Advanced Powder Technology 29 (2018) 1393–1403

Contents lists available at ScienceDirect

Advanced Powder Technology

journal homepage: www.elsevier.com/locate/apt

Original Research Paper

A calibration framework for discrete element model parameters using genetic algorithms

Huy Q. Do *, Alejandro M. Aragón, Dingena L. Schott

Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, Mekelweg 2, 2628CD Delft, The Netherlands

ARTICLE INFO

Article history:
Received 22 November 2017
Received in revised form 18 February 2018
Accepted 1 March 2018
Available online 12 March 2018

Keywords:
Parameter calibration
Discrete element method (DEM)
Genetic algorithm (GA)
Multi-objective optimization
NSGA
Inverse analysis

ABSTRACT

In this research, a universal framework for automated calibration of microscopic properties of modeled granular materials is proposed. The proposed framework aims at industrial scale applications, where optimization of the computational time step is important. It can be generally applied to all types of DEM simulation setups. It consists of three phases: data base generation, parameter optimization, and verification. In the first phase, DEM simulations are carried out on a multi-dimensional grid of sampled input parameter values to generate a database of macroscopic material responses. The database and experimental data are then used to interpolate the objective functions with respect to an arbitrary set of parameters. In the second phase, the Non-dominated Sorting Genetic Algorithm II (NSGA-II) is used to solve the calibration multi-objective optimization problem. In the third phase, the DEM simulations using the results of the calibrated input parameters are carried out to calculate the macroscopic responses that are then compared with experimental measurements for verification and validation.

The proposed calibration framework has been successfully demonstrated by a case study with two-objective optimization for the model accuracy and the simulation time. Based on the concept of Pareto dominance, the trade-off between these two conflicting objectives becomes apparent. Through verification and validation steps, the approach has proven to be successful for accurate calibration of material parameters with the optimal simulation time.

1. Introduction

The discrete element method (DEM) [1] has become a widely accepted numerical technique for computing the behavior of granular materials. However, a major barrier to effective uses of DEM is selecting appropriate input parameters so that simulations can accurately reproduce the behavior of real systems. Furthermore, efficient parameter sets to reduce computational cost are required for industrial applications. Some microscopic input parameters can be determined directly from experiments, while others can be obtained only or more practically by calibration of macroscopic responses [2–14]. Calibration is traditionally carried out by "trial and error" in which an iterative process of adjusting unknown input parameters until the DEM simulated results match the given measured bulk behavior. "Trial and error" is a purely forward, primitive methodology and is hence limited by the parametric multi-dimensionality and computational expense of performing DEM simulations.

From a mathematical point of view, DEM calibration is classified as an inverse problem [15,16]. The process aims at searching for input parameters such that the model response best matches experimental data. An inverse procedure of calibration can be divided into two steps. In the first step, DEM simulations are carried out to construct an objective function that indicates the discrepancy between the solution profile of the model and the experimentally measured profile. To reduce the number of simulations required, approaches such as design of experiment/simulation [17–19], artificial neural network training [20], and Latin hypercube sampling and Kriging [21], have been used. In the second step, optimization is used to search for the optimal parameter set that minimizes the objective function. Several different optimization methods are used for DEM calibration, *e.g.*, Levenberg-Marquardt residual minimization [21], Nelder-Mead simplex [22], weighted least squares [23], Gauss-Newton algorithm [24], and genetic algorithms [25]. In many practical cases, if the profile consists of more than one bulk property to be considered, it is formulated as a multi-objective optimization problem (MOOP). In

* Corresponding author at: 34B-4-340, Mekelweg 2, 2628CD Delft, The Netherlands.
E-mail address: q.h.do@tudelft.nl (H.Q. Do).

https://doi.org/10.1016/j.apt.2018.03.001
0921-8831/

earlier work on DEM calibration [17,21–24], MOOPs have been solved using single-objective optimization. In that work, MOOP is converted to a single objective function, with different weight factors assigned to the corresponding objectives. However, the drawbacks of using single-objective optimization techniques to solve MOOP are evident, not only because the solution obtained cannot capture the tradeoff between the different objectives, but also because the determination of the weight factors tends to substantially influence the solution. Moreover, gradient-based search techniques [21–24] may converge to a local optimum.

For more than four decades, multi-objective evolutionary genetic algorithms (MOEAs) [26–29] have been proven to be an efficient method to overcome the foregoing problems. In this paper, a DEM calibration framework is presented. In the framework, after choosing the calibration setup, an optimization process based on MOEAs is carried out. The framework offers novel aspects that have not been addressed yet in the literature on DEM calibration. The first novelty is the ability to handle any number of objective functions with a unified formulation and accommodate the discrete nature of the design parameters. Secondly, since MOEA is based on the notion of Pareto dominance, visualizations of tradeoffs among conflicting objectives provide essential information for decision making during the optimization process. Thirdly, evolutionary algorithms search for a broad portion of the decision space, and they are hence more likely to reach a set of solutions close to a global optimum.

The paper is structured as follows: Section 2 presents the novel DEM calibration framework based on MOEAs. Section 3 describes the case study for calibration composed of three DEM simulation setups that replicate numerically the experiments. A two-objective optimization problem of calibration, in which both the model error and simulation time are simultaneously minimized, is defined. In Section 4, the numerical calibration results, verification, and validation, are discussed and evaluated. Finally, Section 5 gives conclusions on the performance of the calibration framework.

2. Methods

The flowchart of the calibration approach consists of three phases, as illustrated in Fig. 1. In the first phase, calibration setups are chosen. The calibration setups should reflect the important physics aspects of final target problems, *i.e.*, material behaviors in industrial applications. The calibration setups should also be large enough in relation to particle dimensions to demonstrate realistically material behaviors as on a large-scale (industrial) case. In addition, contact force models should be chosen that properly reflects the nature of the material behaviors. DEM simulations are then carried out on a user-defined, multi-dimensional grid of sampled values of the input parameters being calibrated. The simulated macroscopic outputs and their corresponding experimental data are used as the database to interpolate the objective functions with respect to an arbitrary set of parameters. In the second phase, the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [30], one of MOEAs, is used to solve the multi-objective optimization problem. In the third phase, DEM simulations using the calibrated input parameters obtained from the optimization are carried out to calculate bulk properties that are then compared with experimental measurements for verification.

2.1. Simple genetic algorithm

Genetic algorithms (GAs) are search and optimization methods inspired by Darwin's theory of evolution of natural selection and genetics. In a simple genetic algorithm [31], an initial population of candidate solutions evolves through generations (iterations) towards individuals with better fitness by applying genetic operators such as selection, crossover, and mutation. The fitness is measured by an objective function of the optimization problem being solved. In each iteration, the better fit individuals are stochastically selected from the current population for breeding to create a new generation. The population is then replaced by a new generation and used for the next iteration of the process. The algorithm terminates when fitness values of the individuals satisfying a predefined criterion have been found, a predefined maximum number of generations has been reached, or the best fitness value has reached a plateau such that successive iterations no longer produce any improvement.

