Application of DEM-based metamodels in bulk handling equipment design
Methodology and DEM case study

Fransen, Marc P.; Langelaar, Matthijs; Schott, Dingena L.

DOI
10.1016/j.powtec.2021.07.048
Publication date
2021
Document Version
Final published version
Published in
Powder Technology

Powder Technology 393 (2021) 205–218

Contents lists available at ScienceDirect

Powder Technology

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

Application of DEM-based metamodels in bulk handling equipment design: Methodology and DEM case study

Marc P. Fransen [a,*], Matthijs Langelaar [b], Dingena L. Schott [a]

[a] Delft University of Technology, Department of Maritime and Transport Technology, Delft, the Netherlands
[b] Delft University of Technology, Department of Precision and Microsystems Engineering, Delft, the Netherlands

ARTICLE INFO

Article history:
Received 13 May 2021
Received in revised form 13 July 2021
Accepted 18 July 2021
Available online 24 July 2021

Keywords:
Metamodeling
DEM
Bulk handling equipment
Metamodel construction procedure
Hopper discharge

ABSTRACT

Developments in discrete element modelling (DEM) enable detailed modelling of granular flows in bulk handling equipment (BHE) but due to the computational expense of DEM, wide use in analysing equipment performance is not yet feasible. Metamodels are a viable option to effectively use DEM in analysing BHE performance. Metamodels are able to approximate the behaviour of BHE efficiently for a wide range of design parameter values. We present a methodology to construct and validate DEM-based metamodels as well as a discharging hopper case study illustrating the use and benefits of metamodels in combination with DEM. For three different metamodels trained on a DEM data set, the results show that the metamodel quality highly depends on the number of samples and finding proper hyper-parameter values. The constructed metamodels are found capable of adequately representing the relation between performance and design parameters. It is concluded that methodically constructed metamodels are a valuable addition in describing BHE behaviour.

1. Introduction

Recent developments in modelling of large scale particle systems and an increase in computing power enable researchers and engineers to model behaviour of bulk and powder handling equipment in increasing detail. A powerful modelling technique for particulate systems is the discrete element method (DEM) [14]. While this method enables detailed modelling it still requires a large amount of computational resources, especially when the number of particles increases. Typically, DEM simulations can take hours or even days for large particle assemblies. Therefore, in modelling bulk handling equipment (BHE) these techniques are typically used to evaluate how small design changes affect the behaviour of the particles in the equipment. This is defined as *local* optimization and has proven to be a successful approach in development of equipment. However, local optimization is concerned with a specific design and only explores a small section of the design space. Therefore it is likely to miss superior designs that can be found if the entire design space was evaluated. To bridge the gap between local and global evaluation of behaviour of bulk handling equipment, *metamodels* are an excellent option. These data-driven models of a computationally expensive model such as DEM, which can be used as an inexpensive surrogate. Metamodels can be used at a global level for model approximation, design space exploration, problem formulation, and optimization support [52].

In the past decade the applicability of DEM increased significantly due to the introduction of GPU and parallel computing [16,19]. Using a GPU results in a speed-up up to ten times [16], which makes it feasible to study large scale industrial systems and complex flows with DEM [29,31,32]. Strategies to further increase efficiency of DEM simulations include the use of hierarchical grid [20,21], stiffness reduction [34] and coarse graining or particle upscaling [33,43]. Still, there are limits to the speed-up that can be achieved. In design problems the amount of model evaluations is high which makes it computationally expensive despite speed-up measures. Moreover, bulk handling equipment (BHE) behaviour involves structural and kinematic responses for which coupling with numerical methods such as finite element (FE) and multibody dynamics (MBD) is required ([2,3,12,15,23,35,42,54,56]. Such coupling leads to a further increase in computational costs. Therefore, metamodels show great potential in facilitating usage of DEM in bulk handling equipment design procedures.

There are two main types of metamodeling approaches. The first is *model fitting* (MF) where a mathematical relation between scalar design parameters and key performance indicators (KPIs) is defined. Well known methods for model fitting are response surface methodologies (RSM) [25,52] and surrogate models [27]. The second approach is *reduced order modelling* (ROM) [4,36,44], where physical phenomena present in the system are modelled in a simplified manner while including spatial and transient information. Common methods in reduced

* Corresponding author.
E-mail address: m.p.fransen@tudelft.nl (M.P. Fransen).

https://doi.org/10.1016/j.powtec.2021.07.048

order modelling are projection based reduced order models (PROM) [5], balanced truncation [4], and moment matching [44]. The computational effort required to construct ROM models varies significantly depending on the nature and complexity of the modelled system.

In contrast to their widespread use in other fields, in literature on design of bulk handling equipment with DEM, metamodeling techniques are rarely used. A few related studies for bulk handling can be found within the chemical engineering field. These are focussed on operational parameters rather than on design [1,6,30]. In this field an increase of the use of metamodels is observed because these models can be included in flow sheet descriptions of chemical processes [40,46]. A combination of metamodeling techniques with computationally expensive particle based models such as DEM has been applied. Boukouvala [6] focussed on predicting the velocity profiles in a rotating drum by using a reduced order model. Rogers & Ierapetritou [46] suggest integration of reduced order and model fitting metamodels in flowsheets for unit processes in chemical engineering [46]. Barrasso et al. [1] studied the collision frequency in a continuously stirred reactor with a model fitted metamodel based on an artificial neural network (ANN). [57] used response surfaces to map segregation of particles based on DEM data. However, the scale, material properties, and shape of the materials modelled in these studies is far different from the materials used in bulk material handling where irregular shapes and wide, gapped particle size distributions are common.

These studies show the potential of metamodeling in combination with particle based models to predict behaviour of bulk handling equipment designs. Given the high potential of these techniques and the presently limited use for BHE applications, there is need for a metamodel construction or training procedure that ensures that accurate metamodels are obtained for a design purpose. Model fitting and reduced order modelling are difficult to combine in one training procedure. In design of BHE scalar design parameters and KPIs are commonly used. To find a mathematical relation between those, model fitting is the most adequate approach. Therefore the focus of this study is on a training procedure for model fitted metamodels.

Fig. 1 depicts the proposed framework for the use of DEM-based metamodels. First a DEM object model is developed after characterisation of the bulk handling problem. Instead of directly using this model in optimizing the equipment design, metamodels could be constructed after the DEM model development and before design optimization, as shown in Fig. 1. Here the DEM model is used to generate training data for the metamodel. The metamodel training procedure is shown on the right of Fig. 1. This starts with defining the design space and creating a sampling set for which the DEM model will generate the training data. Secondly, a suitable type of metamodel is chosen based on the distribution and expected trends in the data. Thirdly, hyper-parameters of the metamodel, i.e. additional parameters that affect the resulting shape, are optimized to obtain the most accurate metamodel for this data set. Finally, the model is validated using a validation strategy. Together, these steps form a systematic metamodel training methodology. After training of the metamodels is completed, they can be used in design space exploration, analysis, and optimization at low computational cost.

The aim of this study is to present, analyse and demonstrate the steps involved in methodically training DEM-based model fitting metamodels, with particular attention for intricacies related to the behaviour of bulk handling equipment. To illustrate the use of metamodels the training procedure is applied to a hopper design case. In this case study three common metamodels are evaluated: Polynomial Regression (PR), Radial Basis Function Interpolation (RBFI), and Kriging. Secondly, the design space is sampled, the effect of sample size is analysed and data is filtered. Next, the third step involves optimization of the hyper-parameters to obtain the most accurate metamodel. Lastly, in the fourth step the applied validation strategies are the validation set approach (VSA), k-fold cross-validation (K-fold CV) and leave one out cross-validation (LOOCV) with repetitions. Based on the results for these models recommendations for the use of metamodels in design with DEM models are are given.

Following this introduction, Chapter 2 starts with a general introduction to metamodeling and a detailed description on building a model fitting based metamodel. Additionally, the theory for the three metamodels used in the test case is described. Chapter 3 introduces the DEM model of the hopper used in this study and presents the analysis of the generated data. After sampling, hyper-parameter optimization, and validation techniques for the hopper case are evaluated. Subsequently, in Chapter 4 the results obtained from the metamodels are discussed, after which Chapter 5 presents, the conclusions and recommendations for further research.

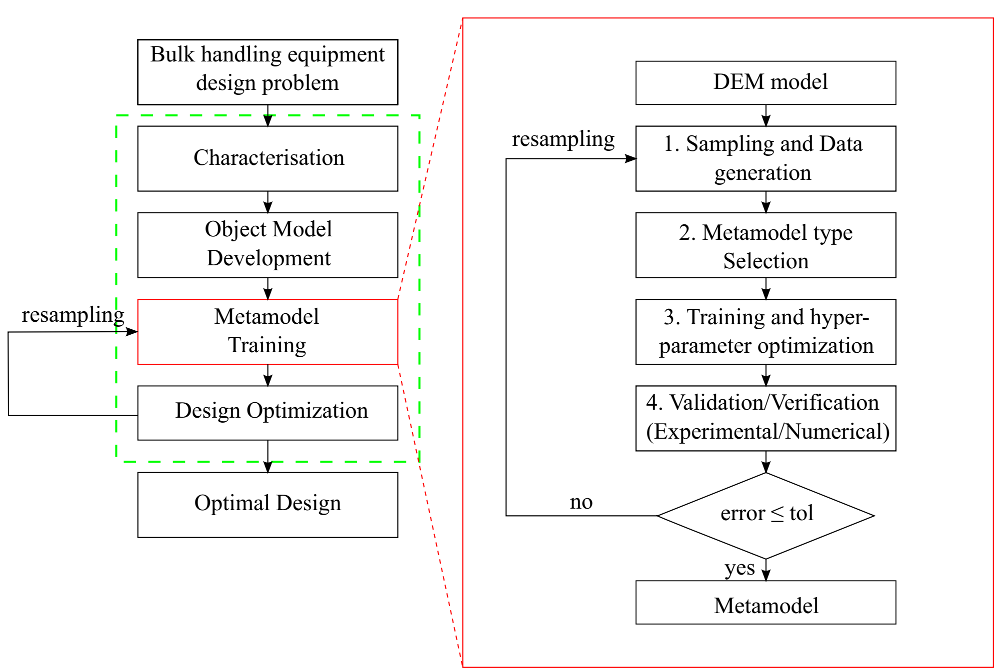

Fig. 1. Framework for bulk handling equipment design based on metamodeling, with the various steps involved in methodical metamodel training outlined on the right.

2. Metamodeling

2.1. Sampling and data generation

Development of a MF metamodel starts by identifying the design space in which the metamodel must be valid. The size and bounds of the design space are defined by the limits of the design problem. After choosing a suitable design space the general procedure to obtain training data is to perform a design of experiments (DoE) or design of simulations (DoS) as it is called if one uses a numerical model for data generation. In these processes sampling locations are generated for the defined design space, using methods such as Sobol sampling, Latin Hypercube Sampling (LHS), Hammersley Sequence Sampling (HSS), Monte Carlo Sampling (MCS), or Direct Sampling (DS) [52]. There are many methods to sample design spaces which all have their own benefits and pitfalls. The reader is referred to [58] for further details.

