Hindawi
Mathematical Problems in Engineering
Volume 2017, Article ID 6401835, 11 pages
https://doi.org/10.1155/2017/6401835

Research Article

Calibrating the Micromechanical Parameters of the PFC2D(3D) Models Using the Improved Simulated Annealing Algorithm

Min Wang[1,2] and Ping Cao[1]

[1]*School of Resources and Safety Engineering, Central South University, Changsha, Hunan 410083, China*
[2]*Hunan Provincial Key Laboratory of Shale Gas Resource Utilization, Hunan University of Science and Technology, Xiangtan, Hunan 411201, China*

Correspondence should be addressed to Min Wang; michaelwong307@outlook.com

Received 15 January 2017; Revised 17 March 2017; Accepted 10 April 2017; Published 26 April 2017

Academic Editor: Yakov Strelniker

PFC2D(3D) is commercial software, which is commonly used to model the crack initiation of rock and rock-like materials. For the PFC2D(3D) numerical simulation, a proper set of microparameters need to be determined before the numerical simulation. To obtain a proper set of microparameters for PFC2D(3D) model based on the macroparameters obtained from physical experiments, a novel technique has been carried out in this paper. The improved simulated annealing algorithm was employed to calibrate the microparameters of the numerical simulation model of PFC2D(3D). A Python script completely controls the calibration process, which can terminate automatically based on a termination criterion. The microparameter calibration process is not based on establishing the relationship between microparameters and macroparameters; instead, the microparameters are calibrated according to the improved simulated annealing algorithm. By using the proposed approach, the microparameters of both the contact-bond model and parallel-bond model in PFC2D(3D) can be determined. To verify the validity of calibrating the microparameters of PFC2D(3D) via the improved simulated annealing algorithm, some examples were selected from the literature. The corresponding numerical simulations were performed, and the numerical simulation results indicated that the proposed method is reliable for calibrating the microparameters of PFC2D(3D) model.

1. Introduction

The discrete element method (DEM) was firstly proposed by Cundall in 1971 [1]. The DEM was investigated further over the following years [2–7]. It has been widely employed in modeling the damage and nonlinear behaviors of materials. The particle flow code in 2 or 3 dimensions (PFC2D(3D)) models the movement and interaction of circular (2D) or spherical (3D) particles using the DEM. It has many advantages [8–10]: It is efficient, as contact detection between circular objects is much simpler than contact detection between angular objects, it is possible for the blocks to break (because they are composed of bonded particles), among others. Thanks to these advantages of the PFC2D(3D), PFC2D(3D) is extensively utilized to solve rock mechanics and rock engineering problems [11–23]. However, the software also has its disadvantages: it requires calibration. In other words, some microparameters must be specified to result in a material with desired macroparameters such as the uniaxial compressive strength (UCS), Young's modulus, Poisson's ratio, and tensile strength.

The relationship between the microparameters and the macroparameters is difficult to quantify and the microparameters cannot be directly determined according to the macroparameters obtained from the physical experiments. In practice, however, the microparameters of a numerical simulation model in PFC2D(3D) can be calibrated based on the macroparameters determined by the physical experiments, for example, UCS, Poisson's ratio, Young's modulus, and tensile strength. According to the difference between the macroparameters obtained from the physical experiments and the numerical simulation, the microparameters are calibrated until the macroparameters obtained from the numerical simulation are sufficiently closed to those from

physical experiments. This calibration procedure is called the "trial and error" method [24]. While the drawbacks of the "trial and error" method are obvious, on the one hand, the calibration procedure is subjective, as it depends on the experiments and the experimenter; if the experimenter is not experienced at calibrating the microparameters, then the calibration process will take a very long time. On the other hand, calibrating the microparameters means modifying the microparameters in the command flow .txt file of the numerical simulations for each step of calibration, which would be hard work. In summary, calibrating the microparameters via the "trial and error" method does not illustrate how to calibrate the microparameters of PFC2D(3D) specifically, as it depends on the experiments or the experimenter. Despite these drawbacks of the "trial and error" approach, the method has been commonly adopted by many researchers [25–33] primarily because there is no other better way to determine the microparameters of PFC2D(3D) models.

To avoid the subjectivity in the process of calibrating the microparameters, Yoon [34] carried out a new approach for calibrating the microparameters of contact-bond models in PFC2D. The relationships between microparameters and UCS, Young's modulus, and Poisson's ratio were constructed. By combining the numerical simulation results, Plackett-Burman designed a central composite method. The optimum set of microparameters were determined, and the macroparameters obtained from the numerical simulation results were in good agreement with the laboratory results, whereas the interaction of different microparameters is not considered in the method presented by Yoon [34], and the approach can only be used in the contact-bond model of PFC2D. Additionally, the method can only determine the microparameters of rock materials with their physical properties falling within the following ranges: UCS (40–170 MPa), Young's modulus (20–50 GPa), and Poisson's ratio (0.19–0.25). In summary, the approach proposed by Yoon [34] has a limited application. Tawadrous et al. [35] conducted a large number of PFC3D numerical simulations. By combining the numerical simulation results and artificial neural networks, the microparameters of parallel-bond of PFC3D can be predicted. However, the numbers of determined microparameters are limited; namely, the parallel-bond and particle elastic modulus, normal-to-shear stiffness ratio, and parallel-bond strength can be determined by utilizing this method. Moreover, the key for the success of artificial neural networks is sufficient training data [36–38], which implies that a large number of numerical simulations should be conducted using this method. Its applicability to the other models of PFC2D(3D) also requires further investigation.

According to the references given above, the difficulty of screening out a proper set of microparameters can be classified into three categories: (1) The subjectivity of calibrating the microparameters: the calibration process should be more objective and should not depend on the experiments or the experimenter; (2) the hard work of calibrating the microparameters: during the calibration process, the microparameters must be changed by hand for each calibration step, which is time-consuming and tedious; (3) the limited use of the microparameters calibration method: the calibration method should be applied to different kinds of numerical models in both PFC2D and PFC3D software.

For convenience of singling out a proper set of microparameters of PFC2D(3D) based on some basic experimental macroparameters (UCS, Young's modules, Poisson's ratio, tensile strength, etc.), a new approach for calibrating microparameters is proposed in this paper. The method is based on the improved simulated annealing algorithm. In addition, Python scripts were developed to accomplish the calibration process automatically. The main merit of the proposed method is decreasing the difficulty of calibrating microparameters in calibration process. Additionally, it avoids the subjectivity in calibrating microparameters, and it can be applied to calibrate the microparameters of contact-bond materials and parallel-bond models in both PFC2D and PFC3D. Additionally, the numbers of microparameters and macroparameters can be increased or decreased according to the specific circumstances in the presented method, which is quite flexible in practical use.