A single-objective optimization problem for a typical calibration is given as:

$$\underset{\text{such that } \mathbf{x}\in\mathbf{S}}{\arg\min}\ O(\mathbf{x}), \tag{1}$$

where: the design variable vector $\mathbf{x} = (x_1, x_2, \ldots x_n)^\intercal$ is a set of input parameters varying within a search space $\mathbf{V} \subset \mathbb{R}^n$, and intervals $S_i = [l_i, u_i]$ (for $i = 1, 2, .., n$) are the given lower and upper bounds of parameter x_i. O is the objective function to be minimized, which is the fitness function of GAs and defined as the discrepancies between DEM model results and experimental measurement data.

To represent input parameters by means of a binary alphabet, a function $cod_i : S_i \rightarrow \{0 \vee 1\}$,has to be specified, which codes each parameter x_i in interval S_i using binary strings of length L_i characters. An input parameter set, a so called individual in GAs, is represented by linking together the coded binary strings of all parameters as $cod(\mathbf{x}) = cod_1(x_1)cod_2(x_2)\ldots cod_n(x_n)$.

For example, a calibration problem involves $n = 2$ input parameters: rolling and sliding friction coefficients, $\mathbf{x} = (x_1, x_2) = (\mu_r, \mu_s)$, in ranges [0.1,0.5] and [0.2,0.7], respectively. If, for instance, 4 and 5 binary bits defining ranges of 0000–1111 and 00000–11111 are used to encode μ_r and μ_s, respectively, the encoding functions cod_1 and cod_2 map the continuous solution space into the numerical discrete solution space. The discrete solution space consists of $2^4 \times 2^5 = 512$ possible values and each value is presented by a string of 9 binary bits. This encoding provides a numerical precision of $\frac{0.5-0.1}{2^4-1} \times \frac{0.7-0.2}{2^5-1} = 0.02666 \times 0.01613$. A parameter set (μ_r, μ_s) with real values, *e.g.*, (0.2862, 0.5387), can be encoded by an individual as (01110101), in which the first four digits are used for μ_r and the last five digits for μ_s. At the beginning of the GA process, an initial population $\mathbf{P}$ with $|\mathbf{P}|$ individuals, is randomly generated.

In the next phase of GAs, selection techniques are used to choose the new population of individuals for the next generation. Selection is an important part of GAs since it affects significantly their convergence. The basic strategy follows the rule: the better fit an individual, the larger the probability of its survival and mating. In this study, the most widely used selection type, the so-called binary tournament selection [31], is applied. It assumes that the probability of selection for crossover is proportional to the fitness of an individual.

The crossover operator mimics the natural process in biology whereby genes from two parents meet to produce new offsprings that are a mix of the parent's genes. Crossover is vitally important in introducing new possible solutions by exploring new domains of the search space. Individuals, as a result of the selection process, are randomly selected and mated in pairs using a single-point crossover [31].

The mutation operator is used to promote genetic diversity from the current population to the next, aiming at introducing new genetic information in the search. Analogous to biological mutation, this operator alters one or more gene values in randomly

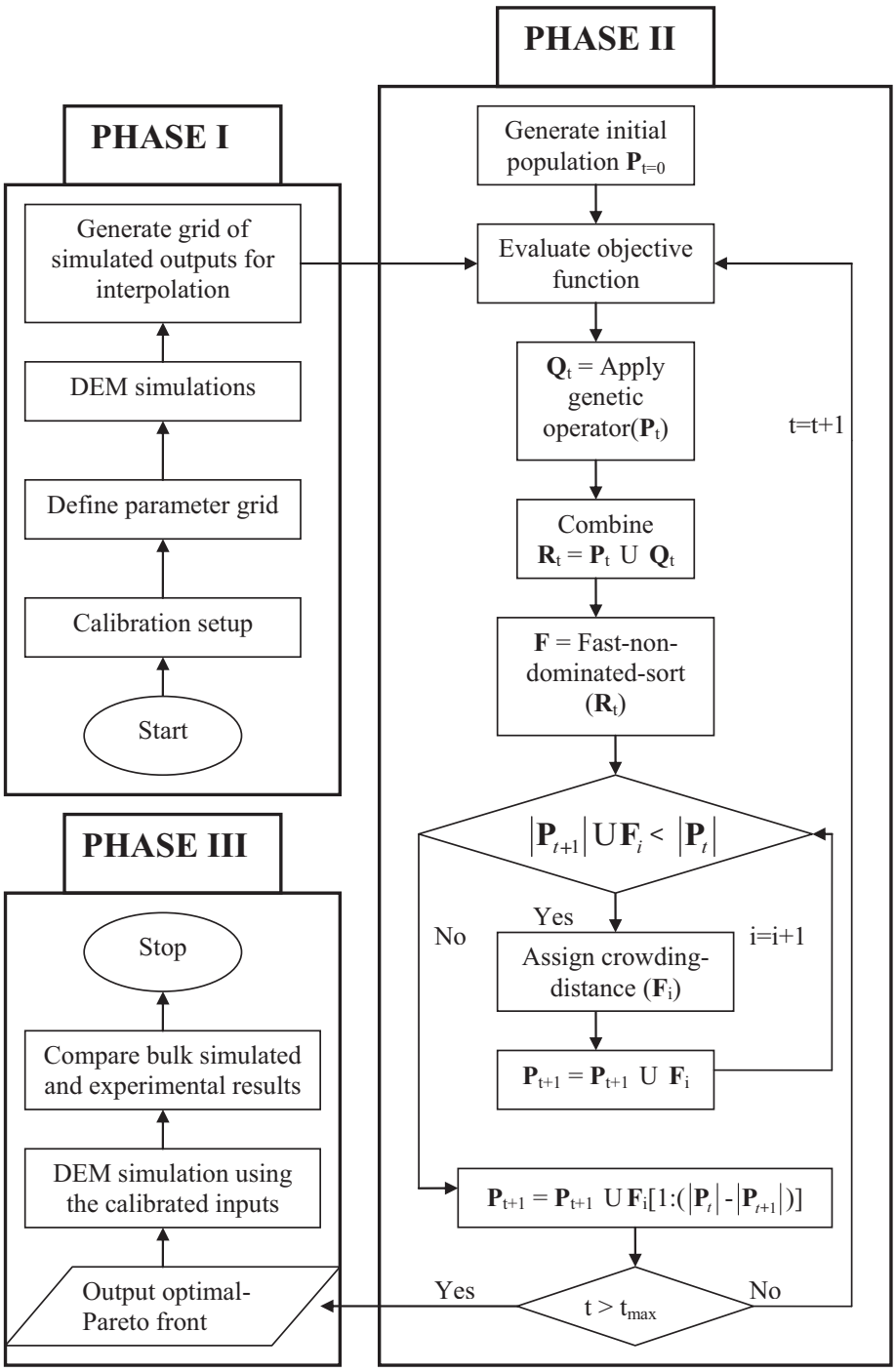

Fig. 1. Flowchart of the calibration approach.

selected individual strings with a certain probability, for instance, 0.01 as used in this study.