The type of method that should be used depends on the desired sample size and properties. However, the sampling set does not have to be generated in one single step. To reduce the computational costs of generating data, the sample set can be gradually expanded by resampling or adaptive sampling [39]. These have become popular methods to find a sufficient sampling while minimizing training data generation costs [18,52]. Especially in relation to DEM the use of adaptive and iterative sampling has been included in several studies [9,10]. To sample the design space further, supervised or unsupervised techniques can be used. In supervised adaptive sampling, new points are added based on evaluating the performance of the metamodel for the previous sampling set. Unsupervised sampling is based on adding samples according to a method such as LHS or grid sampling to improve the model by simply increasing the size of the sampling set. However, in case of DEM data, resampling with many iterations might not be convenient. DEM simulations take a considerable amount of time, and multiple resampling steps would increase the duration of sample generation. Therefore, it should be considered how many resampling iterations are acceptable and how many sample points are added in each iteration. It can be more efficient and faster to have fewer iterations with more sample points than adding a single sample point each iteration.

Moreover, it needs to be considered that metamodels tend to behave poorly at the edges of the design space because most metamodeling techniques are not able to extrapolate well. To improve the metamodel at the boundaries, either sample points should be chosen slightly outside of the domain of interest, or one has to densify the training data on the boundaries of the design space such that they are better defined [52]. However, because DEM simulations are computationally intensive densification of the sampling set is inconvenient. Therefore, sampling a space bigger than the domain of interest is the recommended way to ensure sufficient accuracy in boundary regions.

After the data has been generated for the sample set, it must be processed and analysed before continuing with the second metamodel training step. To increase the quality of the sample set the results from the simulations might require an intermediate step where the data is filtered. Invalid and inaccurate simulation results must be identified and removed, so that the metamodel training is not adversely affected by this data. Of course caution has to be taken when filtering data because there is a risk of leaving out data that is actually representative for the modelled system.

2.2. Metamodel selection

The second aspect of metamodel development is training of the metamodel. For the model fitting approach, this starts at constructing the function space or basis. The functions space contains basis functions such as polynomials, splines, or radial basis functions (RBF) [25,52]. The chosen basis functions should together be able to represent the trends that are present in the data. Methods such as Kriging [26,48], Gaussian process regression [45], artificial neural networks (ANN) [1] and radial basis function interpolation (RBFI) are suitable for capturing highly nonlinear trends and flexible in interpolation and filtering of data. A downside of polynomial regression (PR) is that these methods are based on less flexible polynomial basis functions [25]. On the other hand, this reduced flexibility of a polynomial basis can be exploited when dealing with irregular and noisy data, as it can provide a smoothing effect instead of exact interpolation. It must also be noted that compared to the computational expense of generating the DEM data the cost of training a metamodel is low. After defining the function space the task is to find the coefficients of these functions for which the metamodel fits the data best. Well known fitting methods are least squares regression, best linear predictor, log-likelihood, and multipoint approximation. The type of fitting method depends on how the optimal fit is defined and which technique is most suitable to find this fit.

In chapter 3 a numerical test case is presented involving two design parameters and two performance parameters. The three metamodels are built based on data obtained from a DEM simulation of a discharging hopper. Considered are a Polynomial Regression (PR), Radial Basis Function Interpolation (RBFI), and Kriging metamodel. These have been chosen because of their common use in engineering practice and because they are regression based, interpolation based, and a combination of interpolation and regression respectively. In the following subsections the foundation of these methods is discussed, including the basis-functions used to construct the models.

2.2.1. Polynomial regression

In Polynomial regression a polynomial function is fitted to a set of data points such that a response surface for the design domain is obtained. Even though this is a classical method it is still commonly used in developing response surfaces [17] because of its simplicity and smoothing capability. In this two-dimensional case study a response function f is represented as,

$$f(x_1, x_2) = \sum_{k=0}^{m} \sum_{l \leq k}^{n} a_{kl} x_1^k x_2^l \tag{1}$$

where x_1, x_2 are the polynomial dimensions and m and n are the order of the polynomial in dimensions x_1 and x_2, respectively. The polynomial which is fitted to the data consists of multiple terms which each have a coefficient a_{kl}. In the regression process the values for these coefficients are determined by finding the least squares solution of the mean squared error between the reference value and the predicted value of the polynomial in these training points.

2.2.2. Radial basis function interpolation

Radial basis function interpolation was first presented in [22] and was focussed on representing irregular surfaces with multi-variate functions. In RBF interpolation, a response function is represented by a summation of N radial basis functions $\phi(||\boldsymbol{x} - \boldsymbol{d}_i||)$ located at the training data points, d_i. We consider the commonly used inverse multi quadric radial basis function which is a full rank function which has a high information content,

$$\phi(||\boldsymbol{x} - \boldsymbol{d}_i||) = \frac{1}{\sqrt{1 + (\epsilon||\boldsymbol{x} - \boldsymbol{d}_i||)^2}} \tag{2}$$

Here the $||\boldsymbol{x} - \boldsymbol{d}_i||$ term is the distance from a location $\boldsymbol{x}$ to a training point $\boldsymbol{d}_i$. ϵ is the shape factor of the RBF and determines the width of the radial basis function. To make the RBFs represent the reference values $\boldsymbol{u}_i$ at the training points the correlation between each RBF must be calculated and the coefficients $\boldsymbol{b}_i$ need to be determined. This results in the following system,

$$\Phi_{ij}b_i = u_j \begin{bmatrix} \phi(r_{11}) & \cdots & \phi(r_{1N}) \\ \vdots & \ddots & \vdots \\ \phi(r_{N1}) & \cdots & \phi(r_{NN}) \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_N \end{bmatrix} = \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix} \quad (3)$$

which if solved, results in the coefficients $\boldsymbol{b}_i$. With the coefficients the following interpolation function can be formulated,

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} b_i \phi(||\boldsymbol{x} - \boldsymbol{d}_i||) \quad (4)$$

The predictor function $f(\boldsymbol{x})$ calculates the correlation between the points $\boldsymbol{x}$ and the training points and by multiplying with the coefficients $\boldsymbol{b}_i$ and summation of all contributions, the function value at locations $\boldsymbol{x}$ is predicted. The resulting function exactly reproduces the reference values at the training points, and smoothly interpolates between those.

2.2.3. Kriging

A Kriging model is a model based on both regression and interpolation. The concept of kriging has been developed by Krige [13,28] and finds its origins in geotechnical sciences. Currently there exist many variants and Kriging is a common technique to construct predictive models. A well-known Kriging toolbox is the DACE toolbox [49], which is also used in this study. Kriging is very flexible in fitting nonlinear data trendsbecause the covariances can be tuned by the sample data [25]. The Kriging predictor can be defined as follows

$$f(\boldsymbol{x}) = \sum_{i=1}^{k} c_i g_i(\boldsymbol{x}) + Z(\boldsymbol{x}), \quad (5)$$

and consists of a sum of regression components which are second order polynomials in the first term and a realization of a random stochastic process $Z(x)$ in the second term.

$$Z(\boldsymbol{x}) = \sigma_l^2 \phi(||\boldsymbol{x} - \boldsymbol{d}_i||) \quad (6)$$

Here $\phi(||\boldsymbol{x} - \boldsymbol{d}_i||)$ gives the covariances between the training points based on their Euclidian distance and σ_l is the process variance. Similar to RBF, various choices for ϕ are possible. In the Kriging model in this study the squared exponential Gaussian is used for calculating the covariance between the data points and the points that need to be predicted, given by:

$$\phi(||\boldsymbol{x} - \boldsymbol{d}_i||) = e^{-\theta_i ||\boldsymbol{x} - \boldsymbol{d}_i||} \quad (7)$$

Kriging is more computationally expensive than the RBFI and PR method because it needs to find a fit for the regression and the interpolation components of the model. Finding a good is generally achieved by maximizing the likelyhood of the fit which is a hyper-parameter optimization technique that is discussed in the next section.

2.3. Hyper parameter optimization

To improve the fit of a metamodel the parameters of the used basis functions can be optimized, commonly called hyper-parameter optimization. For the three models described in this section three different approaches are taken to optimize the parameters. For the polynomial regression model the parameter that needs to be optimized is the order of the polynomial in x and y direction, N_x, N_y respectively. It can be argued that this order should be as high as possible such that the more detail can be captured by the model. However, beside the possibility of overfitting, adding higher orders might result in oscillations in the response surface which decreases the accuracy of the metamodel, also known as the Runge phenomenon. Therefore it is recommended that metamodels based on polynomial regression are checked for these artefacts and base the order of the polynomial on the accuracy of the resulting metamodels. In this case the metamodels are built for zeroth to fifth order polynomials for both design parameters and the optimal coefficients correspond to the combination with the lowest root mean squared error.

In the RBFI model the optimization parameter is the support radius c of the RBF. Because the RBFs are located at the training points there will be exact interpolation. The only error at these points is the machine precision error of the system. Therefore the parameters need to be optimized using a different strategy. To identify the error of the metamodel, the RSME is determined by Leave One Out Cross Validation as used by [51]. The support radius c is optimized such that the RMSE is minimized. With this method it must be noted that it is computationally expensive.

The Kriging model which is based on the DACE toolbox [49] uses log-likelihood maximization of the metamodel to determine the optimal shape factor values. In the case of this Kriging model there is a shape parameter for each dimension, θ_1 and θ_2. The optimization of these parameters is implemented in the toolbox. This approach minimizes the process variance is which ensures that the reliability of the metamodel in between the training points is maximized.

2.4. Metamodel validation

A next essential step is validation of the metamodel. The goal of metamodel validation is to verify their ability to predict values in the design domain. There are three frequently used methods for validation: the validation set approach (VSA), k-fold cross validation (k-CV), and leave-one-out cross validation (LOOCV) [41]. In each method, the sampling set is divided into a training and validation subset, and metamodel predictions at the validation points based on the training set are compared against the validation values. The three approaches mainly differ in the way the samples are divided. In the VSA the sampling set is divided into a training and validation set according to a user-defined ratio. The k-CV divides the samples into k subsets of equal size which are combined to form k different cross validation sets. Finally, in LOOCV, a validation set consists of one sample point and all the remaining data points are put in the training set. This is repeated for the total number of data points. All three validation methods will be considered in the case study presented in the next section.

The random division of the sampling set in VSA and k-CV introduces some variability in the results, and additional methods exist to improve consistency. One method to mention is stratification, which ensures that the validation and training sets contain data points from every section of the domain. However, use of stratification requires heuristic information on the model. Another method to obtain consistent results from the validation strategies is to repeat the procedure with new randomized divisions. This gives an insight into the stability of the validation error prediction. In both the VSA and k-CV approach common validation set sizes are 10 and 20% of the entire data.