2. Calibration Process via the Improved Simulated Annealing Algorithm

2.1. Introduction of the Simulated Annealing Algorithm. The simulated annealing algorithm was a stochastic search method that was first carried out by Metropolis et al. [46], and, then, the simulated annealing algorithm was successfully applied to solving the optimization problems by Kirkpatrick et al. [47]. The simulated annealing algorithm is analogous to the annealing process of materials. Boltzmann [48] reckoned that if a system was in thermal equilibrium at a temperature T, the probability $P_T(s)$ of the system being in a given state s could be expressed as follows:

$$P_T(s) = \frac{\exp(-E(s)/kT)}{\sum_{s\in S}\exp(-E(s)/kT)}, \tag{1}$$

where $E(s)$ is the energy of the state s, k is the Boltzmann constant (cooling coefficient), and S is the set of all the possible states. However, (1) does not give any information on how the material reaches thermal equilibrium at a given temperature. Metropolis et al. [46] proposed an algorithm that simulated the process described by Boltzmann: when the system is in the original state s_{old} with original energy $E(s_{\text{old}})$, a random neighborhood state s_{new} is selected, which leads to a new energy $E(s_{\text{new}})$. Based on the Metropolis criterion, if $E(s_{\text{new}}) \le E(s_{\text{old}})$, then the new state is accepted. If $E(s_{\text{new}}) > E(s_{\text{old}})$, then the probability of accepting the new state can be written as follows:

$$P_{\text{accept}} = \exp\left(-\frac{E(s_{\text{old}}) - E(s_{\text{new}})}{kT}\right). \tag{2}$$

To reach the thermal equilibrium completely, the process will be repeated *Markov* times at each temperature, and *Markov* is thus called the Markov chain [49]. For better understanding of the simulated annealing algorithm, its flowchart is illustrated in Figure 1.

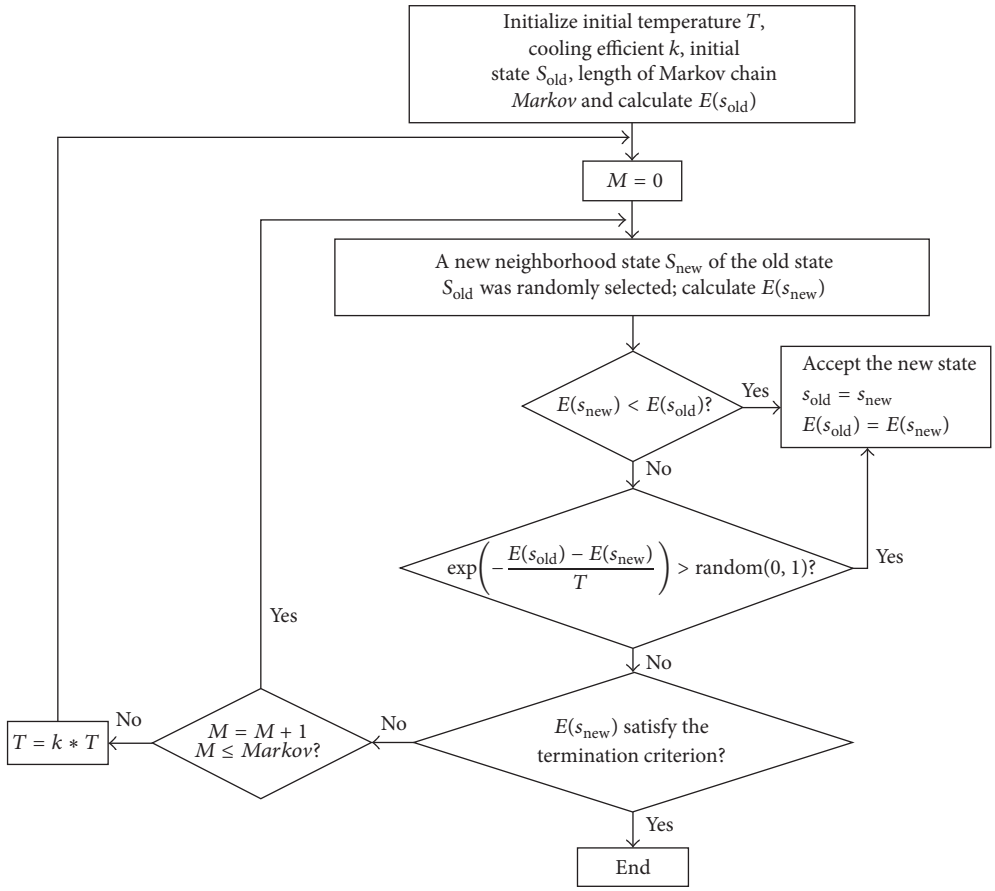

Figure 1: Flowchart of the simulated annealing algorithm.

In recent years, the simulated annealing algorithm has been widely applied in the field of optimization [50–58]. The process of calibrating microparameters can be described as minimizing the difference between the macroparameters obtained from the numerical simulations and the physical experiments, which is a typical optimization problem: the calibration process is to find the minimum value of the difference between the macroparameters obtained from the numerical simulation and the physical experiments, and the corresponding set of microparameters are the proper one. Accordingly, the calibration process is transformed into an optimization problem. However, the simulated annealing algorithm should be improved to minimize the total iterations, which will be given in the following section.

2.2. Disadvantages of the Simulated Annealing Algorithm. The simulated annealing algorithm has a drawback: the computation times are large [59, 60], which is mainly caused by the characteristics of the simulated annealing algorithm itself. There are four conditions for the success of the simulated annealing algorithm which are as follows [61–65]: (1) The initial temperature must be high enough; (2) the cooling speed for the temperature must be slow enough; (3) for each temperature, the length of the Markov chain must be long enough; (4) the final temperature must be low enough. These four conditions increase the total computation times directly or indirectly, which results in the slow convergence speed. Thereafter, it is necessary to improve the simulated annealing algorithm so as to decrease the total iterations. To improve the simulated annealing algorithm, some procedures of the simulated annealing algorithm were changed, which will be explained explicitly in the following part.

2.3. Improved Simulated Annealing Algorithm. In this paper, the simulated annealing algorithm would be improved from two aspects: the disturbance method and the cooling method.

2.3.1. The Change of the Cooling Temperature Method. For the original simulated annealing algorithm, the cooling process can be expressed as follows:

$$T = kT_0, \tag{3}$$

where T_0 is the initial temperature, k is the cooling efficient, and T is the cooling temperature.

According to Ingber [66–68], the convergence speed of the simulated annealing algorithm will increase when the following cooling strategy is adopted:

$$T = T_0 \exp\left(-Ci^{1/N}\right), \tag{4}$$

where i is the iteration time, C is a constant, and N is the number of input parameters in the simulated annealing algorithm.

