis as an alternative numerical analysis method, which considers the poroelastic effects and provides rational, high-precision results. Note that the proposed method can determine the maximum horizontal in situ stress without estimating the poroelastic coefficient, which is a difficult parameter to obtain.
Table 1: Comparison of in situ stresses and mechanical parameters between MSVM model and other preexisted models.

<table>
<thead>
<tr>
<th>Model</th>
<th>( \sigma_H ) (MPa)</th>
<th>( \sigma_h ) (MPa)</th>
<th>Young's modulus ( E ) (GPa)</th>
<th>Poisson's ratio ( \nu )</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSVM</td>
<td>24.46</td>
<td>14.33</td>
<td>44.02</td>
<td>0.25</td>
</tr>
<tr>
<td>ANN-GA(^*)</td>
<td>35.83</td>
<td>14.968</td>
<td>39.92</td>
<td>0.26</td>
</tr>
<tr>
<td>Eq. (1)(^#)</td>
<td>34.549</td>
<td>15.637</td>
<td>-16.8006</td>
<td>-25.2559</td>
</tr>
</tbody>
</table>

\(^*\)The model proposed by Zhang and Yin [2]. \(^\#\)The model proposed by Hubbert and Willis [23].

![Figure 2: Parameters and model of MSVM in the Excel VBA platform.](image-url)
Moreover, there are four borehole pressures, namely, formation breakdown pressure (FBP) \( P_1 \), fracture propagation pressure (FPP) \( P_2 \), instantaneous shut-in pressure (ISIP) \( P_3 \), and leak–off pressure (LOP) \( P_4 \). A comparison of back analysis on borehole pressures obtained by three different methods is presented in Figure 6. The borehole pressure calculated by the proposed MSVM method is very close to the measured pressure. The relative error is less than 3%. On the other hand, the convergence processes of the algorithm and fitness variations are shown in Figures 7 and 8. Initially, the data was distributed randomly in the searching space (Figure 8) and then converged to the solution of the problem in the 500th generation. This indicates that the proposed method can determine both the in situ stress and elastic mechanical parameters of the elastic rock with excellent converging performance.

5. Conclusions

In this paper, a new borehole pressure-based back analysis approach to determine the stress and mechanical parameters of rock mass is proposed. The method combines a coupling...
the in situ stress and mechanical parameters of rock from hydraulic fracturing.

Symbols

\( \sigma_v \): Principal stress in y direction
\( \sigma_{H_{\text{max}}} \): The greatest horizontal principle stress
\( \sigma_{h_{\text{min}}} \): The smallest horizontal principal stress
\( \nu \): Poisson's ratio
\( \beta \): Poroelastic effect parameter
\( W \): The weight vector
\( \text{MSVM}(X) \): The predicted pressure using the MSVM model
\( C \): Hyper parameter that determines trade-off between the regularization and the error reduction term
\( \varepsilon \): Tolerant error
\( Q \): Number of output
\( \phi_{ij} \): A random number in the range \([-1, 1]\]
\( b \): A constant for classification threshold
\( P_b \): Breakdown pressure
\( P_p \): Pore pressure
\( E \): Young's modulus
\( \alpha \): Biot poroelastic parameter
\( T \): Rock tensile strength
\( N \): Number of input
\( \tau \): Sum of constant terms that do not depend on either \( W \) or \( b \)
\( \text{rand}(0, 1) \): A random number between \([0, 1]\)
\( \text{fitness} \): Fitness value of the solution.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors gratefully acknowledge the National Natural Science Foundation of China (Grant no. 41372315), the State Key Research Development Program of China (Grant...
no. 2016YFC0600702), and Innovative Research Team (in Science and Technology) in University of Henan Province (no. 15IRTSTHN029).

References


Submit your manuscripts at https://www.hindawi.com