3. DEM-based metamodel test case: discharging hopper

3.1. DEM (object) model

In this case study the object model used is a semi-two-dimensional silo, shown in Fig. 2. This model has been built in MercuryDPM, an open source discrete element software package [53], The material properties that will be used for this study are fictive and only valid for this numerical example. In reality, every bulk material has to be characterized experimentally to find the correct values for its properties. Bulk properties are heavily affected by environmental conditions such as the humidity and temperature. Additionally, the particle size distributions and surface properties can differ between batches of material. As the focus of this case study is on demonstrating the process of constructing a DEM-based metamodel, these complications are not taken into account. The metamodeling techniques described in Sections 2.2.1–3 are applied to the data generated with the object model. For the hopper example

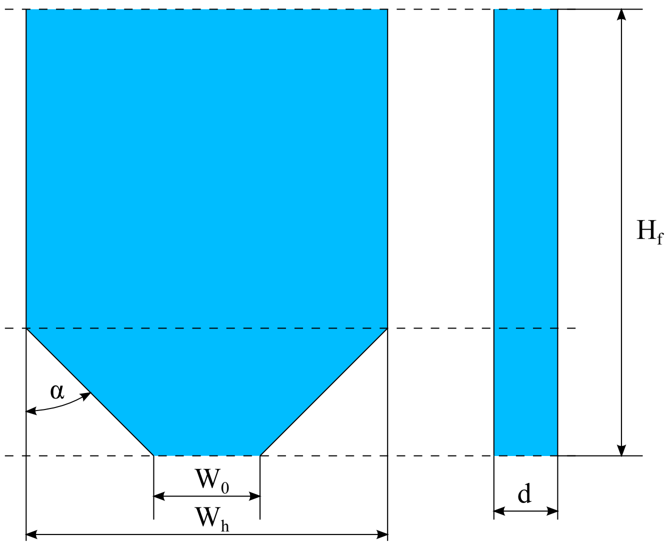

Fig. 2. The silo geometry used in the DEM model. α represents the angle of the hopper and W_o describes the width of the outlet opening. The silo width is described by W_h, silo depth by d, and the fill height of the silo by H_f.

the angle α and the discharge opening W_o are the design parameters. The KPIs are the discharge rate and coefficient of variation [50].

3.1.1. Model geometry description

The geometry of the silo is fixed except for the hopper angle α and discharge opening W_o, which are chosen as the design parameters. A cross section of the silo and its dimensions are shown in Fig. 2. In order for the silo model to have a feasible geometry the range for the hopper angle α is 10 to 100°. The size of the discharge opening ranges from 15 to 210 mm for a fixed silo width W_h of 0,6m as denoted in Table 1. The ratio of hopper width to discharge opening approximately equals 3 which meets the condition $W_h > 2{,}5W_o$ set for having constant discharge rate during hopper discharge [7].

3.1.2. Material description

The bulk material is modelled by spherical particles for which the particle size is described by a normal distribution with an average diameter of 8 mm and a standard deviation of 2,0 mm. Particle sizes in this range are common in bulk handling applications. However, because the time-step size depends on the smallest particle size, the particle size distribution is truncated to the range of 5 to 14mm to limit computation time. The density of the particles is set to 2500 $\frac{kg}{m^3}$ which is similar to the density of gravels and sands. The bulk stiffness E_b of the material is set at 70 MPa and is used to calculate the contact stiffness k following this relation [38],

$$k = \frac{E_b V}{C_n r_{avg}^2} \tag{8}$$

where V is the average particle volume, C_n the contact number, for loose packing equal to 4 [55], and the average particle radius r_{avg} of the particle size distribution.

Table 1
Geometric properties silo

Property	Value
α	$10-100°$
W_o	$15-210\ mm$
W_h	0,6 m
H_f	0,8 m
d	$5{,}3\times10^{-2}m$

The time step Δt is based on the response time t_c of the contact between two particles which is calculated as

$$t_c = \frac{\pi}{\sqrt{\frac{k}{m_{eff}} - \left(\frac{\gamma}{2m_{eff}}\right)^2}} \tag{9}$$

Here k is the contact stiffness, m_{eff} is the effective mass of the two copies of the smallest particle, and γ is the damping of the contact.

To ensure a stable simulation the time step for integration should be far smaller than the response time, $\Delta t << t_c$ [37]. A safe ratio that is commonly used in MercuryDPM for large scale simulations is $\Delta t = \frac{t_c}{10}$ [53].

3.1.3. Contact model description

In this study a linear visco-elastic friction contact model has been used to model particle-particle and particle-wall contact [38]. The particle-particle contact is shown in Fig. 3 and consists of two spring-damper combinations and a figure to represent the friction between the particles. The contact stiffness k has a component k_n in normal and k_t in tangential direction. For the damping of the contact normal and tangential components γ_n and γ_t apply. Friction between the particles is represented by sliding and rolling friction coefficients, μ_s and μ_r. The torsion in this model has been turned off to reduce the complexity of the contact model. The property values for each contact model can be found in Table 2. These settings will ensure that in the simulated hopper designs the dominant flow mode is core flow. The contact stiffness of the walls is, P-W1 and P-W2 are set to two times the contact stiffness of the particle-particle (P-P) interaction [11]. Where the side walls (P-W1) have a high friction coefficient of 0,5 and the front and back wall of the hopper (P-W2) have a lower friction coefficient of 0,3. The friction coefficient for the side wall is in the same range found for bonded iron particles on steel [8]. The friction value for the front and back walls is set to a lower value because we assume less friction on this wall. The damping in the entire system is the same for particles and walls and is equal to 0,3 $\left(\frac{Ns}{m}\right)$. In this case study we have assumed values for the contact properties. However, these properties can be determined with experiments on a micro scale directly and inversely through macro scale experiments. Examples are the wall friction coefficients by using a shear cell [24] or an inclined surface tester [8] and the bulk modulus and internal shear angle by means of a compression test.

3.1.4. Simulation settings

Before the start of the simulation the silo is filled by a randomized particle generator (Section 3.1.2) while the outlet remains closed. After starting the simulation the particles are allowed to settle and at $t = 1{,}5\ s$ all particles above the fill height are removed. The fill height is 0,8 m which ensures that the discharging time of the silo is sufficient for analysis of the granular flow. At $t = 1{,}6\ s$ the outlet opens and allows particles to discharge from the hopper. The total simulation time is set

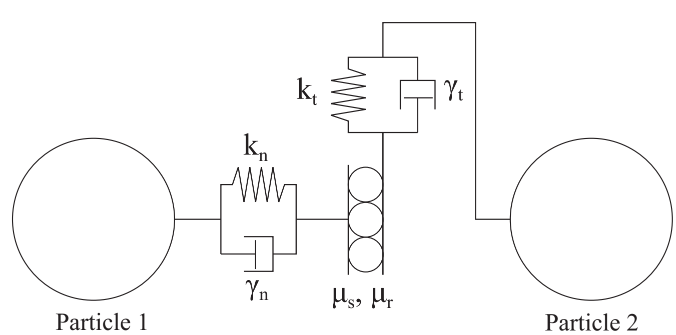

Fig. 3. Description of the normal, tangential, rolling, and sliding contact between the particles which is also used to model particle-wall contact.

Table 2
Properties of the contact between particle-particle, particle-wall 1, and particle-wall 2.

Property	P-P	P-W1	P-W2
μ_s	0,2	0,5	0,3
μ_r	0,2	–	–
$k_n\left(\frac{N}{m}\right)$	2,93 10^5	5,86 10^5	5,86 10^5
$k_t\left(\frac{N}{m}\right)$	$\frac{2}{7}k_n$	$\frac{2}{7}k_n$	$\frac{2}{7}k_n$
$\gamma_n\left(\frac{Ns}{m}\right)$	0,3	0,3	0,3
$\gamma_t\left(\frac{Ns}{m}\right)$	$\frac{2}{7}\gamma_n$	$\frac{2}{7}\gamma_n$	$\frac{2}{7}\gamma_n$

to 25 s to ensure that for all configurations emptying of the silo is achieved. A stopping criterion has been added to the model which stops when the ratio between the kinetic and elastic energy becomes smaller than 10^{-6}. This stopping criterion ensures that simulations are stopped when the flow of material stopped or the hopper is empty. Therefore unnecessary simulation steps are prevented. For stability of the simulation the time step is set to $8{,}7 \cdot 10^{-6}$ s which is equal to the contact time divided by ten, $\frac{t_c}{10}$.

3.2. DEM data analysis

The DEM simulations provide particle location and velocity information which are used to identify material flow characteristics in the hopper. In this case study the mean discharge rate ϕ and the coefficient of variation ψ of the discharge rate are used as the KPIs of the hopper. These values are calculated by using the method described by [50]. The average discharge rate coefficient of variation is determined in the steady flow region of the discharge as shown in Fig. 4. The data obtained from the simulations is filtered after determining the KPIs where the training points which have no flow are removed from the dataset.

3.3. Sampling

We assume that there is no prior knowledge on the behaviour of the DEM model. Therefore, it is desired to get a uniformly distributed sampling set which covers the design space equally. To obtain this set, the Sobol sampling technique is used because one of its properties is that it produces a highly uniform sampling of the domain. The design space which ranges from 20 to 90° hopper angles and 25 to 200 mm discharge opening sizes is the desired domain. In order to obtain sufficient samples near the boundaries, the design space ranging from 10 to 100° hopper angles and discharge opening sizes of 15 to 210 mm. Covering this with 100 samples results in 72 interior points and 28 exterior points, as shown in Fig. 5.

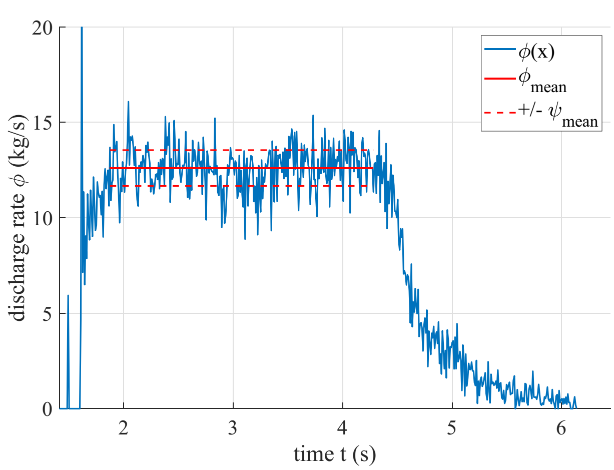

Fig. 4. Discharge rate versus time obtained from simulations data, red line is the average discharge rate and the red dotted lines represent the CoV.