In practice use, (4) can be simplified as follows:

$$T = T_0 \beta^{i^{1/N}}, \tag{5}$$

where β is a constant, which is falling within $0.7 < \beta < 1$. In this paper, β is set to 0.99.

2.3.2. The Change of the Disturbance Method for the Variables. In the traditional simulated annealing algorithm, the disturbance for the variable can be denoted as follows:

$$v_{j\text{-new}} = v_{j\text{-old}} + \left(B_{v_j\text{-upper}} - B_{v_j\text{-lower}}\right)\left(u_{\text{normal}} - 0.5\right), \tag{6}$$

where $v_{j\text{-new}}$ is the new generated value for the variable v_j, $v_{j\text{-old}}$ is the original value of v_j, $B_{v_j\text{-upper}}$ is the upper bound of the variable v_j, $B_{v_j\text{-lower}}$ is the lower bound of the variable v_j, and u_{normal} is a random number between 0 and 1, which is subjected to the normal distribution.

Based on the study by Ingber [66, 69, 70], the Cauchy random distribution function for the disturbance is conducive for the convergence of the simulated annealing algorithm:

$$v_{j\text{-new}} = v_{j\text{-old}} + \left(B_{v_j\text{-upper}} - B_{v_j\text{-lower}}\right)\left(u_{\text{Cauchy}} - 0.5\right), \tag{7}$$

where u_{Cauchy} is the random number between 0 and 1 and the generated value u_{Cauchy} is subjected to the Cauchy distribution.

2.4. Calibrating Microparameters via the Improved Simulated Annealing Algorithm. Before using the improved simulated annealing algorithm for calibrating the microparameters, some changes are required. After the PFC2D(3D) numerical simulations based on the set of microparameters, the corresponding macroparameters can be obtained. In the calibration process, the macroparameters determined by numerical simulations are defined as the state of the system. To implement the microparameters calibration process via the improved simulated annealing algorithm, the energy function (objective function) $E(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}})$ calculated by the macroparameters of the numerical simulation is denoted as follows:

$$\begin{aligned} &E\left(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}}\right) \\ &= \max\left[\text{abs}\left(\frac{\text{MA}_{1\text{-experiment}} - \text{MA}_{1\text{-PFC}}}{\text{MA}_{1\text{-experiment}}}\right),\right. \\ &\text{abs}\left(\frac{\text{MA}_{2\text{-experiment}} - \text{MA}_{2\text{-PFC}}}{\text{MA}_{2\text{-experiment}}}\right), \ldots, \\ &\left.\text{abs}\left(\frac{\text{MA}_{i\text{-experiment}} - \text{MA}_{i\text{-PFC}}}{\text{MA}_{i\text{-experiment}}}\right)\right]. \end{aligned} \tag{8}$$

where $\text{MA}_{i\text{-experiment}}$ is the set of macroparameters obtained from the physical experiments and $\text{MA}_{i\text{-PFC}}$ is the set of macroparameters determined by the numerical simulations. The objective function in (8) indicates that the value of $E(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}})$ is the maximum difference between the macroparameters obtained from the numerical simulation and the physical experiments. The macroparameters $(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}})$ of the numerical simulation are determined by performing the numerical simulations by using the microparameters $(\text{MI}_1, \text{MI}_2, \ldots, \text{MI}_j)$. It should be noted that the numbers of macroparameters and microparameters can be increased or decreased according to the specific circumstances, which is one of the advantages of the proposed technique.

To search for the microparameters completely within the determined range, the length of the Markov chain is utilized. The length of the Markov chain represents the iteration times at each temperature. With increasing length of the Markov chain, the corresponding iteration time will increase as well; in this paper, the length of the Markov chain is set to 100.

Another important factor is the bounding of the microparameters. The new microparameters should be generated within a certain range during the calibration, or the numerical simulations of PFC2D(3D) cannot proceed successfully. For example, all the microparameters in PFC2D(3D) should be larger than 0, or the numerical simulations cannot run successfully.

In this paper, the termination criterion of the calibration process is $E(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}}) < 10\%$, meaning that the maximum error between the macroparameters obtained from the numerical simulations and the physical experiments is less than 10%.

By combining the improved simulated annealing algorithm and some basic parameters for the calibration process, the procedure of calibrating microparameters via the improved simulated annealing algorithm can be described step-by-step as follows.

Step 1. The initial temperature T, the cooling efficient k, the length of Markov chain *Markov*, the numbers of the microparameters N, and the range of microparameters are initialized, and the initial iteration time i is set to 1. A set of microparameters $(\text{MI}_1, \text{MI}_2, \ldots, \text{MI}_j)_{\text{old}}$ is randomly generated; the microparameters should be within the determined range. By using the microparameters $(\text{MI}_1, \text{MI}_2, \ldots, \text{MI}_j)_{\text{old}}$, PFC2D or PFC3D numerical simulations are conducted to obtain the macroparameters $(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}})_{\text{old}}$. $E(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}})_{\text{old}}$ is estimated based on (8).

Step 2. If $E(\text{MA}_{1\text{-PFC}}, \text{MA}_{2\text{-PFC}}, \ldots, \text{MA}_{i\text{-PFC}})_{\text{old}} < 10\%$, then go to Step 10.

Step 3. $M = 0$.

Step 4. A new set of random neighborhood microparameters $(\text{MI}_1, \text{MI}_2, \ldots, \text{MI}_j)_{\text{new}}$ near the microparameters $(\text{MI}_1, \text{MI}_2, \ldots, \text{MI}_j)_{\text{old}}$ is generated; the new generated microparameters should be within the determined range. Taking

the microparameter MI_j as example, the new generated microparameter $\mathrm{MI}_{j\text{-new}}$ can be expressed as follows:

$$\mathrm{MI}_{j\text{-new}} = \mathrm{MI}_{j\text{-old}} + \left(B_{j\text{-upper}} - B_{j\text{-lower}}\right) * \left[u_{\text{Cauchy}} - 0.5\right], \quad B_{j\text{-lower}} < \mathrm{MI}_{j\text{-new}} < B_{j\text{-upper}}, \tag{9}$$

where $\mathrm{MI}_{j\text{-new}}$ is the newly generated microparameter, $\mathrm{MI}_{j\text{-old}}$ is the original microparameter, $B_{j\text{-upper}}$ is the upper bound of the microparameter MI_j, $B_{j\text{-lower}}$ is the lower bound of the microparameter MI_j, and u_{Cauchy} represents a random value between 0 and 1; besides, the generated value is subjected to the Cauchy distribution. The macroparameters $(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{new}}$ are determined by the PFC2D or PFC3D numerical simulations by using the microparameters $(\mathrm{MI}_1, \mathrm{MI}_2, \ldots, \mathrm{MI}_j)_{\text{new}}$. $E(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{new}}$ is calculated based on (9).