2.2. Multi-objective genetic algorithm

Multi-objective evolutionary algorithms (MOEAs) [32] are found to have more advantages than the simple GA in solving multi-objective optimization problems (MOOPs). With MOEAs, tradeoffs among conflicting objectives can be visualized using a Pareto-optimization process, in which individuals are sorted by all objectives to select the best candidates for the next generation. The concept of Pareto dominance is defined here to understand how the MOEA operates. An individual dominates another if it has a better fitness in at least one objective function and is no worse in the rest of the objectives [33]. With this concept, the key difference from a single-objective optimization is that the selection process of MOEA remains non-dominated solutions. An evolutionary process of two objective-optimization is illustrated in Fig. 2, where a population $\mathbf{P}$ with a fixed number $|\mathbf{P}|$ of individuals is randomly created and then optimized. Through the evolutionary process from generations $t = 0$ to $t = t_{max}$, individuals relocate to the so-called Pareto-optimal front. All individuals lying on this front are non-dominated and optimal solutions to the MOOP.

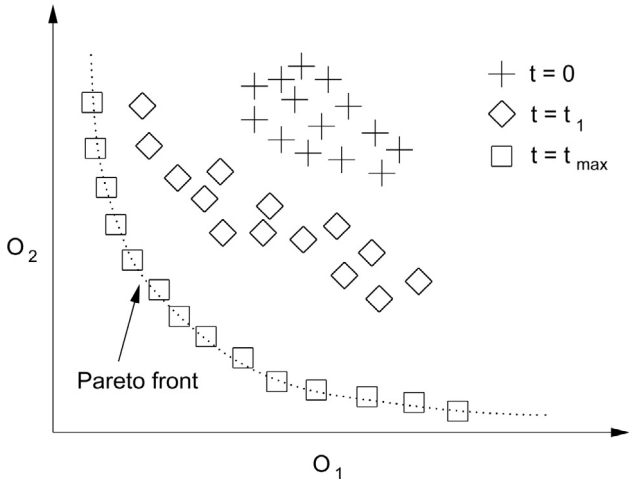

Fig. 2. Schematic diagram of two-objective optimization, showing a randomly initiated population of individuals at $t = 0$ evolving towards a Pareto-optimal front at $t = t_{max}$.

In this study, the Non-dominated Sorting Genetic Algorithm II (NSGA-II) proposed by Deb et al. [30] is used. The flowchart of NSGA-II is illustrated as phase II in Fig. 1. At a certain generation t, an offspring population $\mathbf{Q}_t$ is generated by applying the genetic operators, selection, crossover, and mutation, on the current population $\mathbf{P}_t$, forming a combined parent-offspring population $\mathbf{R}_t$ of size of 2 $|\mathbf{P}|$. New parents are sorted from the combined population $\mathbf{R}_t$ by using binary tournament selection based on their non-domination rank into fronts $\mathbf{F}_i$ $(i = 1, 2, \ldots)$: the first front $\mathbf{F}_1$ being completely non-dominated in the current combined parent-offspring population $\mathbf{Q}_t$; the second front $\mathbf{F}_2$ being dominated only by individuals in the first front; and so forth. To select exactly $|\mathbf{P}|$ individuals that will constitute the parent population in the next generation $t + 1$, the algorithm first selects all fronts (with the priority for lower/better ranks) that can be completely packed within the population size of $|\mathbf{P}|$. The remaining individuals (if any) are selected by considering a crowding distance parameter, used to distribute individuals more evenly along the non-dominated front.

2.3. Discrete element modeling

DEM is a numerical technique for computing the motion and collisions of a large number of small particles. In this study, the open-source software LIGGGHTS [34] is chosen to conduct DEM simulations. The classical Hertz-Mindlin contact theory with elastic-plastic spring-dashpots rolling friction model is employed in describing non-cohesive interaction between individual particles. In a simulation, the motions of particles are integrated forward in time using discrete time steps. The sand grains are modeled as soft-spherical particles whose realistic deformation under collisions are simplified by their overlaps. At a certain time step, the total force from friction and gravity acting on each particle is calculated. The change in the position and the velocity of each particle during this time step is computed using an integration scheme such as the Verlet algorithm [35]. The new positions are used to compute the forces during the next time step, and this loop is repeated until the simulation ends.

3. Case study

To demonstrate the performance of the calibration framework, a case study is investigated. In this case study, hourglass, ledge, and conical pile formation tests are simulated for the calibration, verification, and validation process. The experimental results of these three tests were obtained and reported by Derakhshani et al. [36]. The hourglass test is used to calibrate and verify an input parameter set of micro properties by comparing observable macro properties on the bulk density, angle of repose (*AoR*), and discharging time obtained from DEM simulation and experiment. Then, the ledge and conical pile tests are used to validate these calibrated input parameters by comparing the *AoR* obtained from DEM simulation and experiment.

3.1. Sandglass simulation

The simulation setup of the hourglass test on quartz sand material is sketched in Fig. 3. The properties of the quartz sand as the DEM input parameters with their possible values are listed in Table 1. Although, for the use of the algorithm, any interval can be chosen, some of these parameter intervals are defined according to the data from the study by Derakhshani et al. [36], while the other, *e.g.*, the lower bound of Young's modulus, are chosen based on feasible and acceptable bounds for LIGGGHTS's input. An amount of 25 g of mono-sized spherical particles is randomly poured under gravity into an empty cylindrical hopper of 32 mm with its initially closed neck of 5 mm in diameter. After the particles are settled, the filled bulk density is calculated using the total particle mass over the filled volume. The plug of the hopper neck is then quickly removed and particles start to discharge under gravity. The mean kinetic energy of the particles remaining in the hopper is used as a criterion in the DEM simulations to assess the stability of the slope. In this study, a value of mean kinetic energy of 10^{-15} J of particles remaining in the hourglass is defined as the reference for a stable state of the system. Then, the discharging time is calculated as the time between removal of the plug until the stable state has been reached and the *AoR* of the slope can then be computed.

3.2. Ledge simulation

The simulation setup of the ledge test is depicted in Fig. 4. To reduce the simulation time, periodic boundary conditions were applied on the front and back sides of the ledge test. The periodic boundary condition means that when an object passes through one side of the rectangular box, it re-appears on the opposite side with the same velocity. Mono-sized spherical particles of sand are randomly filled into a rectangular box of $16 \times 16 \times 3.2$ mm^3 in length, height and width, respectively. These dimensions of the rectangular box are large enough to eliminate the wall effect on the simulated *AoR*. After the particles are settled, the flap opens to release them and to form the slope of the material. When the mean kinetic energy of particles remaining in the ledge has been reached to the reference value of 10^{-15} J, the *AoR* of the slope can be computed.

3.3. Conical pile simulation

The simulation setup of the conical pile formation test using the cylinder method is sketched in Fig. 5. Mono-sized spherical particles are randomly poured into an empty cylinder of 32 mm in diameter. The particles are given some time to settle to a desired height of 20 mm. The cylindrical wall is lifted upward with a constant speed of 10 mm s^{-1} and then it is kept in the fixed height of 10 mm from the base. The particles are discharged and form a conical pile. The *AoR* of the pile can then be computed when the mean kinetic energy of the particles of the pile has been reached to the reference value of 10^{-15} J.