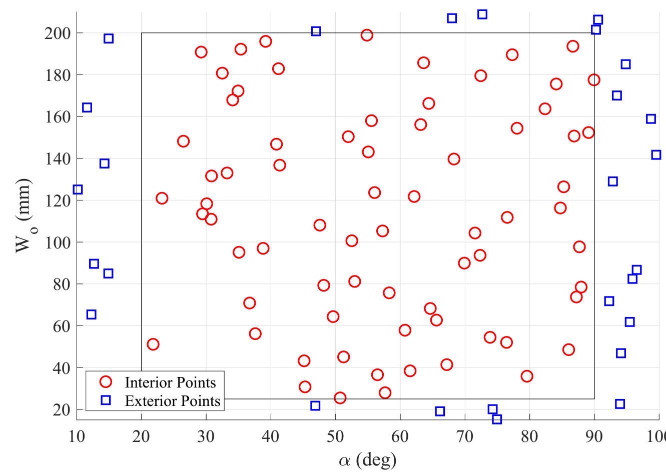

Fig. 5. Distribution as a result of Sobol sequence sampling.

3.4. Hyper-parameter optimization

In general it is difficult to manually determine the appropriate values of the hyper-parameters of the basis-functions which will result in an accurate metamodel. Therefore optimization of these hyper-parameters as described in Section 2.3 is an important step in metamodel construction. To demonstrate the effect these hyper-parameters have on the resulting metamodels a set of metamodels with predefined hyper-parameter values and a set with optimization hyper-parameter values are compared. Our aim is to highlight, by means of this example, that hyper-parameter optimization is important to construct high quality metamodels. In the case study initial models have been built with the set of hyper-parameter values shown in Table 3. Here, N_1 and N_2 are the order of the polynomials, c is the value for the shape factor of the multi-quadric radial basis function, and θ_1 and θ_2 are the shape factors of the basis functions of the Kriging model for the two design parameters.

3.5. Verification

For the metamodels in this paper the VSA, k-CV, and LOOCV validation strategies are performed where the root mean squared error (RMSE), given as

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(f - f^*)^2}{N}} \tag{10}$$

between the metamodel predictions and validation values is used as the error measure. For the VSA a 20 and 10% validation set size of all data points is evaluated. For k-CV the values k = 5 and k = 10 are used which means an equal subset size of 20 and 10% respectively. Both the approaches are repeated ten times to take the effect of random subset generation into account. The LOOCV method has to be run only one single time because it is deterministic, but consists of $N = 80$ individual validations. In Table 4 the sizes of the training and validation sets are

Table 3
Initial parameter values used without hyper-parameter optimization

	Optimized parameter	Discharge rate ϕ	Coefficient of variation ψ
PR (polynomial order)	N_1, N_2	2, 2	2, 2
RBFI (Inverse multi-quadric)	c	1	1
Kriging (Hyper-parameter correlation function)	θ_1, θ_2	1, 1	1, 1

Table 4
Validation strategies and applied settings

	Training set	Validation set	Number of iterations
VSA 10%	(90%)	(10%)	1
VSA 20%	(80%)	(20%)	1
k-CV 10	(90%)	(10%)	10
k-CV 5	(80%)	(20%)	5
LOOCV	N-1 data points	1 data point	N

shown for each method and the number of iterations that are included in the validation strategy.

4. Results

4.1. DEM simulation results

The data for training the metamodels has been obtained by running simulations with the model described in Section 3.1. The system used to execute the simulation sample is a DELL Precision 5820 with an Intel Xeon W-2223 CPU @ 3.60 GHz × 8 cores. The whole set of simulations took around 28 days in serial model using all 8 cores. The average simulation time was 53,8 h. However, the simulation time is geometry dependent. A simulation with a large discharge opening and low hopper angle is faster than one with a small discharge opening and high hopper angle. In Fig. 6 (a) a screenshot of a discharging hopper with an angle $\alpha = 47{,}6^{\circ}$ and a discharge opening $W_o = 108{,}1mm$ is shown. With the current wall friction settings and this specific combination of angle and discharge opening core flow is observed in the hopper. Moreover, on the left and right side of the hopper stagnant zones are visible where the particle velocity stays zero during discharge. Fig. 6 (b) shows the hopper with an angle $\alpha = 45{,}3^{\circ}$ and a discharge opening of 30,8 mm which results in arching of the material in the hopper and consequently no flow. The total number of simulations is 100 corresponding to the Sobol sampling of the design space shown in Fig. 5. With the simulation data, the discharge rate and coefficient of variation (CoV) have been calculated and are used for all the models in this section. The simulations results are shown as data points and contour plots in Fig. 7 with the discharge rate in (a, b) and the coefficient of variation in (c, d). These performance parameters are essential in hopper design, because in general the aim is to achieve continuous flow with low CoV [47]. In the figures the data points are denoted by dots. The black dots (80 data points) represent silo designs where there was flow in the silo, whereas the red dots (20 data points) represent the designs which have no flow. In this case study the aim is to develop metamodels that can predict flow conditions of hopper designs with a discharge rate of 2 kg/s and up. To avoid the steep transition from no-flow to flow regime, based on this analysis the model is trained only for the data points having flow and is only valid for discharge rates higher than 2 kg/s.

As seen in Fig. 7 (a), the discharge rate increases monotonically with the discharge opening W_o. Along the α-axis the hopper angle is shown where the data seems to follow a more constant level. This indicates that the discharge rate mainly depends on the size of the discharge opening and that the hopper angle has a limited effect. This is in line with the theory on hopper flow by Schulze [47]. In Fig. 7 (b) the data in (a) is represented by a contour plot where the isolines show the same trend. Moreover, in the 25 to 60mm zone for all angles the transition from flow to no-flow is visible by the change from black to red dots. Decreasing the size of the discharge opening causes the formation of arches in the hopper. These arches continuously collapse until they become stable at the transition from no-flow to flow. This can also be seen in the CoV which increases when the discharge opening becomes smaller. To prevent arch formation in designs a minimum discharge opening is used which is equal to 8–10 times the average particle size [47]. We use a truncated particle size distribution with an average particle diameter $r_{avg} = 10{,}2\ mm$. Using the lower bound of 8 times, would result in a discharge opening of 81,6 mm which safely ensures flow of the material even above the $2\frac{kg}{s}$ threshold as can be seen in Fig. 7 (b).

In Fig. 7 (b) a dependency of the discharge rate on the hopper angle is visible by a curve shape in all isolines. In the transition area between 50 and 70° hopper angles and above the 150mm discharge opening a shift in discharge behaviour is observed where discharge rate for 9, 11, and 13 $\frac{kg}{s}$ isolines show a move in the upward direction which is different from the smoother curves at lower discharge rates. This behaviour can be explained by the behaviour at the walls for the lower hopper angles. In the hopper model a wall friction coefficient, $\mu_{p,\ w1} = 0{,}5$, between particles and wall is defined which corresponds to a wall friction angle of 26,6°. We would expect that sliding of the material along the wall will stop or decrease at hopper angles 63,4° and higher. When the angle stays below 63,4° the wall friction force is likely to be lower than the force exerted by the particles on the wall, which enables flow along the walls. If the hopper angle becomes higher than 63,4°, stagnant zones will form which shifts the sliding interface from

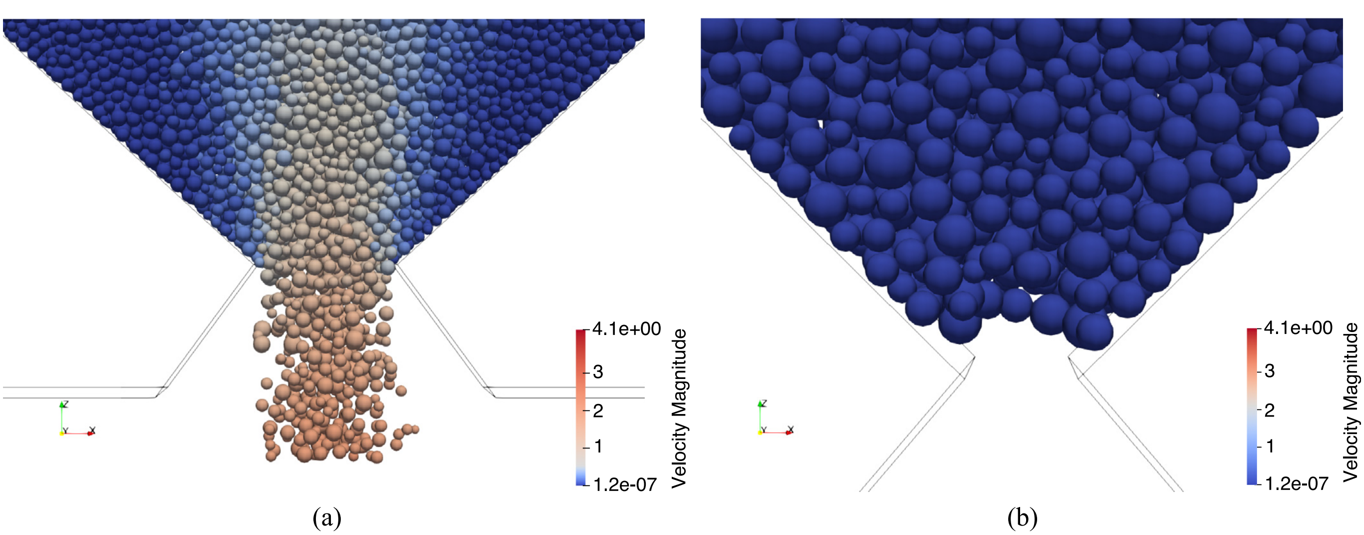

Fig. 6. DEM simulation stills (a) Core flow hopper discharge ($\alpha = 47{,}6^{\circ}$, $W_o = 108{,}1$ mm) and (b) an example of no flow due to arching ($\alpha = 45{,}3^{\circ}$, $W_o = 30{,}8$ mm).