Step 5. If $E(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{new}} < E(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{old}}$, then the new microparameters $(\mathrm{MI}_1, \mathrm{MI}_2, \ldots, \mathrm{MI}_j)_{\text{new}}$ and $E(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{new}}$ are accepted, while if $E(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{new}} \geq E(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{old}}$, then the probability of accepting the new microparameters $(\mathrm{MI}_1, \mathrm{MI}_2, \ldots, \mathrm{MI}_j)_{\text{new}}$ and $E(\mathrm{MI}_1, \mathrm{MI}_2, \ldots, \mathrm{MI}_j)_{\text{new}}$ can be expressed as follows:

$$P_{\text{accept}} = \exp\left(-\frac{E\left(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}}\right)_{\text{old}} - E\left(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}}\right)_{\text{new}}}{T}\right). \tag{10}$$

Step 6. If $E(\mathrm{MA}_{1\text{-PFC}}, \mathrm{MA}_{2\text{-PFC}}, \ldots, \mathrm{MA}_{i\text{-PFC}})_{\text{new}} < 10\%$, then go to Step 10.

Step 7. $i = i + 1$.

Step 8. $M = M + 1$; if $M \leq Markov$, then go to Step 4.

Step 9. $T = 0.99^{i^{1/N}} T$; go to Step 3.

Step 10. End.

Because the microparameters need to be changed and the PFC2D(3D) software needs to be manipulated many times for each step of calibration, the process of calibrating microparameters is accomplished using Python script. Python (https://www.python.org) is a scripting language that is very easy to use and widely applied in many fields [71–76].

The numerical simulation mentioned in this section includes four parts: (1) The command flow .txt file based on the microparameters is written; (2) the particle assembly is carried out; (3) the numerical simulation is performed to obtain the macroparameters; (4) the output of the macroparameters is produced.

2.4.1. Writing the Command Flow .txt File Based on the Microparameters. In practice, the command flow is frequently complied in a .txt file, which will be convenient for the PFC2D(3D) numerical simulations. Due to the change in the microparameters for each step of calibration, the command flow .txt file is written by the Python scripts, and the microparameters for each calibration are changed according to the improved simulated annealing algorithm.

2.4.2. The Particle Assembly. Before the numerical simulation, a PFC2D(3D) assembly is generated, and the process involves particle generation, packing the particles, isotropic stress installation (stress initialization), floating particle elimination, and bond installation, which is described in detail by Itasca [9, 77] and will not be repeated in this paper.

2.4.3. The Numerical Simulation to Obtain the Macroparameters. To obtain the macroparameters, the corresponding numerical simulations should be carried out. The UCS, Young's modulus, and Poisson's ratio can be obtained by the uniaxial compression test. Meanwhile, the tensile strength can be determined via the Brazilian disc test. Other numerical simulations can also be utilized to obtain the corresponding macroparameters. In practice, the numbers of macroparameters and microparameters can be increased or decreased according to the specific circumstances. The corresponding command flow .txt file should be changed accordingly.

2.4.4. The Output of the Macroparameters. The aim of the PFC2D(3D) numerical simulations is to obtain the numerical macroparameters. Thereafter, the macroparameters will be output in the log file when the numerical simulations terminate [9, 10].

3. Examples of Calibrating Microparameters via the Improved Simulated Annealing Algorithm

3.1. A Detailed Example

3.1.1. Basic Information about the Numerical Simulations. To illustrate the calibration process of microparameters more specifically, an example was selected. In [78], sandstone was utilized to investigate the Kaiser effect. Moreover, the PFC2D was employed to reproduce the Kaiser effect, the deformation, and the rate of deformation. Meanwhile, the contact-bond model in PFC2D was utilized, and the size of the numerical simulation for the macroparameters was 61.14 mm × 153.32 mm. According to the physical experiments, the UCS and Young's modulus of the sandstone were 83.1 MPa and 14.3 GPa, respectively.

Table 1: Microparameters of a contact-bonded material.

Parameter	Description
$R_{\min}$	Minimum ball radius [m]
$R_{\max}/R_{\min}$	Ball size ratio, uniform distribution [ ]
ρ	Ball density [kg/m^3]
E_c	Ball-ball contact modulus [Pa]
k_n/k_s	Ball stiffness ratio [ ]
μ	Ball friction coefficient
σ_c (mean)	Contact-bond normal strength, mean [Pa]
σ_c (std. dev.)	Contact-bond normal strength, std. dev. [Pa]
τ_c (mean)	Contact-bond shear strength, mean [Pa]
τ_c (std. dev.)	Contact-bond shear strength, std. dev. [Pa]

In total, 10 microparameters determine the contact-bonded model in PFC2D. The microparameters and their corresponding descriptions are listed in Table 1.

As listed in Table 1, the density of the ball can be treated as the density of the sandstone, which is 2397 kg/m^3 [78]. The effect of gravity does not need to be considered, as the specimens are small in the physical experiments and the gravity-induced stress has a negligible effect on the macroparameters [77]. In addition, the microparameters should be in a certain range, which guarantees the success of the numerical simulations. For example, $R_{\max}/R_{\min}$, the ball size ratio, should be greater than 1, or the numerical simulation cannot run successfully. Therefore, it is necessary to determine a reasonable range of microparameters. Table 2 gives the range of the microparameters of the contact-bonded model in the numerical simulations.

It should be noted that the minimum ball radius should be larger than $0.1e-3$ m, or the error "*Error: Fewer balls generated*" will occur during the numerical simulations, which will stop the numerical simulation from running. The upper bound of the microparameters in Table 2 can be increased slightly, but this will increase the computation times.

3.1.2. Basic Parameters of the Improved Simulated Annealing Algorithm. Some basic parameters of the improved simulated annealing algorithm need to be determined before the numerical simulations; these are listed in Table 3.

3.1.3. Numerical Simulation Platform. In the numerical simulation, the operation system is Linux Mint 18 Sarah (on a personal computer with an Intel Core i5, 3.30 GHz CPU, and 8 GB RAM); the operation system is open source. The Python version is 3.5. Additionally, to implement the improved simulated annealing algorithm and manipulate the PFC2D(3D) software, two extra Python packages were installed, *numpy* (to accomplish the improved simulated annealing algorithm) and *subprocess* (to manipulate the PFC2D(3D) software). The PFC2D version 3.1 in this paper was adapted to run on the Linux system. *Wine*, an open source software for running the Windows system application, was used for this purpose.