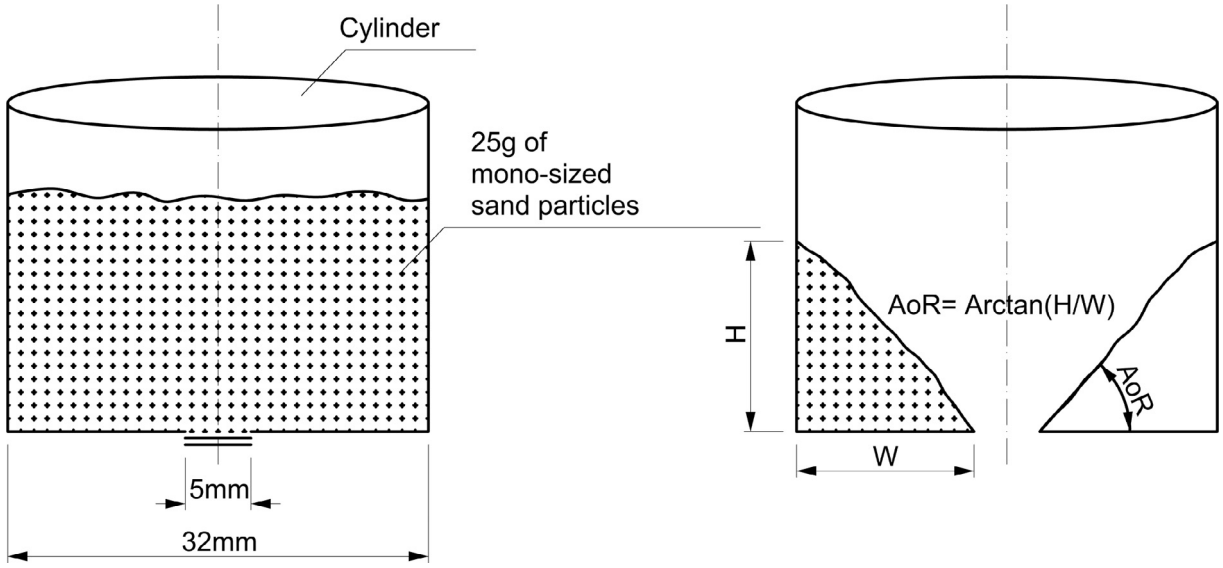

Fig. 3. Simulation setup of the hourglass test.

Table 1
Input parameters that varies with their values used in DEM simulations to generate the interpolation grid. The lower and upper bounds of each parameter interval are specified by their minimum and maximum sampling values. The cross product of all parameter intervals defines the search space of the optimization problem.

Input parameter	Symbol	Unit	Interval	Number of sampling
Rolling friction coefficient	μ_r	–	[0.1, 0.5]	5
Sliding friction coefficient	μ_s	–	[0.2, 0.7]	6
Particle radius	r	mm	[0.3, 0.4]	3
Particle density	ρ_p	kg m^{-3}	[2600, 2800]	3
Particle Young's modulus	E	MPa	[5, 500]	3

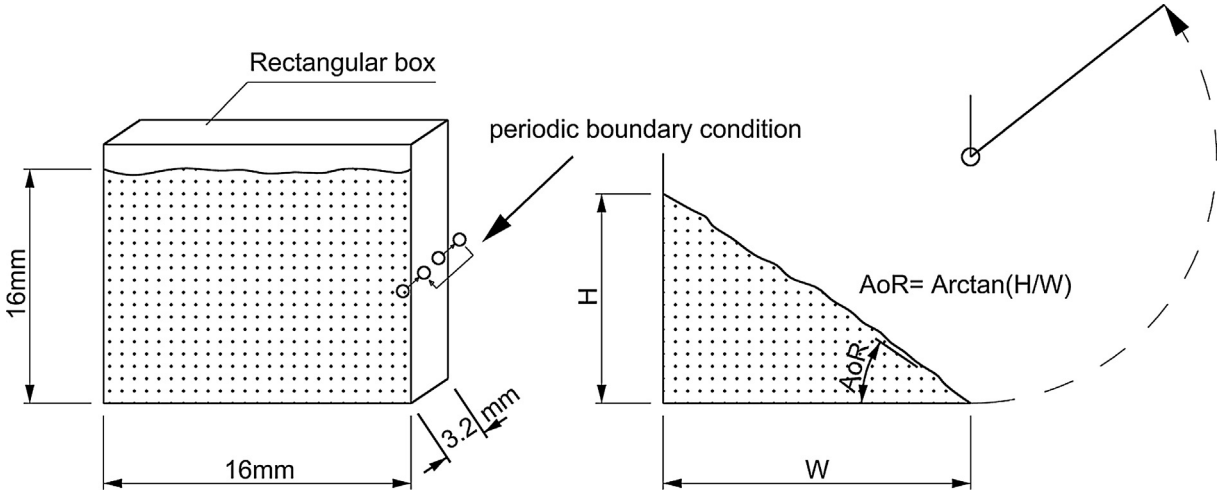

Fig. 4. Simulation setup of the ledge test.

3.4. Two-objective optimization problem setup

Five independent variables, listed in Table 1, are being calibrated using the sand glass simulations: static friction and rolling friction coefficients between particles, radius, density and Young's modulus of particles. The other input parameters, which are fixed for all DEM simulations, are listed in Table 2. The actual input time step used for the calibration process is 2×10^{-6} s to guarantee the stability and accuracy of all DEM runs. Three macroscopic properties from the experimental observations, listed in Table 3, are considered as the target references for the simulated outputs: bulk density, *AoR*, and discharging time of the sand material. The calibration process searches for the optimal input parameter set whose simulation produces the best match of simulated outputs with the experimental measurements and/or consumes the least simulation time.

Even though a reduction in simulation time can be obtained by increasing the simulation time step, a sufficiently short time step is required to ensure the numerical stability and accuracy of DEM computations. The input simulation time steps are suggested to