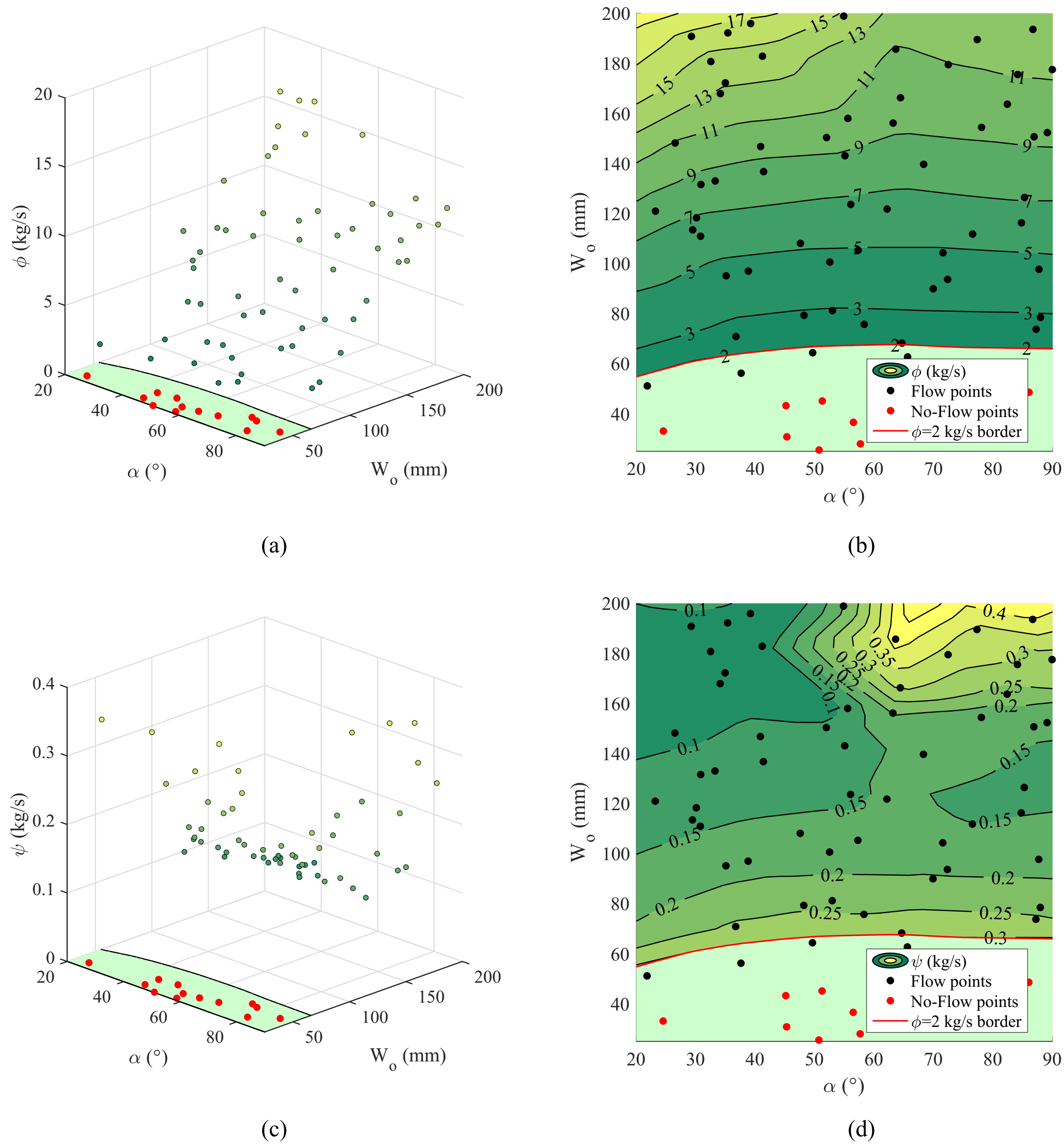

Fig. 7. Results of discharge rates ϕ with (a) the data points and (b) a contour plot, coefficient of variation ψ results in (c) 3D plot of data points, (d) contour plot.

particle-wall to particle-particle and therefore changing the flow behaviour.

Fig. 7 (c) shows the data points for the second KPI, the coefficient of variation ψ for the discharge rates in Fig. 7 (a). The fluctuations in the discharge rate show a different dependency on the design parameters than the discharge rate. However, the area in Fig. 7 (b) depicting the change in flow behaviour also shows changes in the CoV because of the accumulation of isolines in that area. As with the discharge rate this can be attributed to the change from particle-wall to particle-particle interface. Based on the CoV data it can be seen that for angles below the transition area the CoV becomes lower and therefore the discharge rate becomes more stable. In the transition area the CoV starts to increase for increasing hopper angles which results in unstable discharge. In the 60 to 150*mm* region we can see a valley in which the CoV increases when the discharge opening becomes smaller for all angles although at a different rate. With smaller discharge openings and at higher hopper angles the flow becomes less stable.

4.2. Effect of hyper-parameter optimization on metamodel quality

In Fig. 8 the three metamodels for the discharge rate without hyper-parameter optimization are shown. The PR metamodel in Fig. 8 (a) has been built with a second order polynomial for the two design parameters. This figure shows a curved surface fitted through the data, which is below the data points for a hopper angle between 45° and 90° and lies above the data points for lower angles. The ability of a PR model to fit to the data highly depends on the trends in the data, distribution of samples over the design space and the order of the polynomial [52]. In Fig. 8 (b) and (c) the RBFI and Kriging metamodels are shown where irregularities in the surface are present between 150 and 200 *mm* and hopper angles between 50 and 70° and for the Kriging model

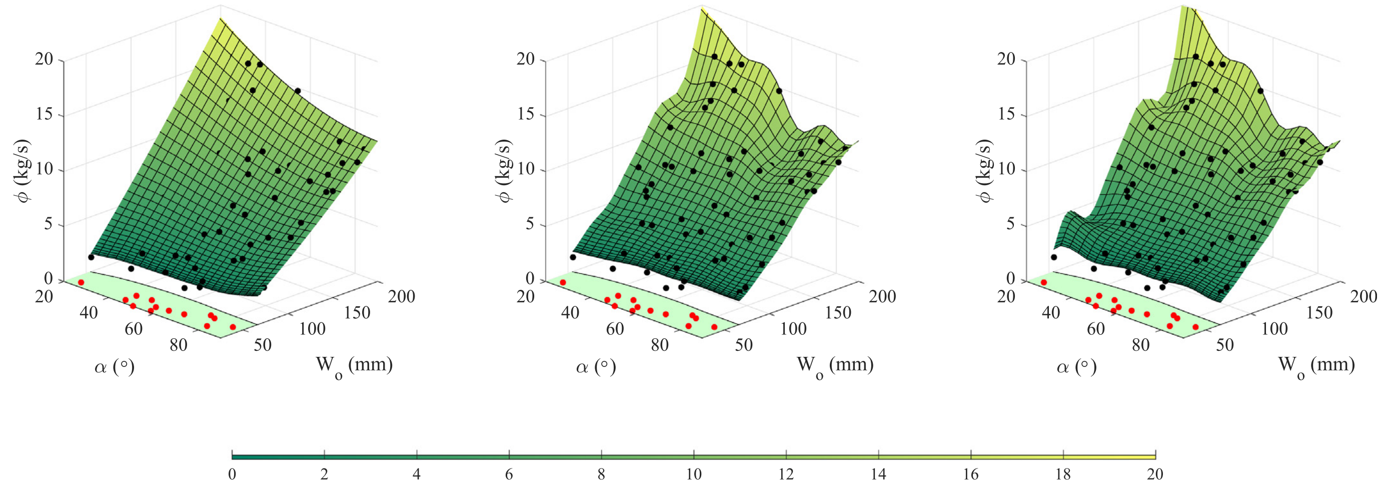

Fig. 8. Metamodels without hyper-parameter optimization for the average discharge rate (a) Polynomial Regression (b) Radial Basis Function Interpolation (c) Kriging

we see fluctuations at the 50 *mm* and 20° point. These undulations present in the RBFI and Kriging metamodel might be caused by the inappropriate values for the shape parameters of the basis-functions. Concluding, with polynomial regression a smooth surface is obtained. The RBFI and Kriging models in (b,c) show more local fluctuations when the distance between data points increases but are capable of capturing both the nonlinear behaviour at smaller discharge openings and the global trend of the dataset.

The metamodels based on the CoV data are shown in Fig. 9. In Figure (a) the PR metamodel shows that the fit of the polynomial surface is able to capture the trend of the data on a global level but the regression function is not able to capture local detail in the data set. The RBFI and Kriging models show the same behaviour as with the discharge rate. Several fluctuations are visible along the 200 *mm* line for all angles as well as for the 20° line, which indicate that the shape parameter is too small to capture the actual curve.

To obtain the best possible metamodels, hyper-parameter optimization was carried out for 50 random initial guesses where the best performing parameter values were chosen as the optimal hyper-parameter values (Table 5). For the PR metamodel it was found that the fifth order polynomial should be used for both design parameters and the KPIs. The optimal RBF shape parameter of the CoV is larger than the one for the discharge rate, because of the presence of flat areas in the CoV surface, which requires an RBF with a larger radius. For the Kriging model we can see the same behaviour as for the RBFs, where a smaller value for theta results in a narrow Gaussian whereas a larger value gives a wider Gaussian. Unlike the RBF, the Gaussian has two shape parameter values for each performance parameter, one in the direction of each design parameter.

Table 5
Optimized Hyper-parameters

	Optimized parameter	Discharge rate	Coefficient of Variation
PR (polynomial order)	N_1, N_2	5, 5	5, 5
RBFI (Inverse multi-quadric)	c	2.05	4.09
Kriging (Hyper-parameter correlation function)	θ_1, θ_2	7.98, 10.16	4.19, 19.04

Using the results of the hyper-parameter optimization, new metamodels have been trained for the hopper data set. Fig. 10 shows the results for the discharge rate, together with the data points. While based on the same data, the three models are different compared to the default metamodel results. The higher order polynomial enables the PR metamodel to fit better to the trend in the data. With the RBFI

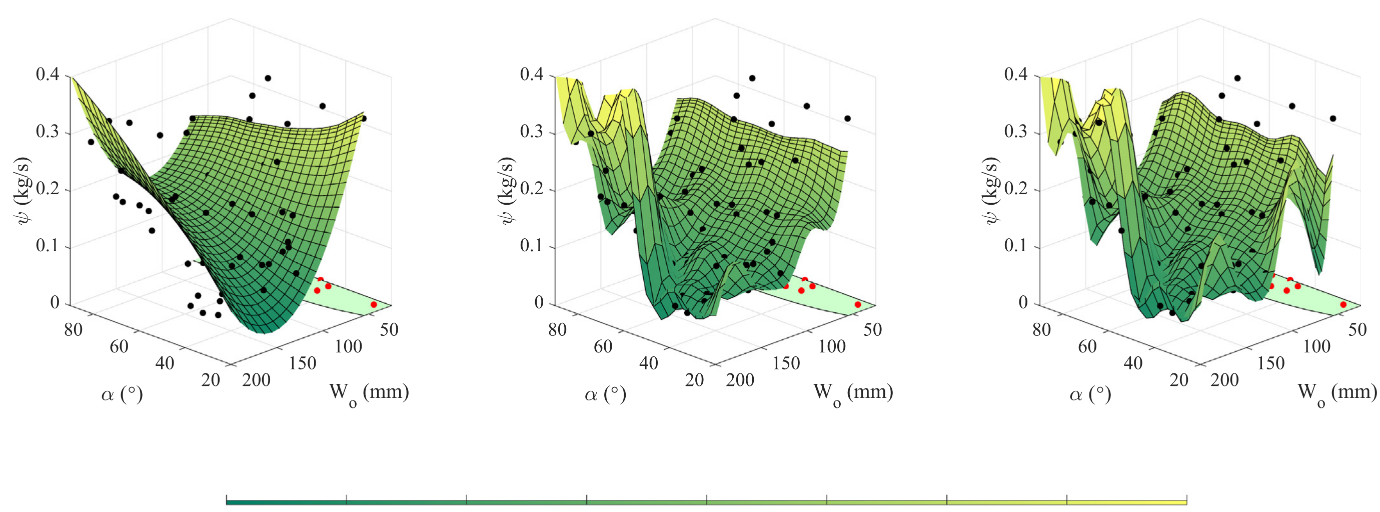

Fig. 9. Metamodels without hyper-parameter optimization for the coefficient of variation (a) Polynomial Regression (b) Radial Basis Function Interpolation (c) Kriging