3.1.4. Numerical Simulations Results. Based on the improved simulated annealing algorithm, after 2739 iterations, the termination criterion is satisfied and the numerical simulation is terminated. The UCS and Young's modulus determined by the numerical simulation are 74.356 MPa and 14.824 GPa, respectively, which is quite close to the physical experiment results (UCS 82.1 MPa and Young's modulus 14.3 GPa.). The corresponding microparameters are obtained and are listed in Table 4.

The microparameters can successfully be determined by the improved simulated annealing algorithm. This saves a tremendous amount of hard work, as it is not necessary to change the microparameters and manipulate PFC2D by hand for each step of calibration, which is convenient for use.

3.2. More Examples. From the analysis of the detailed example in Section 3.1, it is found that the calibration method is independent of the numerical simulation model; that is, once the command flow for numerical simulations is fixed, the calibration process can change the microparameters in the command flow .txt file for each step of calibration. Thus, it can be used for contact bond and parallel bond in both PFC2D and PFC3D. Recall that the numbers of microparameters and macroparameters can be increased or decreased according to the specific circumstances, in which case the corresponding Python scripts should also be changed correspondingly. To further validate the proposed method, more numerical simulation tests were performed. The numerical simulation results are listed in Table 5.

As listed in Table 5, calibrating the microparameters via the improved simulated annealing algorithm is a viable strategy for both parallel bond and contact bond in PFC2D(3D). It has wide practical use in PFC2D(3D) numerical simulations.

4. Discussion

PFC2D(3D) is a typical DEM, which is widely applied to model the damage and nonlinear behaviors of rock materials. However, the microparameters in PFC2D(3D) need to be determined before the numerical simulations. In theory, the microparameters can be calibrated on the basis of macroparameters determined by the physical experiments. In practice, determining the proper set of microparameters based on the macroparameters obtained from the physical experiments is very difficult but is critically important for the success of the PFC2D(3D) numerical simulations.

The microparameters are mainly calibrated based on the macroparameters determined by the physical experiments. In recent years, the most commonly used method for calibrating microparameters in PFC2D(3D) is called the "trial and error" method [24]. Its main process for calibrating the microparameters is comparing the macroparameters determined by the numerical simulation and the physical experiments; then, the microparameters are changed according to the difference between the macroparameters of the numerical simulation and the physical experiments. However, the calibration process is quite subjective; it is influenced by the experiment and experimenter significantly. In principle, this means that if a user is familiar with the influence of microparameters on the macroparameters, it will not take very long. In contrast, if a user is inexperienced, the calibration process will be lengthy. Another significant

Table 2: The range of the microparameters of the contact-bonded material.

Parameter	Description	Range
R_{min}	Minimum ball radius [m]	$0.1e-3$~$1e-3$
R_{max}/R_{min}	Ball size ratio, uniform distribution [ ]	1~20
E_c	Ball-ball contact modulus [Pa]	0~$100e9$
k_n/k_s	Ball stiffness ratio [ ]	0~10
μ	Ball friction coefficient	0~1
σ_c (mean)	Contact-bond normal strength, mean [Pa]	0~$1000e6$
σ_c (std. dev.)	Contact-bond normal strength, std. dev. [Pa]	0~$100e6$
τ_c (mean)	Contact-bond shear strength, mean [Pa]	0~$1000e6$
τ_c (std. dev.)	Contact-bond shear strength, std. dev. [Pa]	0~$100e6$

Table 3: Basic parameters of the improved simulated annealing algorithm.

Parameter	Value
The length of the Markov chain	100
Cooling efficient k	0.95
The initial temperature T	100
The number of microparameters N	9

Table 4: Microparameters obtained by the improved simulated annealing algorithm.

Parameter	Value
R_{min}	$0.746e-3$ m
R_{max}/R_{min}	1.567
E_c	20.688 GPa
k_n/k_s	1.833
μ	0.590
σ_c (mean)	88.916 MPa
σ_c (std. dev.)	36.950 MPa
τ_c (mean)	83.768 MPa
τ_c (std. dev.)	28.932 MPa

issue is that this calibration process is accomplished by hand. In the calibration process, each step of calibrating microparameters requires a series of repeated work: changing the microparameters in the command flow .txt file, opening the PFC2D(3D) software, and calling the command flow .txt file. With increasing iterations, it takes much time and work. Therefore, accomplishing the calibration process automatically has great practical utility in PFC2D(3D) numerical simulations. As described above, there exist two unsolved problems in calibrating microparameters: how to avoid the subjectivity in the calibration process and how to accomplish the calibration process automatically.

In this paper, the improved simulated annealing algorithm was applied to calibrate the microparameters based on the macroparameters determined from the physical experiments, and the calibration process was completely accomplished by Python scripts (the corresponding Python scripts in this paper will be published on GitHub (https://github.com/)). The improved simulated annealing algorithm was mainly utilized to obtain the proper set of microparameters. By using this method, the microparameters are changed on the basis of the improved simulated annealing algorithm; when the macroparameters of the numerical simulations are quite close to the macroparameters obtained from the physical experiments, the calibration process terminated. The calibration process was accomplished by Python scripts, eliminating the necessity of human input at each step. Additionally, this method can be applied to different kinds of the bond models in PFC2D and PFC3D. Consequently, the method proposed in this paper has a practical use in PFC2D(3D) numerical simulations.

However, the calibration process is realized in the Linux operation system in this paper; meanwhile, the PFC2D(3D) are mainly used in the Windows operation system. Thereafter, whether the complying Python scripts are suitable in the Windows operation system needs to be further investigated and will be our next work.

5. Conclusions

To determine a proper set of microparameters based on the macroparameters obtained by physical experiments, the improved simulated annealing algorithm was utilized to calibrate the microparameters of the PFC2D(3D) models. Moreover, the Python scripts were developed to accomplish the calibration process successfully. The main conclusions of this paper are as follows.

(1) The improved simulated algorithm was utilized to calibrate the microparameters based on the macroparameters obtained from the physical experiments, avoiding the subjectivity in the calibration process. Moreover, the Python scripts were developed to calibrate the microparameters by changing the microparameters in the command .txt file for each step of calibration based on the improved simulated annealing algorithm. This method could be implemented automatically, and the calibration program terminated when a proper set of microparameters were obtained.

(2) By using the improved simulated algorithm, a set of numerical simulations were performed. It is indicated that

Table 5: Macroparameters obtained by the improved simulated annealing algorithm.