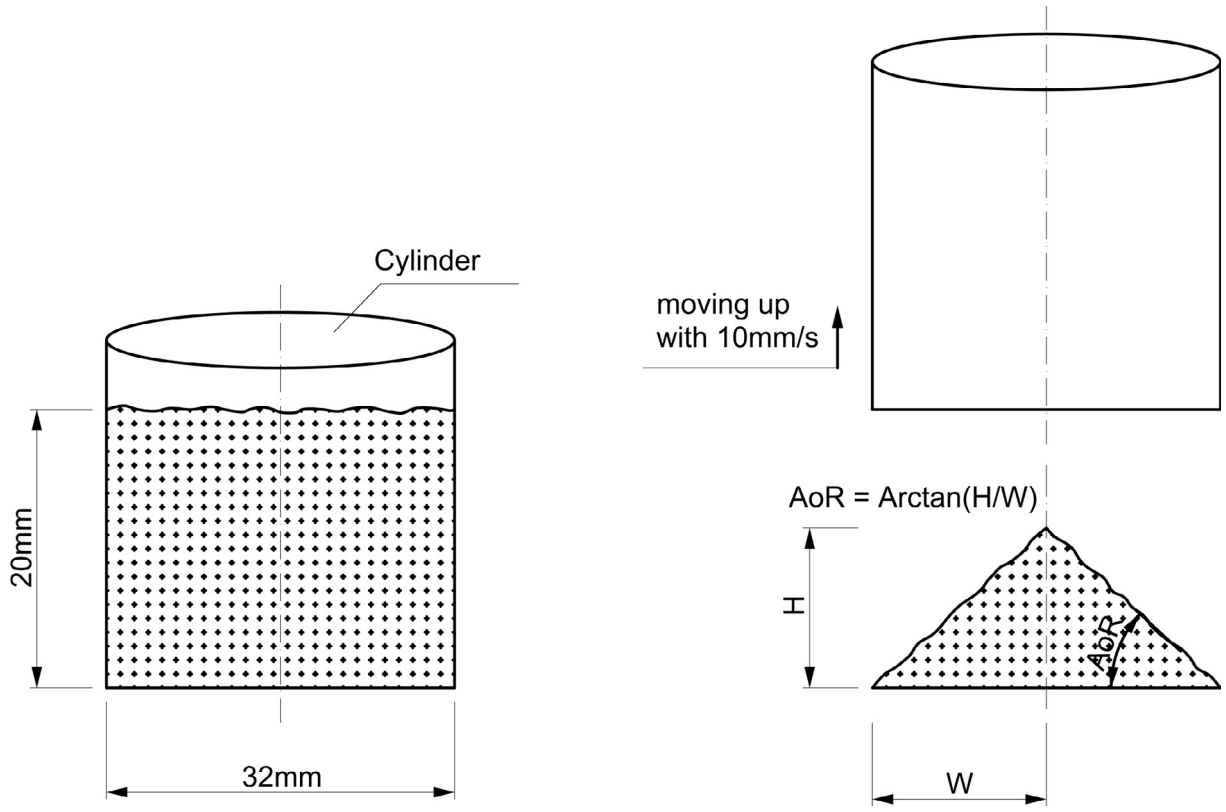

Fig. 5. Simulation setup of the conical pile test.

Table 2
Input parameters that are fixed for all DEM simulations to generate the interpolation grid.

Input parameter	Symbol	Unit	Value
Poisson's ratio	v	–	0.3
Coefficient of restitution	e	–	0.9
Simulation time step	t	sec	2×10^{-6}

Table 3
Experimental measurements are considered as the targeted references for simulated outputs [36].

Experimental measurement	Symbol	Unit	Value
Bulk density	ρ_{ex}	kg m^{-3}	1530
Angle of repose	AoR_{ex}	deg	41.57
Discharging time	Dis_{ex}	sec	6.56

be smaller than 20% of the critical time step that is determined by Rayleigh's time as:

$$T_R = \frac{\pi r \sqrt{\frac{\rho_p}{G}}}{0.163v + 0.8766}. \tag{2}$$

Therefore (2) indicates that DEM simulations with a bigger radius, a higher density, and a smaller Young's modulus of particles, will reduce the simulation time. Since the stable Rayleigh's time step is dependent on the parameter values, the time step is included in the optimization process as an additional cost criterion. Hence, the result from the optimization process will not only be an accurate set of parameter values but will also result in the largest minimum time step. This is especially important for computationally intensive large-scale industrial applications.

In this optimization problem, two objective functions are defined for minimization of the simulation error and the simulation time. The first objective function O_1 is defined as the total relative discrepancies between the numerical outputs and the experimental measurements on AoR, discharging time, and bulk density:

$$O_1 = \frac{|AoR_{si} - AoR_{ex}|}{AoR_{ex}} + \frac{|Dis_{si} - Dis_{ex}|}{Dis_{ex}} + \frac{|\rho_{si} - \rho_{ex}|}{\rho_{ex}}, \tag{3}$$

and in short:

$$O_1 = Er_{AoR} + Er_{Dis} + Er_{\rho} \tag{4}$$

In this particular optimization example, the weights of three error components are equally contributing to the total simulation error. Thus, the weights of each term in the right hand side of Eq. (3) are set to be the same. Generally, the weights can be set to any value.

The second objective function O_2 aims to reduce the simulation time:

$$O_2 = -\left(\frac{r - r^{\min}}{r^{\max} - r^{\min}}\right) - \left(\frac{\rho_p - \rho_p^{\min}}{\rho_p^{\max} - \rho_p^{\min}}\right) + \left(\frac{E - E^{\min}}{E^{\max} - E^{\min}}\right) + 2, \tag{5}$$

where $[r^{\max}, r^{\min}], [\rho_p^{\max}, \rho_p^{\min}], [E^{\max}, E^{\min}]$ are the intervals for the radius, density, and Young's modulus of particles, respectively, which are listed in the column "Interval" in Table 1.

In (5), the second objective function O_2 is defined as the summation of three linear functions of three independent variables that are the parameters being calibrated: radius, density, and Young's modulus of particles. Minimizing O_2 leads to maximizing the Rayleigh's time step T_R, and thus reducing the simulation time. In the objective function O_2, the negative sign for the radius and density functions and the positive sign for the Young's modulus

imply that higher radius, higher density, and lower Young's modulus values are preferred. The addition of 2 guarantees a positive objective function, since:

$$\min(O_2) = O_2(r = r^{\max}, \rho_p = \rho_p^{\max}, E = E^{\min}) = 0 \quad (6)$$

$$\max(O_2) = O_2(r = r^{\min}, \rho_p = \rho_p^{\min}, E = E^{\max}) = 3 \quad (7)$$

Input parameters for the GA are listed in Table 4. An initial population of 5000 individuals is randomly created. This population size is found to be good enough to demonstrate the performance of the NSGA-II algorithm for the optimization problem proposed in Section 3.4. By using 8 bits for encoding each calibrated input parameter, as shown in Table 1, the number of possible values for each parameter is $2^8 = 256$. Each individual in the population is encoded by a binary string of length 40 bits. The properties of this encoding are listed in Table 5. As it is usually done in GAs [31], a high probability of crossover and a low probability of mutation are chosen. The population evolves through 100 generations by which the Pareto front has stabilized and successive generations no longer produce any further improvement.

4. Results and discussion

The microscopic input properties are calibrated, and then verified using simulations of the hourglass test. Afterwards, these calibrated properties are validated using simulations of the ledge test and the conical pile test.

4.1. Calibration using sandglass simulation

A typical running process using NSGA-II for the optimization problem proposed in Section 3.4 is shown in Fig. 6. This figure presents snapshots of the two objective functions at generations $t = 0, t = 1, t = 2, t = 3, t = 20$, and $t = 100$. The initial population of size 5000 is scattered on the search planes, shown in Fig. 6a. The total relative error O_1 varies from 0 to around 0.5 and the simulation time characteristic of O_2 varies from 0 to 3. As expected, the variation of O_2 is in agreement with Eqs. (6) and (7). A comparison between Fig. 6a and b indicates that a large amount of dominated individuals surrounding the upper-right corner is removed from the population during the first generation. This is because after obtaining the combined population of size $2 \times |\mathbf{P}|$, those individuals are badly ranked by the non-dominated sorting algorithm. In the next generations, as shown in Fig. 6c, d, e, and f), the individuals densely populate and gradually relocate towards the origin at the lower-left corner. In this case study, the number of individuals remains the same, though the populated area tends to shrink at each generation. Through the evolution process, the population gradually stabilizes to its final best front that does not show any further improvement in the next generations. This means that the algorithm has converged. The final front, at $t = 100$, shown in Fig. 6f, clearly demonstrates the trade-off between the two objectives functions.