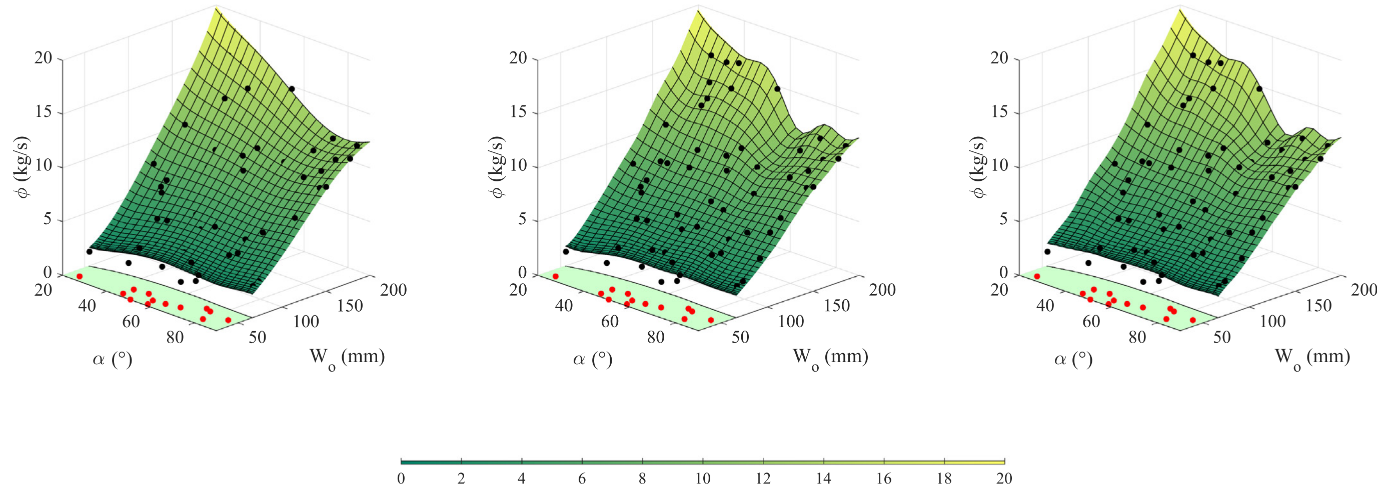

Fig. 10. Metamodels for average discharge rate (a) Polynomial Regression (b) Radial Basis Function Interpolation (c) Kriging

and Kriging metamodels, the fluctuations in the surface are reduced or absent. All three models show a transition zone for large discharge openings (>150*mm*) and angles between 50 and 70°. However, the PR model is less able to describe this transition zone compared to RBFI and Kriging models because it is less capable to capture local changes in the trend.

The dataset containing the CoV at the data points and the corresponding metamodels are shown in Fig. 11. All metamodels are able to capture the global trend. However, the PR model is not able to capture the local changes in the data set but produces a smooth trend surface. In addition, at the 20° and 200*mm* point the PR model will predict negative CoV values which are infeasible. The RBFI and Kriging models do show a more irregular surface but are able to capture the local behaviour better. The effect of hyper-parameter optimization on the resulting metamodels can also be achieved by adjusting the hyper-parameters through trial and error until the quality of the model is maximized. However, hyper-parameter optimization with the described methods is far more efficient and will become even more convenient when the number of design and performance parameters increases.

To a certain extent, metamodels are able to predict the behaviour of the discharge rate and CoV within the bounds of the design space. It needs to be realized that all models developed on a data set need to be evaluated on their ability to predict the actual behaviour. In the case of low dimensional problems it is possible to visualize the data but when the dimensionality increases this becomes more difficult. Therefore, quantitative measures are required, which will be discussed in the following section.

4.3. Metamodel validation

The validation strategies introduced in Section 2.4 have been evaluated to determine the accuracy of the PR, RBFI, and Kriging metamodel in predicting values at unknown design points. The three models used to evaluate the validation strategies are trained with the optimized parameters presented in Table 5 using the percentages of the 80 sample dataset denoted in Table 4. All the strategies have been repeated 10 times to get a measure on the reliability of the measured errors. All of these strategies give insight on how well the metamodels performs at predicting KPI values at new design points.

The bar charts in Fig. 12 and Fig. 13 show the average RMSE between the validation set values and the metamodel predictions for the discharge rate and coefficient of variance, respectively. As a results of the 10 repetitions the variance of the average RMSE can be shown. For the discharge rate results in Fig. 12 it can be seen that for the VSA and kCV approaches the prediction error is in the order of 4 to 7,5% and for the LOOCV is below 0,2%. In terms of the variance it can be seen that it is

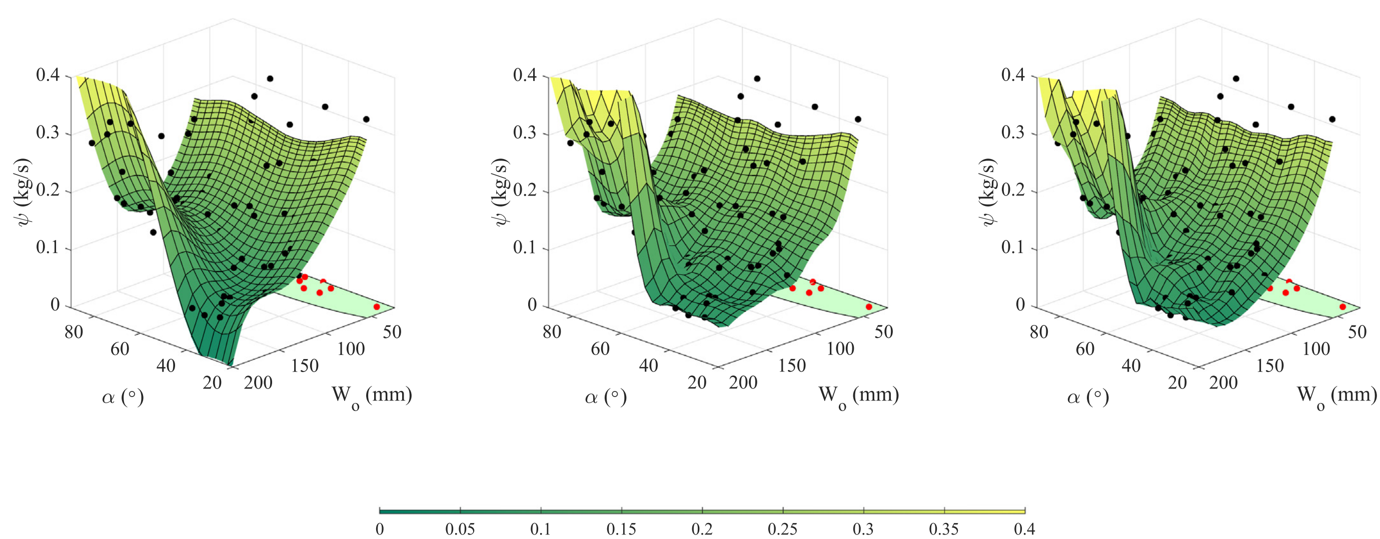

Fig. 11. Metamodels for the coefficient of variation (a) Polynomial Regression (b) Radial Basis Function Interpolation (c) Kriging

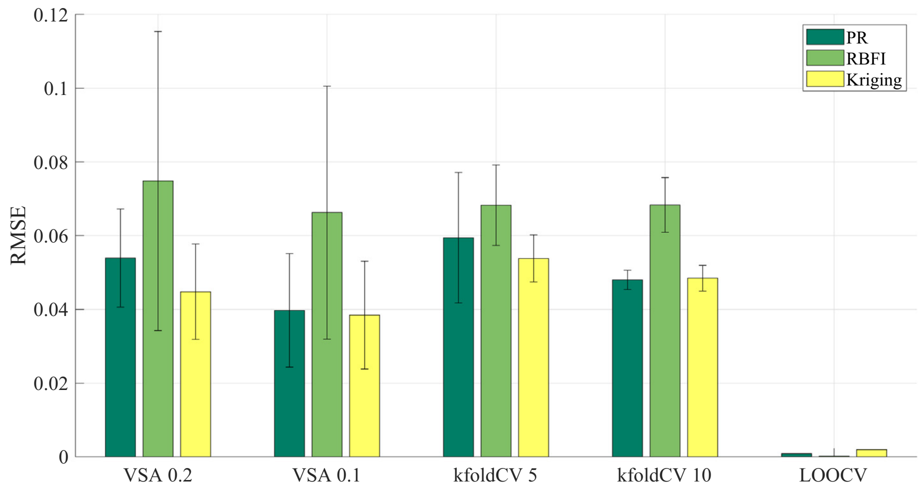

Fig. 12. Validation strategy errors for the discharge rate ϕ

nearly absent for the LOOCV approach but that it is larger for the VSA approach then for the kCV approach. For the CoV results in Fig. 13 we see that the average error is much higher and ranges from 15 to 20% which can be explained because the trends in the CoV data are more complex than those in the discharge rate data. If data near the validation points is not included in the training set it becomes more difficult to predict, especially when the behaviour is non-linear. For the LOOCV approach we see a small error of at most 1,5%. In terms of variance the same effect can be observed as with the discharge rate where the variance of the VSA approach is higher than the kCV variance. Results for both KPIs show that prediction errors are large if a data set of 72 (90%) or 64 (80%) points are used. This suggests that additional data points should be generated.

As mentioned, the major difference that can be observed for both results is that the variance of the kCV approach is smaller than of the VSA approach. This indicates that the kCV approach is more reliable in giving insight on the validity of a metamodel than the VSA approach. Compared to the kCV and VSA approach the LOOCV shows that the errors given by the metamodels is very small. Here it is important to consider that with the LOOCV approach more data points are used for training a metamodel compared to the kCV and VSA approach. As a consequence, if the ratio between number of validation points and training points becomes too small the effect of leaving one data point out will reduce and therefore lead to low validation errors.

To evaluate the validation error the LOOCV method can be used if the training data set is small, in the order of 50 data points. For larger data sets one should use the VSA or kCV approach where the kCV approach gives the most reliable validation error. In terms of time consumption the LOOCV approach is the most expensive looping through all the data points. Next is the kCV approach which uses *k* iterations in determining the validation error, therefore computational expenses increase with *k*. Followed by the VSA approach which only evaluates the validation error of a single division of the data into a training and validation set. With respect to DEM-based metamodels the computing time of these validation errors is irrelevant due to the cost of DEM-data generation. Evaluating metamodels by multiple validation strategies leads to a more complete insight on their accuracy and allows a designer to make a better choice for the type of metamodel that is going to be used.