Author	Material species	Numerical model	Macroparameters determined by the numerical simulations (error)
Chang et al. 2001 [39]	Kimachi sandstone	PFC2D Contact-bond	*Experimental results*: σ_c: 48 MPa υ: 0.27 E: 6.5 GPa *Numerical results*: σ_c: 44.372 MPa (7.558%) υ: 0.275 (1.851%) E: 6.477 GPa (0.353%)
Chang 2002 [40]	Hwangdeung granite	PFC2D Contact-bond	*Experimental results*: σ_c: 162 MPa υ: 0.28 E: 50.7 GPa *Numerical results*: σ_c: 171.357 MPa (5.775%) υ: 0.283 (1.071%) E: 48.492 GPa (4.355%)
Hong 2004 [41]	Daejeon granite	PFC2D Contact-bond	*Experimental results*: σ_c: 161.7 MPa υ: 0.22 E: 50.9 GPa *Numerical results*: σ_c: 159.172 MPa (1.563%) υ: 0.239 (8.636%) E: 46.943 GPa (7.774%)
Wang et al. 2016 [12]	Rock cored from the rock mass in the Heishan open pit mine in Chengde city, China	PFC2D Contact-bond	*Experimental results*: σ_c: 132.3 MPa E: 12.0 GPa *Numerical results*: σ_c: 130.329 MPa (1.489%) E: 11.924 GPa (0.633%)
Yang et al. 2014 [42]	Red sandstone	PFC2D Parallel-bond	*Experimental results*: σ_c: 52.98 MPa E: 14.42 GPa ε_{1c}: 5.113 *Numerical results*: σ_c: 57.832 MPa (9.158%) E: 15.490 GPa (7.420%) ε_{1c}: 4.650 (9.055%)
Zhou et al. 2016 [43]	Concrete	PFC2D Parallel-bond	*Experimental results*: σ_c: 23.3 MPa υ: 0.17 E: 37 GPa *Numerical results*: σ_c: 23.525 MPa (0.965%) υ: 0.159 (6.470%) E: 35.822 GPa (3.183%)
Yang et al. 2015 [44]	Sandstone	PFC2D Parallel-bond	*Experimental results*: σ_c: 190.80 MPa E: 35.64 GPa *Numerical results*: σ_c: 173.583 MPa (9.023%) E: 39.035 GPa (9.525%)
Fan et al. 2015 [25]	Rock-like materials	PFC3D Contact-bond	*Experimental results*: σ_c: 8.27 MPa E: 4.04 GPa *Numerical results*: σ_c: 8.947 MPa (8.186%) E: 4.228 GPa (4.653%)
Duan et al 2015 [45]	Beishan granite	PFC3D Parallel-bond	*Experimental results*: σ_c: 136.4 MPa υ: 0.266 E: 43.167 GPa *Numerical results*: σ_c: 127.017 MPa (6.879%) υ: 0.265 (0.375%) E: 42.652 GPa (1.193%)

σ_c: Uniaxial compressive strength; υ: Poisson's ratio E: Young's modulus; σ_t: tensile strength; ε_{1c}: peak axial strain.

the proposed approach can be used for the calibration of microparameters. This method can be used for the contact-bond or parallel-bond models in PFC2D(3D). The method has a practical value for the PFC2D(3D) numerical simulations. However, the corresponding Python scripts were developed on the Linux operation system platform, and it is necessary to realize the calibration process on the Windows operation system; it requires further investigation.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was funded by the National Natural Science Foundation of China (51174088, 51174228, 51274253, and 51474252); China Postdoctoral Science Foundation and the Postdoctoral Science Foundation of Central South University; the National Basic Research Program of China (013CB035401); the Fundamental Research Funds for the Central Universities of Central South University (2015zzts077, 2014zzts055); and the Open Research Fund Program of Hunan Provincial Key Laboratory of Shale Gas Resource Utilization, Hunan University of Science and Technology (E21527).

References

[1] P. A. Cundall, "A computer model for simulating progressive large scale movements in blocky system," in *Proceedings of the Symposium for International Society for Rock Mechanics*, L. E. D. Muller, Ed., vol. 1, pp. 8–12, Balkema A A, Rotterdam, The Netherlands, 1971.

[2] O. D. L. Strack and P. A. Cundal, "The distinct element method as a tool for research in granular," Part I. Report to the National Science Foundation, University of Minnesota, Minnesota, USA, 1978.

[3] P. A. Cundall and O. D. L. Strack, "The distinct element method a tool for research in granula media," Part II. Report to

the National Science Foundation, University of Minnesota, Minnesota, Minn, USA, 1979.

[4] P. A. Cundall and O. D. L. Strack, "A discrete numerical model for granular assembles," *Geotechnique*, vol. 29, no. 1, pp. 47–65, 1979.

[5] O. R. Walton, "Particle dynamics modeling of geological materials," Report UCRL-52915, Lawrence Livermore National Lab, 1980.

[6] C. S. Campbell and C. E. Brennen, "Computer simulation of granular shear flows," *Journal of Fluid Mechanics*, vol. 151, pp. 167–188, 1985.

[7] C. S. Campbell and C. E. Brennen, "Chute flows of granular material: some computer simulations," *Journal of Applied Mechanics*, vol. 52, no. 1, pp. 172–178, 1985.

[8] T. Wang, W. Zhou, J. Chen, X. Xiao, Y. Li, and X. Zhao, "Simulation of hydraulic fracturing using particle flow method and application in a coal mine," *International Journal of Coal Geology*, vol. 121, pp. 1–13, 2014.

[9] *PFC2D (Particle Flow Code in 2 Dimensions)*, Version 3.1, Itasca Consulting Group, Minneapolis, Minn, USA, 1999.

[10] *PFC3D (Particle Flow Code in 3 Dimensions)*, Version 3.1, Itasca Consulting Group, Minneapolis, Minn, USA, 1999.

[11] M. Zhou and E. Song, "A random virtual crack DEM model for creep behavior of rockfill based on the subcritical crack propagation theory," *Acta Geotechnica*, vol. 11, no. 4, pp. 827–847, 2016.

[12] P. Wang, T. Yang, T. Xu, M. Cai, and C. Li, "Numerical analysis on scale effect of elasticity, strength and failure patterns of jointed rock masses," *Geosciences Journal*, vol. 20, no. 4, pp. 539–549, 2016.

[13] Z. Wang, F. Jacobs, and M. Ziegler, "Experimental and DEM investigation of geogrid-soil interaction under pullout loads," *Geotextiles and Geomembranes*, vol. 44, no. 3, pp. 230–246, 2016.

[14] J. J. Oetomo, E. Vincens, F. Dedecker, and J.-C. Morel, "Modeling the 2D behavior of dry-stone retaining walls by a fully discrete element method," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 40, no. 7, pp. 1099–1120, 2016.

[15] M. Bahaaddini, P. C. Hagan, R. Mitra, and M. H. Khosravi, "Experimental and numerical study of asperity degradation in the direct shear test," *Engineering Geology*, vol. 204, pp. 41–52, 2016.