Table 4
Input parameters of GA.

GA's parameter	Value
Population size	5000
Maximum number of generation	100
Chromosome length (bit)	40
Crossover probability	0.9
Mutation probability	0.01

Table 5
Input parameter intervals are encoded using binary bits and their corresponding numerical precision.

Parameter	Interval	Numerical precision
μ_r	[0.1, 0.5]	1.57×10^{-3}
μ_s	[0.2, 0.7]	1.96×10^{-3}
r	[0.3, 0.4]	3.92×10^{-4}
ρ_p	[2600, 2800]	0.7843
E	[5, 500]	1.9412

To measure the performance of NSGA-II on this optimization problem, the mean generational distance G_i of front Pf_i at generation i is computed as:

$$G_i = \frac{\sum_{j=1}^{n} d_j}{n}, \quad (8)$$

where n is the number of points on front Pf_i, d_j is the distance from point j on front Pf_i to its nearest point on the final front at generation $t_{\max} = 100$.

The two-objective optimization problem is carried out three times to compute the mean generational distances, as plotted in Fig. 7a. The overlapping of the three mean generational distance curves clearly indicates that the algorithm provides consistent results of Pareto fronts. All mean generational distances quickly drop down after a few generations, then approach to zero after about 30 generations. This means the fronts, which are from 30th generation on, already converge onto the best front shown in Fig. 6f.

To take a closer look at the convergence rate of the optimization problem, the mean generational distance curves are plotted again in the logarithmic scale. Fig. 7b indicates that the convergence rate of the algorithm is in a logarithmic law with respect to generation.

In Fig. 6f, four individuals **A**, **B**, **C**, and **D** are extracted from the final front at generation $t = 100$. Individual **A** represents a solution with the parameter set that consumes the longest simulation time, but gives the smallest total simulation error of zero. This means that there is a perfect match between the simulated outputs and the referred experimental measurements, simultaneously on bulk density, *AoR*, and discharging time. On the contrary, individual **D** represents a solution with the parameter set that gives the largest total simulation error of 0.0428, but consumes the least simulation time corresponding to the optimal value of O_2 of zero. This is achieved when the variables are assigned to their bound values, *i.e.*, $r = r^{\max}, \rho_p = \rho_p^{\max}$, and $E = E^{\min}$. The results of the calibrated input parameters and the objective function values of individuals **A**, **B**, **C**, and **D** are listed in Table 6.

4.2. Verification using sandglass simulation

For verification of the calibration results, four DEM simulations using the optimized input parameter sets of individuals **A**, **B**, **C**, and **D** are carried out to calculate bulk properties. Subsequently, the values of objective functions O_1 and O_2 at the optimized input parameter sets are calculated using Eqs. (3) and (5). The values of O_1 and O_2 listed in Table 7 are in good agreement with those listed in Table 6. The simulation errors of individual **D** mainly stem from the bulk density error that is strongly controlled by the input particle density value. It is found that the values of rolling and sliding friction coefficients of individuals **A** and **D** can be swapped to produce similar *AoR* outputs.

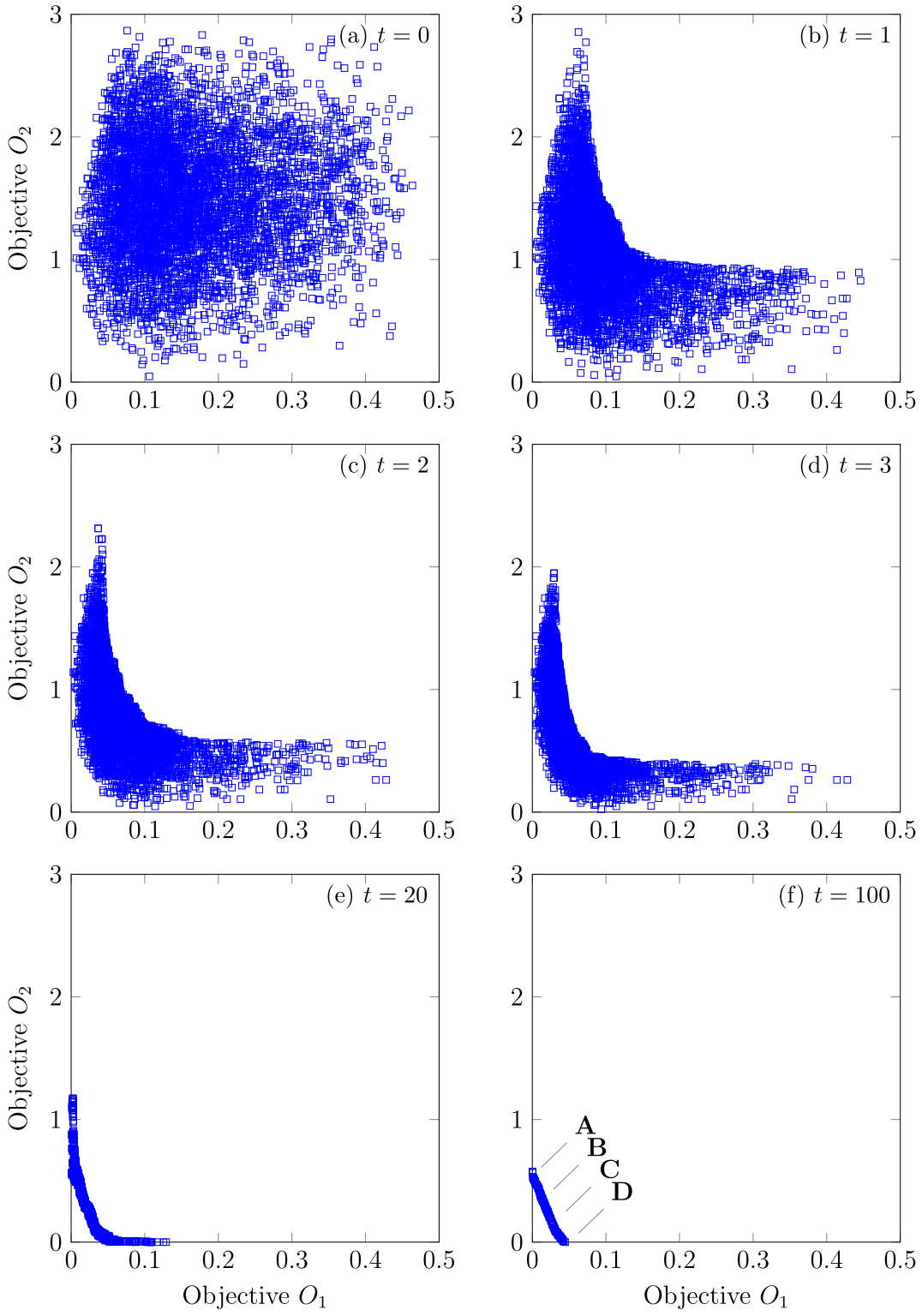

Fig. 6. Pareto front evolution of the two objective functions in NSGA-II.