4.4. Effect of sample size on accuracy

Generating DEM data is computationally expensive, therefore insight on the effect of sample size on the quality of the metamodel is required. To study the effect of sample size on the RMSE of the metamodel we gradually build the three models by increasing the sample size from 1 to 80 by 1. These 80 samples are the points in the data set representing material flow. After each increase in sample size a training set is drawn from the full dataset to train the three models, using the metamodel

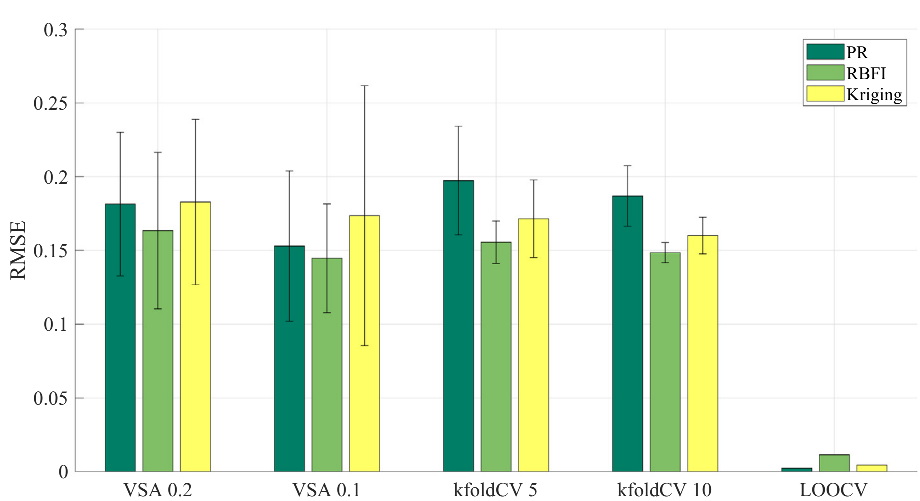

Fig. 13. Validation strategy errors for the coefficient of variation ψ.

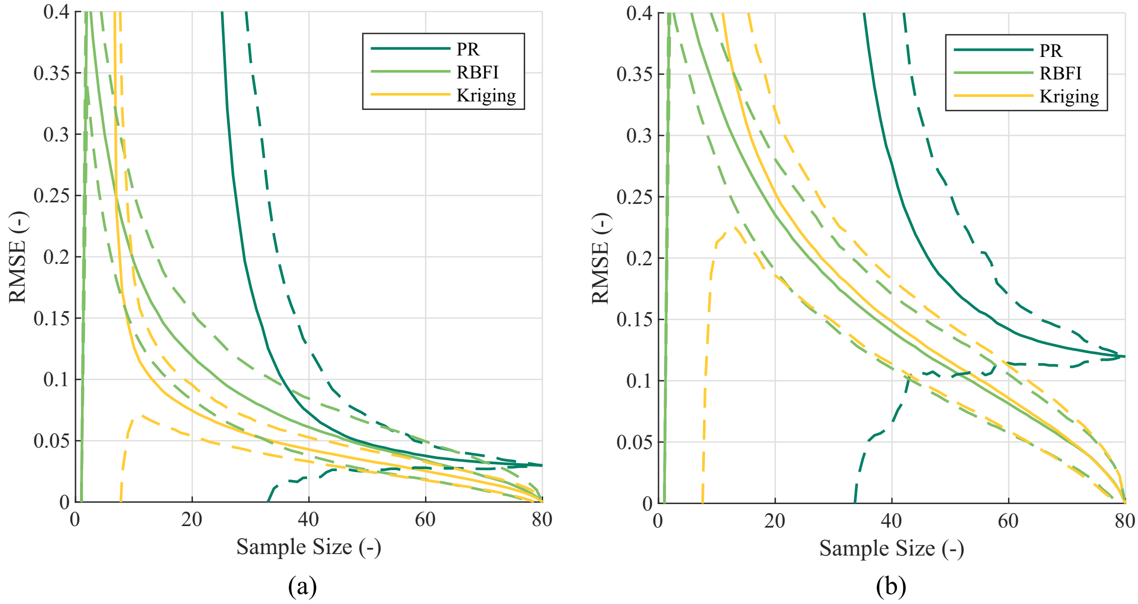

Fig. 14. The effect of sample size on the RMSE of the metamodel (a) averaged RMSE vs. sample size for the discharge rate (b) averaged RMSE vs. sample size for the coefficient of variation where in both figures the dashed lines represent the standard deviation

training procedures discussed in Section 2.2. To avoid any bias of the order of the subset, this process has been repeated 1000 times where for each repetition the order of the subsets is changed randomly. Finally, the average RMSE is calculated. In Fig. 14 the development of the averaged RMSE is shown for the Polynomial Regression, RBFI, and Kriging metamodels of both the discharge rate and CoV.

Note that some models cannot be built beneath a certain sample size. The PR metamodel uses a 5th order polynomial fit, which can only be determined when more than 21 data points for a two-variable problem are available because that equals the amount of coefficients. However, Jin [25] suggests that the amount of samples should be at least twice or three times the amount of coefficients of the polynomial to obtain accurate metamodels. Building an RBFI model is already possible from a single sample. However, low sample numbers will not lead to representative metamodels. The Kriging model built with the DACE toolbox requires a minimum of 5 data points for training based on the number of undetermined coefficients for the second order regression part of the model. For the RBFI and Kriging models it can also be seen that the error goes to zero at a sample size of 80. This occurs because both RBFI and Kriging have almost exact interpolation at the data points, therefore the error in the data points is at machine precision.

Fig. 14 (a) and (b) show that in all models the error reduces for increasing sample sizes. For the discharge rate, the Kriging model performs better over the entire range compared to the PR and RBFI models. It can be observed that the RBFI model outperforms the PR model over the entire range but performs similarly at a sample size of 50 data points. The results for the coefficient of variation data show that the PR model is not able to get an accurate prediction of the CoV while, both the RBFI and Kriging model show a large increase in quality when the sample size increases. Note also that the PR fit quality with lower sample numbers shows a larger standard deviation, indicating a stronger dependence on the selected design points.

For DEM-based metamodels it is essential to know the amount of samples required to reach a certain quality level of a metamodel. This depends on the accuracy that is required for the prediction of mean performance parameters. Table 6 shows the number of points required to reach a 2 and 5% error of the model. The PR model only reaches the 5% error limit for the discharge rate but is not able to get to 2%. The RBFI and Kriging model perform similar and are able to reach the thresholds with this dataset. However, the Kriging model requires less data points to reach the 5% and 2% threshold than the RBFI model for the discharge rate. For the CoV they require the same amount of data points. The better performance of the Kriging metamodel can be related to the basis of training the model which is minimizing the global process variance of the Kriging model whereas for the RBFI model the error in the data points is used which does not imply global optimality.

Based on the results of this test case it is advised to start with a sufficient sample size of 50 data points and gradually expand the amount of data points until the desired quality is reached. However, this number changes when the number of design parameters and KPIs changes. If the dimensionality (number of variables) increases the sample size should be increased accordingly to maintain a sufficient sampling density. To determine if more points need to be added the effect of sample size should be used. The process of expanding the sample set is referred to as adaptive or sequential sampling in literature [39,52]. In the case of DEM-based metamodels the time required for generating data is much higher compared to training, validating, and updating of the metamodel itself. Therefore an approach where some time is invested in determining the quality of the metamodel based on the sample size before additional simulations are started is most efficient.

4.5. Summary of findings

The three metamodels used in this study showed that hyper-parameter optimization is an essential step for obtaining accurate metamodels, regardless of the metamodel type. Hyper-parameter optimization can be performed on a trial and error basis but automatic hyper-parameter optimization is preferred.

Table 6
Minimum amount of data points required for 2 and 5% accuracy levels

Discharge rate ϕ	PR	RBFI	Kriging
2%	–	75	70
5%	45	45	32
Coefficient of variation ψ	**PR**	**RBFI**	**Kriging**
2%	–	78	78
5%	–	70	70

The results in this section show that it is necessary to determine which data is useful and relevant before building a metamodel. Filtering or excluding specific data might lead to more accurate metamodels but caution has to be taken because the data set may become less representative for the process.

In order to acquire accurate metamodels a representative error measure should be used to assess the quality. The results show that the use of k-fold CV gives reliable information on the validity of the metamodels. A single metamodel type that fits all datasets is not found which makes it worthwhile to test several metamodels and even have a different type for each performance parameter. Although this is time consuming the time spent is small compared to the time spent on developing the DEM model and generating the data through simulations.

To determine if the size of the sampling set is sufficient the effect of sample size and the validation error should be evaluated. Based on these errors the need for more data can be determined. To start metamodel construction, a sufficiently large data set should be trained to have a basis, in this hopper case study with two design parameters this is 50 data points. In additional resampling steps the size of the sampling set can be expanded.

The results for the three metamodels used in this case study showed that the Polynomial Regression model was the least accurate model and could not reach validation errors less than 3% for the discharge rate and 12% for the CoV. The Kriging model performed better for the discharge rate than the RBFI model while both models performed nearly identical for the CoV. Therefore, unless a particular polynomial trend is expected, RBFI and in particular Kriging should be preferred for their efficiency and generality. A 5% error for the discharge rate was reached with a small amount of data points whereas the models for the CoV required at least 70 points. Based on the results it can be seen that increasing complexity of the trends in the data requires larger data sets if accurate metamodels are desired.

5. Conclusion

In this study a methodology for constructing DEM-based metamodels has been presented and demonstrated on a case study. Different metamodels were trained and the effect of hyper-parameter optimization, sample size, and validation strategy was analysed for the first time in context of DEM. From this study it can be concluded that DEM-based metamodels can aid in revealing and understanding trends in the performance of bulk handling equipment in relation to selected design parameters, at acceptable computational cost. In using metamodels combined with DEM it is not advised to universally apply one single type of metamodel. The behaviour of performance parameters might match certain metamodel types better than others. Moreover, metamodel training is far less computationally demanding compared to the DEM data generation phase. Therefore it is advised to evaluate several types of metamodels and use the most adequate type for each performance parameter. To further increase the quality of metamodels hyper-parameter optimization should be applied to obtain the best possible metamodel for a given data set.

As a proof of concept, we analysed and validated the application of three model fitting metamodeling techniques using a representative BHE example: polynomial regression, radial basis function interpolation, and Kriging, and showed the ability of these methods to capture the discharge behaviour and the coefficient of variation of a silo in a wide design space. In this study the Kriging model performed best in predicting the discharge rate whereas the Kriging and RBFI models were better in predicting the coefficient of variance. Polynomial regression showed the strongest smoothing behaviour, which may be desired in case of noisy datasets. The overall results show that metamodels based on these techniques provide an sufficiently accurate representation of the bulk handling equipment behaviour for use in the equipment design process.