[16] S.-Q. Yang, W.-L. Tian, Y.-H. Huang, P. G. Ranjith, and Y. Ju, "An experimental and numerical study on cracking behavior of brittle sandstone containing two non-coplanar fissures under uniaxial compression," *Rock Mechanics and Rock Engineering*, vol. 49, no. 4, pp. 1497–1515, 2016.

[17] H. Haeri, "Experimental and numerical study on crack propagation in pre-cracked beam specimens under three-point bending," *Journal of Central South University*, vol. 23, no. 2, pp. 430–439, 2016.

[18] M. Oliaei and E. Manafi, "Static analysis of interaction between twin-tunnels using discrete element method (DEM)," *Scientia Iranica*, vol. 22, no. 6, pp. 1964–1971, 2015.

[19] C.-H. Lin and M.-L. Lin, "Evolution of the large landslide induced by Typhoon Morakot: a case study in the Butangbunasi River, southern Taiwan using the discrete element method," *Engineering Geology*, vol. 197, pp. 172–187, 2015.

[20] A. Turichshev and J. Hadjigeorgiou, "Experimental and numerical investigations into the strength of intact veined rock," *Rock Mechanics and Rock Engineering*, vol. 48, no. 5, pp. 1897–1912, 2015.

[21] X. Yang, P. H. S. W. Kulatilake, H. Jing, and S. Yang, "Numerical simulation of a jointed rock block mechanical behavior adjacent to an underground excavation and comparison with physical model test results," *Tunnelling and Underground Space Technology*, vol. 50, pp. 129–142, 2015.

[22] C. Khazaei, J. Hazzard, and R. Chalaturnyk, "Damage quantification of intact rocks using acoustic emission energies recorded during uniaxial compression test and discrete element modeling," *Computers and Geotechnics*, vol. 67, pp. 94–102, 2015.

[23] S. Bock and S. Prusek, "Numerical study of pressure on dams in a backfilled mining shaft based on PFC3D code," *Computers and Geotechnics*, vol. 66, pp. 230–244, 2015.

[24] D. O. Potyondy and P. A. Cundall, "A bonded-particle model for rock," *International Journal of Rock Mechanics and Mining Sciences*, vol. 41, no. 8, pp. 1329–1364, 2004.

[25] X. Fan, P. H. S. W. Kulatilake, and X. Chen, "Mechanical behavior of rock-like jointed blocks with multi-non-persistent joints under uniaxial loading: a particle mechanics approach," *Engineering Geology*, vol. 190, pp. 17–32, 2015.

[26] Q. Zhang, H.-H. Zhu, and L. Zhang, "Studying the effect of non-spherical micro-particles on Hoek-Brown strength parameter mi using numerical true triaxial compressive tests," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 39, no. 1, pp. 96–114, 2015.

[27] Z. J. Wen, X. Wang, Q. H. Li et al., "Simulation analysis on the strength and acoustic emission characteristics of jointed rock mass," *Technical Gazette*, vol. 23, no. 5, pp. 1277–1284, 2016.

[28] Y. Franco, D. Rubinstein, and I. Shmulevich, "Prediction of soil-bulldozer blade interaction using discrete element method," *Transactions of the ASABE*, vol. 50, no. 2, pp. 345–353, 2007.

[29] Y. H. Jong and C. G. Lee, "Suggested method for determining a complete set of micro-parameters quantitatively in PFC2D," *Tunnel and Underground Space*, vol. 16, no. 4, pp. 334–346, 2006.

[30] H. Lee, T. Moon, and B. C. Haimson, "Borehole breakouts induced in Arkosic sandstones and a discrete element analysis," *Rock Mechanics and Rock Engineering*, vol. 49, no. 4, pp. 1369–1388, 2016.

[31] J. A. Vallejos, K. Suzuki, A. Brzovic, and D. M. Ivars, "Application of Synthetic Rock Mass modeling to veined core-size samples," *International Journal of Rock Mechanics and Mining Sciences*, vol. 81, pp. 47–61, 2016.

[32] H. Hofmann, T. Babadagli, J. S. Yoon, A. Zang, and G. Zimmermann, "A grain based modeling study of mineralogical factors affecting strength, elastic behavior and micro fracture development during compression tests in granites," *Engineering Fracture Mechanics*, vol. 147, pp. 261–275, 2015.

[33] J. Liu, P. Cao, C.-H. Du, Z. Jiang, and J.-S. Liu, "Effects of discontinuities on penetration of TBM cutters," *Journal of Central South University*, vol. 22, no. 9, pp. 3624–3632, 2015.

[34] J. Yoon, "Application of experimental design and optimization to PFC model calibration in uniaxial compression simulation," *International Journal of Rock Mechanics and Mining Sciences*, vol. 44, no. 6, pp. 871–889, 2007.

[35] A. S. Tawadrous, D. DeGagné, M. Pierce, and D. M. Ivars, "Prediction of uniaxial compression PFC3D model micro-properties using artificial neural networks," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 33, no. 18, pp. 1953–1962, 2009.

[36] J. V. Tu, "Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes," *Journal of Clinical Epidemiology*, vol. 49, no. 11, pp. 1225–1231, 1996.

[37] M. Smith, *Neural Networks for Statistical Modeling*, Van Nostrand Reinhold, New York, NY, USA, 1993.

[38] G. E. Hinton, "Connectionist learning procedures," *Artificial Intelligence*, vol. 40, no. 1–3, pp. 185–234, 1989.

[39] S. H. Chang, M. Seto, and C. I. Lee, "Damage and fracture characteristics of Kimachi sandstone in uniaxial compression," *Geosystem Engineering*, vol. 4, no. 1, pp. 18–26, 2001.

[40] S. H. Chang, *Characterization of stress-induced damage in rock and its application on the analysis of rock damaged zone around a deep tunnel [Ph.D thesis]*, School of Civil, Urban and Geosystem Eng, Seoul National University, Seoul, South Korea, 2002.

[41] J. S. Hong, *Characteristics of creep deformation behavior of granite under uniaxial compression, [MS thesis]*, School of Civil, Urban and Geosystem Engineering, Seoul National University, 2004.

[42] S.-Q. Yang, Y.-H. Huang, H.-W. Jing, and X.-R. Liu, "Discrete element modeling on fracture coalescence behavior of red sandstone containing two unparallel fissures under uniaxial compression," *Engineering Geology*, vol. 178, pp. 28–48, 2014.

[43] S. Zhou, H. Zhu, Z. Yan, J. W. Ju, and L. Zhang, "A micromechanical study of the breakage mechanism of microcapsules in concrete using PFC2D," *Construction and Building Materials*, vol. 115, pp. 452–463, 2016.