4.3. Validations using ledge and conical pile simulations

The validity of the verified DEM calibration is assessed by simulating ledge and conical pile formation tests as presented in Section 2.3. For each numerical test, four DEM simulations using the optimized input parameters sets of individuals **A**, **B**, **C**, and **D** are run to calculate *AoR*. The numerical *AoR* results from ledge and conical pile simulations are listed in Tables 8 and 9, respectively. A good agreement between the numerical results and the experimental measurements for both these test setups confirms that the input parameter sets are validated with high accuracy.

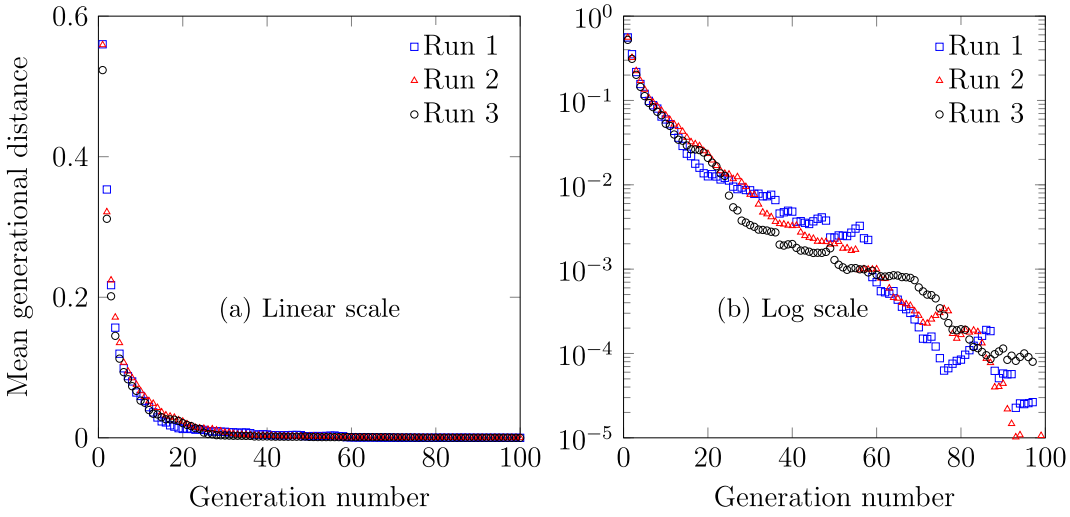

Fig. 7. Mean generational distance to the final front at generation $t_{max} = 100$.

Table 6
Calibration results.

Individual	A	B	C	D
μ_r	0.429	0.395	0.361	0.327
μ_s	0.351	0.377	0.403	0.429
r	0.4	0.4	0.4	0.4
ρ_p	2712.16	2741	2770	2800
E	4.96×10^7	3.47×10^7	1.98×10^7	5×10^6
Er_{AoR}	0.0001	0.0004	0.0008	0.0003
Er_{Dis}	0.0005	0.0009	0.0017	0.0016
Er_ρ	0.0005	0.0135	0.0272	0.0427
O_1	0.0011	0.0148	0.0297	0.0445
O_2	0.5294	0.3529	0.1765	0.0000

Table 7
Verification results.

	A	B	C	D	Experiment
AoR	41.60	41.66	41.66	41.71	41.57
Dis	6.54	6.62	6.64	6.66	6.56
ρ	1531	1554	1586	1594	1530
Er_{AoR}	0.0007	0.0022	0.0022	0.0034	
Er_{Dis}	0.0030	0.0091	0.0122	0.0152	
Er_ρ	0.0008	0.0157	0.0366	0.0419	
O_1	0.0045	0.0270	0.0510	0.0605	
O_2	0.5294	0.3529	0.1765	0.0000	

Table 8
Validation results using ledge test.

	A	B	C	D	Experiment
AoR	35.65	35.59	34.96	35.79	35.1
Er_{AoR}	0.0157	0.0140	0.0040	0.0197	

Table 9
Validation results using conical pile test.

	A	B	C	D	Experiment
AoR	33.70	33.89	32.55	33.17	33.00
Er_{AoR}	0.0212	0.0270	0.0136	0.0052	

4.4. Conclusion on the case study

In this case study, rolling and sliding friction coefficients, size, density, and Young's modulus of particles of a DEM model are determined based on the known bulk properties of density, *AoR*, and discharging time. The verification and validation on the calibrated parameter sets have confirmed that the calibration approach can be utilized to predict accurately micro properties of granular materials with high reliability.

When the Rayleigh's time step is actively considered as a calibration target, the actual time step size assigned in simulations can be increased significantly, while the overall numerical calibration error is still negligible. The algorithm is able to capture the tradeoff between the model accuracy and the simulation time.

The generational distance analysis indicates that NSGA-II has a logarithmic convergence rate and the convergence has been obtained after 30 generations.

5. Conclusion

In this paper, an automated calibration framework based on NSGA-II, a multi-objective optimization evolutionary algorithm, has been presented. Through a case study with two-objective optimization, the model accuracy and the computational time, the calibration framework has been successfully demonstrated. The Pareto-optimal front visualizes the best compromise between the two conflicting objectives. This feature of the calibration framework is important for large-scale applications. Pareto-fronts can be used to reduce optimization run time by indicating candidate solutions of parameter set that will be used for large-scale problems with acceptable user-defined accuracy and simulation time.

The case study illustrates that the population has been stabilized onto the Pareto-optimal front by which subsequent generations create individuals that spread uniformly along it. However, the algorithm has difficulties when trying to find the individuals with extreme values on the Pareto-optimal front.

The Pareto-optimal front indicates that there is a solution domain of several feasible input parameter sets that result in similar simulated bulk responses as the desired calibration targets. To narrow the solution domain down, more targets should be included in the objective function for matching bulk responses with experiments. This would require further appropriate experimental and numerical tests. In such cases, it emphasizes the need for a systematic and automated calibration approach such as the universal framework as presented in this study.

The genetic algorithm is robust and stable in a variety of calibration applications. When laboratory calibration setups and contact force models are appropriately chosen, *i.e.*, reflect the dominant material behavior in the application, the proposed optimization procedure can be generally applied. The procedure can be used with any contact force model. It is expected that laboratory model setups with a sufficient dimension relative to the particle size can be used to obtain parameter sets for large-scale industrial applications. In that case similarity of material behavior in laboratory and application can be expected.

In principle, any contact force model can be used. However, when a high number of parameters of contact force models needs to be calibrated, the procedure will take longer. To reduce computational cost and then improve the calibration framework, it could be combined with other methods. The genetic algorithm with a coarse gird of the design space is first used to identify the areas with desirable values of the objective functions. Then, gradient-based optimizations on these areas can be used to identify the solutions with higher accuracy.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.apt.2018.03.001.

References

[1] P. Cundall, O. Strack, A discrete numerical model for granular assemblies, Geotechnique 29 (1979) 47–65.
[2] F. Alonso-Marroquín, Á. Ramírez-Gómez, C. González-Montellano, N. Balaam, D. Hanaor, E. Flores-Johnson, Y. Gan, S. Chen, L. Shen, Experimental and numerical determination of mechanical properties of polygonal wood particles and their flow analysis in silos, Granul. Matter 15 (6) (2013) 811–826.
[3] G. Barrios, R. de Carvalho, A. Kwade, L. Taveres, Contact parameter estimation for DEM simulation of iron ore pellet handling, Powder Technol. 248 (2013) 84–93.
[4] M. Marigo, E. Stitt, Discrete element method (DEM) for industrial applications comments on calibration and validation for the modelling of cylindrical pellets, Kona Powder Part. J. 32 (2015) 236–252.
[5] C. Coetzee, D. Els, Calibration of discrete element parameters and the modelling of silo discharge and bucket filling, Comput. Electron. Aric. 65 (2) (2009) 198–212.
[6] A. Grima, P. Wypych, Investigation into calibration of discrete element model parameters for scale-up and validation of particle-structure interactions under impact conditions, Powder Technol. 212 (2011) 198–209.
[7] M. Combarros, H. Feisei, H. Zetzener, A. Kwade, Segregation of particulate solids: experiments and DEM simulations, Paticuology 12 (1) (2014) 25–32.
[8] Q. Li, M. Feng, Z. Zou, Validations and calibration approach for discrete element simulation of burden charging in pre-reduction shaft furnace of COREX process, ISJI Int. 58 (8) (2013) 1365–1371.
[9] M. Ucgul, J. Fielke, C. Saunders, Thee-dimensional discrete element element modelling of tillage: determination of suitable contact model and parameters for a cohesionless soil, Biosyst. Eng. 121 (2014) 105–117.
[10] P. Frankowski, M. Morgeneyer, Calibration and validation of DEM rolling and sliding friction coefficents in angle of repose and shear measurements, in: AIP Conf. Proc., vol. 1542, 2013, pp. 851–854.
[11] I. Marczewska, J. Rojek, R. Ka, R. Kaianauska, Investigation of the effective elastic parameters in the discrete element model of granular material by triaxial compression test, Arch. Civ. Mech. Eng. 16 (1) (2016) 64–75.
[12] N. Belheine, J. Plassiard, F. Donz, F. Darve, Numerical simulation of drained triaxial test using 3d discrete element modeling, Comput. Geotech. 36 (1–2) (2009) 320–331.
[13] C. Gonzlez-Montellano, Ramrez, E. Gallego, F. Ayuga, Validation and experimental calibration of 3d discrete element models for the simulation of the discharge flow in silos, Chem. Eng. Sci. 66 (21) (2011) 5116–5126.
[14] H. Do, A. Aragón, D. Schott, Automated discrete element method calibration using genetic and optimization algorithms, in: Powders and Grains 2017 8th International Conference on Micromechanics on Granular Media, vol. 140, 2017.
[15] A. Tarantola, Inverse Problem Theory and Methods for Model Parameter Estimation, Society for Industrial and Applied Mathematics, Philadelphia, 2005.
[16] R. Aster, B. Borchers, C. Thurber, Parameter Estimation and Inverse Problem, Elsevier, 2013.
[17] J. Yoon, Application of experimental design and optimization to PFC model calibration in uniaxial compression simulation, Int. J. Rock. Mech. Min. Sci. 44 (6) (2007) 871–889.
[18] K. Hanley, C. O'Sullivan, J. Oliveira, K. Cronin, E. Byrne, Application of Taguchi methods to DEM calibration of bonded agglomerates, Powder Technol. 210 (2011) 230–240.
[19] S. Wilkinson, S. Turnbull, Z. Yan, E. Stitt, M. Marigo, A parametric evaluation powder flowability using freeman rheometer through statistical and sensitivity analysis: a discrete element method (DEM) study, Comput. Chem. Eng 97 (2017) 161–174.
[20] L. Benvenuti, C. Kloss, S. Pirker, Identification of DEM simulation parameters by Artificial Neural Networks and bulk experiments, Powder Technol. 291 (2016) 456–465.
[21] M. Rackl, K. Hanley, A methodical calibration procedure for discrete element models, Powder Technol. 307 (2017) 73–83.
[22] Z. Asaf, D. Rubinstein, I. Shmulevich, Determination of discrete element model parameters required for soil tillage, Soil Tillage Res. 92 (1-2) (2007) 227–242.
[23] J. Quist, M. Evertsson, Framework for DEM model calibration and validation, in: Proceedings of the 14th European Symposium on Comminution and Classification, Gothenburg, Sweden, 2015, pp. 103–108.
[24] R. Briend, P. Radziszewski, D. Pasini, Virtual soil calibration for wheel soil interaction simulations using the discrete-element method, Can. Aeronaut. Space J. 57 (1) (2011) 59–64.
[25] F. Elskamp, H. Kruggel-Emden, M. Hennig, U. Teipel, A strategy to determine dem parameters for spherical and non-spherical particles, Granul. Matter 19 (3) (2017) 1–13.
[26] J. Horn, N. Nafpliotis, D. Goldberg, A niched pareto genetic algorithm for multiobjective optimization, in: IEEE World Congress on Computational Intelligence, Proceedings of the First IEEE Conference on Evolutionary Computation, vol. 1, Orlando, FL, USA, 1994.

[27] N. Srinivas, K. Deb, Muiltiobjective optimization using nondominated sorting in genetic algorithms, Evol. Comput. 2 (3) (1995) 221–248.
[28] C. Fonseca, P. Fleming, Genetic algorithms for multiobjective optimization: Formulation discussion and generalization, in: S. Forrest (Ed.), Genetic Algorithms: Proceedings of the Fifth International Conference, Morgan Kaufmann, San Mateo, CA, 1993.
[29] S. Kukkonen, J. Lampinen, GDE3: the third evolution step of generalized differential evolution, in: The 2005 IEEE Congress on Evolutionary Computation, vol. 1, 2005.
[30] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, A fast and elitist multi-objective genetic algorithm: NSGA-II, IEEE Transact. Evol. Comput. 6 (2) (2002) 181–197.
[31] D. Goldberg, Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company, Masachusetts, 1989.
[32] K. Deb, Multi-objective Optimization using Evolutionary Algorithms, Wiley, Chichester, UK, 2001.
[33] V. Pareto, Manuale di Economia Politica, Piccola Biblioteca Scientifica, Milan, 1906, translated into English by Ann S. Schwier (1971), Manual of Political Economy, MacMillan, London.
[34] C. Kloss, A. Goniva, S. Hager, S. Pirker, Models, algorithms and validation for opensource DEM and CFD-DEM, Prog. Comput. Fluid. Dy. 12 (2/3) (2012) 140–152.
[35] L. Verlet, Computer experiments on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules, Phys.Rev. 159 (98) (1967) 98–103.
[36] S. Derakhshani, D. Schott, G. Lodewijks, Micro-macro properties of quartz sand: Experimental investigation and DEM simulation, Powder Technol. 269 (2015) 127–138.