[44] S.-Q. Yang, Y.-H. Huang, P. G. Ranjith, Y.-Y. Jiao, and J. Ji, "Discrete element modeling on the crack evolution behavior of brittle sandstone containing three fissures under uniaxial compression," *Acta Mechanica Sinica*, vol. 31, no. 6, pp. 871–889, 2015.

[45] K. Duan, C. Y. Kwok, and L. G. Tham, "Micromechanical analysis of the failure process of brittle rock," *International Journal for Numerical and Analytical Methods in Geomechanics*, vol. 39, no. 6, pp. 618–634, 2015.

[46] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[47] S. Kirkpatrick, J. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[48] C. Cercignani, *The Boltzmann Equation and Its Applications*, Springer-Verlag, Berlin, Germany, 1988.

[49] R. K. Puri and J. Aichelin, "Simulated annealing clusterization algorithm for studying the multifragmentation," *Journal of Computational Physics*, vol. 162, no. 1, pp. 245–266, 2000.

[50] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[51] A. D. Mackerell Jr., M. Feig, and C. L. Brooks III, "Extending the treatment of backbone energetics in protein force fields: limitations of gas-phase quantum mechanics in reproducing protein conformational distributions in molecular dynamics simulation," *Journal of Computational Chemistry*, vol. 25, no. 11, pp. 1400–1415, 2004.

[52] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, no. 2, pp. 60–68, 2001.

[53] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *BioSystems*, vol. 43, no. 2, pp. 73–81, 1997.

[54] T. D. Braun, H. J. Siegel, N. Beck et al., "A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems," *Journal of Parallel and Distributed Computing*, vol. 61, no. 6, pp. 810–837, 2001.

[55] M. Sambridge, "Geophysical inversion with a neighbourhood algorithm—I. Searching a parameter space," *Geophysical Journal International*, vol. 138, no. 2, pp. 479–494, 1999.

[56] M. V. Petoukhov and D. I. Svergun, "Global rigid body modeling of macromolecular complexes against small-angle scattering data," *Biophysical Journal*, vol. 89, no. 2, pp. 1237–1250, 2005.

[57] S. Maritorena, D. A. Siegel, and A. R. Peterson, "Optimization of a semianalytical ocean color model for global-scale applications," *Applied Optics*, vol. 41, no. 15, pp. 2705–2714, 2002.

[58] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, "A simulated annealing-based multiobjective optimization algorithm: AMOSA," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 3, pp. 269–283, 2008.

[59] S.-F. Hwang and R.-S. He, "Improving real-parameter genetic algorithm with simulated annealing for engineering problems," *Advances in Engineering Software*, vol. 37, no. 6, pp. 406–418, 2006.

[60] I.-K. Jeong and J.-J. Lee, "Adaptive simulated annealing genetic algorithm for system identification," *Engineering Applications of Artificial Intelligence*, vol. 9, no. 5, pp. 523–532, 1996.

[61] S. Mahmoodpour and M. Masihi, "An improved simulated annealing algorithm in fracture network modeling," *Journal of Natural Gas Science and Engineering*, vol. 33, pp. 538–550, 2016.

[62] T. Zhang, Z. Chen, Y. Zheng, and J. Chen, "An improved simulated annealing algorithm for bilevel multiobjective programming problems with application," *Journal of Nonlinear Sciences and Applications*, vol. 9, no. 6, pp. 3672–3685, 2016.

[63] G. Arbelaez Garces, A. Rakotondranaivo, and E. Bonjour, "Improving users' product acceptability: an approach based on Bayesian networks and a simulated annealing algorithm," *International Journal of Production Research*, vol. 54, no. 17, pp. 5151–5168, 2016.

[64] M. Dai, D. Tang, A. Giret, M. A. Salido, and W. D. Li, "Energy-efficient scheduling for a flexible flow shop using an improved genetic-simulated annealing algorithm," *Robotics and Computer-Integrated Manufacturing*, vol. 29, no. 5, pp. 418–429, 2013.

[65] X.-G. Li and X. Wei, "An improved genetic algorithm-simulated annealing hybrid algorithm for the optimization of multiple reservoirs," *Water Resources Management*, vol. 22, no. 8, pp. 1031–1049, 2008.

[66] L. Ingber, "Very fast simulated re-annealing," *Mathematical and Computer Modelling*, vol. 12, no. 8, pp. 967–973, 1989.

[67] M. Jenkinson and S. Smith, "A global optimisation method for robust affine registration of brain images," *Medical Image Analysis*, vol. 5, no. 2, pp. 143–156, 2001.

[68] W. Huyer and A. Neumaier, "Global optimization by multilevel coordinate search," *Journal of Global Optimization*, vol. 14, no. 4, pp. 331–355, 1999.

[69] B. White, J. Lepreau, L. Stoller et al., "An integrated experimental environment for distributed systems and networks," *ACM SIGOPS Operating Systems Review*, vol. 36, supplement 1, pp. 255–270, 2002.

[70] G. Rudolph, "Local convergence rates of simple evolutionary algorithms with cauchy mutations," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 4, pp. 249–258, 1997.

[71] P. D. Adams, P. V. Afonine, G. Bunkóczi et al., "PHENIX: a comprehensive Python-based system for macromolecular structure solution," *Acta Crystallographica D: Biological Crystallography*, vol. 66, part 2, pp. 213–221, 2010.

[72] P. Fabian, V. Gaeel, G. Alexandre, and et al., "Scikit-learn: machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[73] M. F. Sanner, "Python: a programming language for software integration and development," *Journal of Molecular Graphics and Modelling*, vol. 17, no. 1, pp. 57–61, 1999.

[74] A. Simon, P. P. Theodor, and W. Huber, "HTSeq—a Python framework to work with high-throughput sequencing data," *Bioinformatics*, vol. 31, no. 2, pp. 166–169, 2015.

[75] T. E. Oliphant, "Python for scientific computing," *Computing in Science and Engineering*, vol. 9, no. 3, pp. 10–20, 2007.

[76] J. W. Peirce, "PsychoPy-Psychophysics software in Python," *Journal of Neuroscience Methods*, vol. 162, no. 1-2, pp. 8–13, 2007.

[77] A. Ghazvinian, V. Sarfarazi, W. Schubert, and M. Blumel, "A study of the failure mechanism of planar non-persistent open joints using PFC2D," *Rock Mechanics and Rock Engineering*, vol. 45, no. 5, pp. 677–693, 2012.

[78] S. P. Hunt, A. G. Meyers, and V. Louchnikov, "Modelling the Kaiser effect and deformation rate analysis in sandstone using the discrete element method," *Computers and Geotechnics*, vol. 30, no. 7, pp. 611–621, 2003.