In relation to the design of BHE it is essential that metamodels with high accuracy can be trained for small or limited data sets because of the computational burden of DEM simulations. In this study the focus was on obtaining accurate predictions for average performance values. However, in bulk handling processes the behaviour of bulk material is stochastic by nature. Therefore, further research is required in training of metamodels including stochastic data such that this information can be included in exploring design options.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

Department of Maritime and Transport Technologies, TU Delft: Resources, Funding.

References

[1] D. Barrasso, A. Tamrakar, R. Ramachandran, A reduced order PBM – ANN model of a multi-scale PBM – DEM description of a wet granulation process, Chem. Eng. Sci. 119 (2014) 319–329, https://doi.org/10.1016/j.ces.2014.08.005.
[2] G.K.P. Barrios, R.M. De Carvalho, A. Kwade, L. Marcelo, Contact parameter estimation for DEM simulation of iron ore pellet handling, Powder Technol. 248 (2013) 84–93, https://doi.org/10.1016/j.powtec.2013.01.063.
[3] G.K.P. Barrios, L.M. Tavares, A preliminary model of high pressure roll grinding using the discrete element method and multi-body dynamics coupling, Int. J. Miner. Process. 156 (2016) 32–42, https://doi.org/10.1016/j.minpro.2016.06.009.
[4] P. Benner, M. Ohlberger, A. Cohen, K. Willcox, Model Reduction and Approximation: Theory and Algorithms. Model Reduction and Approximation, 2017https://doi.org/10.1137/1.9781611974829.
[5] B. Besselink, U. Tabak, A. Lutowska, N. Van De Wouw, H. Nijmeijer, A comparison of model reduction techniques from structural dynamics, numerical mathematics and systems and control, J. Sound Vib. 332 (19) (2013) 4403–4422, https://doi.org/10.1016/j.jsv.2013.03.025.
[6] F. Boukouvala, Y. Gao, F. Muzzio, M.G. Ierapetritou, Reduced-order discrete element method modeling, Chem. Eng. Sci. 95 (2013) 12–26, https://doi.org/10.1016/j.ces.2013.01.053.
[7] R.L. Brown, J.C. Richards, Profile of flow of granules through apertures, Trans. Instit. Chem. Eng. 38 (1960) 243–250.
[8] G. Chen, Surface Wear Reductuon of Bulk Solids Handling Equipment Using Bionic Design, TU Delft University, 2017https://doi.org/10.4233/uuid.
[9] H. Cheng, T. Shuku, K. Thoeni, P. Tempone, S. Luding, V. Magnanimo, An iterative Bayesian filtering framework for fast and automated calibration of DEM models, Computer Methods in Applied Mechanics and Engineering, 2018 , January.
[10] H. Cheng, T. Shuku, K. Thoeni, H. Yamamoto, Probabilistic calibration of discrete element simulations using the sequential quasi-Monte Carlo filter, Granul. Matter 20 (1) (2018) 1–19, https://doi.org/10.1007/s10035-017-0781-y.
[11] C. Coetzee, Calibration of the discrete element method: strategies for spherical and non-spherical particles, Powder Technol. 364 (2020) 851–878, https://doi.org/10.1016/j.powtec.2020.01.076.
[12] C.J. Coetzee, D.N.J. Els, G.F. Dymond, Discrete element parameter calibration and the modelling of dragline bucket filling, J. Terrramech. 47 (1) (2010) 33–44, https://doi.org/10.1016/j.jterra.2009.03.003.
[13] N. Cressie, The origins of Kriging, Math. Geol. 22 (3) (1990) 239–252.
[14] P.A. Cundall, O.D.L. Strack, A discrete numerical model for granular assemblies, Geotechnique (29) (1979) 47–65, https://doi.org/10.1680/geot.1979.29.1.47.
[15] D. Forsström, P. Jonsén, Calibration and validation of a large scale abrasive wear model by coupling DEM-FEM local failure prediction from abrasive wear of tipper bodies during unloading of granular material, Eng. Fail. Anal. 66 (2016) 274–283, https://doi.org/10.1016/j.engfailanal.2016.04.007.
[16] J.Q. Gan, Z.Y. Zhou, A.B. Yu, A GPU-based DEM approach for modelling of particulate systems, Powder Technol. 301 (2016) 1172–1182, https://doi.org/10.1016/j.powtec.2016.07.072.
[17] J.D. Gergonne, The application of the method of least squares to the interpolation of sequences, Hist. Math. 1 (4) (1974) 439–447, https://doi.org/10.1016/0315-0860(74)90034-2.
[18] D. Gorissen, T. Dhaene, A surrogate modeling and adaptive sampling toolbox for computer based design, J. Mach. Learn. Res. 11 (2010) 2051–2055.
[19] N. Govender, D.N. Wilke, S. Kok, Blaze-DEMGPU: modular high performance DEM framework for the GPU architecture, SoftwareX 5 (2015) 62–66, https://doi.org/10.1016/j.softx.2016.04.004.
[20] N. Guo, J. Zhao, A coupled FEM / DEM approach for hierarchical multiscale modelling of granular media, Int. J. Numer. Methods Eng. 99 (2014) 789–818, https://doi.org/10.1002/nme.

[21] N. Guo, J. Zhao, Parallel hierarchical multiscale modelling of hydro-mechanical problems for saturated granular soils, Comput. Methods Appl. Mech. Eng. 305 (2016) 37–61, https://doi.org/10.1016/j.cma.2016.03.004.
[22] R.L. Hardy, Multiquadric equations of topography and other irregular surfaces, J. Geophys. Res. 76 (8) (1971) 1905–1915, https://doi.org/10.1029/JB076i008p01905.
[23] G. Hess, C. Richter, A. Katterfeld, Simulation of the dynamic interaction between bulk material and heavy equipment: calibration and validation, ICBMH 2016 - 12th International Conference on Bulk Materials Storage, Handling and Transportation, Proceedings, July 2016, pp. 427–436.
[24] A.W. Jenike, Storage and flow of solids. Bulletin no. 123; 53, 26, November 1964, Bull. Univ. Utah 53 (March) (1976) 209, https://doi.org/10.2172/5240257.
[25] R. Jin, X. Du, W. Chen, The use of metamodeling techniques for optimization under uncertainty, Struct. Multidiscip. Optim. 25 (2003) 99–116, https://doi.org/10.1007/s00158-002-0277-0.
[26] J.P.C. Kleijnen, Kriging metamodeling in simulation : a review, Eur. J. Oper. Res. 192 (2009) 707–716, https://doi.org/10.1016/j.ejor.2007.10.013.
[27] S. Koziel, L. Leifsson, Surrogate-Based Modeling and Optimization: Applications in Engineering, Springer, 2010.
[28] D.G. Krige, Journal of the chemical metallurgical & mining society of South Africa, J. Chem. Metall. Soc. South Min. Afr. 52 (6) (1951) 119–139http://journals.co.za/content/saimm/52/6/AJA0038223X_4792.
[29] H. Kureck, N. Govender, E. Siegmann, P. Boehling, C. Radeke, J.G. Khinast, Industrial scale simulations of tablet coating using GPU based DEM: a validation study, Chem. Eng. Sci. 202 (2019) 462–480, https://doi.org/10.1016/j.ces.2019.03.029.
[30] Y. Lang, A. Malacina, L.T. Biegler, S. Munteanu, J.I. Madsen, S.E. Zitney, W. Virginia, Reduced Order Model Based on Principal Component Analysis for Process Simulation and Optimization, 3, 2009 1695–1706.
[31] G.Y. Liu, W.J. Xu, N. Govender, D.N. Wilke, A cohesive fracture model for discrete element method based on polyhedral blocks, Powder Technol. 359 (2020) 190–204, https://doi.org/10.1016/j.powtec.2019.09.068.
[32] G.Y. Liu, W.J. Xu, Q.C. Sun, N. Govender, Study on the particle breakage of ballast based on a GPU accelerated discrete element method, Geosci. Front. 11 (2) (2020) 461–471, https://doi.org/10.1016/j.gsf.2019.06.006.
[33] S. Lommen, M. Mohajeri, G. Lodewijks, D. Schott, DEM particle upscaling for large-scale bulk handling equipment and material interaction, Powder Technol. 352 (2019) 273–282, https://doi.org/10.1016/j.powtec.2019.04.034.
[34] S. Lommen, D. Schott, G. Lodewijks, DEM speedup: stiffness effects on behavior of bulk material, Particuology 12 (1) (2014) 107–112, https://doi.org/10.1016/j.partic.2013.03.006.
[35] S.W. Lommen, D.L. Schott, G. Lodewijks, Multibody dynamics model of a scissors grab for co-simulation with discrete element method, FME Trans. 40 (4) (2012) 177–180.
[36] D.J. Lucia, P.S. Beran, W.A. Silva, Reduced-order modeling: new approaches for computational physics, Prog. Aerosp. Sci. 40 (1–2) (2004) 51–117, https://doi.org/10.1016/j.paerosci.2003.12.001.
[37] S. Luding, About contact force-laws for cohesive frictional materials in 2D and 3D, Proceedings Issue: Behavior of Granular Media, 2006.
[38] S. Luding, Cohesive, frictional powders : contact models for tension, Granul. Matter 10 (2008) 235–246, https://doi.org/10.1007/s10035-008-0099-x.
[39] J. Martin, T. Simpson, Use of adaptive metamodeling for design optimization, 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, September, 1–9., 2012https://doi.org/10.2514/6.2002-5631.
[40] K. McBride, K. Sundmacher, Overview of surrogate modeling in chemical process engineering, Chemie-Ingenieur-Technik 91 (3) (2019) 228–239, https://doi.org/10.1002/cite.201800091.
[41] M. Meckesheimer, A.J. Booker, T.B. Company, R.R. Barton, T.W. Simpson, Computationally inexpensive metamodel assessment strategies, AIAA J. 40 (10) (2002) https://doi.org/10.2514/2.1538.
[42] A.J. Morrison, I. Govender, A.N. Mainza, D.J. Parker, The shape and behaviour of a granular bed in a rotating drum using Eulerian flow fields obtained from PEPT, Chem. Eng. Sci. 152 (2016) 186–198, https://doi.org/10.1016/j.ces.2016.06.022.
[43] D.S. Nasato, C. Goniva, S. Pirker, C. Kloss, Coarse graining for large-scale DEM simulations of particle flow - an investigation on contact and cohesion models, Proc. Eng. 102 (2015) 1484–1490, https://doi.org/10.1016/j.proeng.2015.01.282.
[44] A. Quarteroni, G. Rozza, Reduced Order Methods for Modeling and Computational Reduction, 2014https://doi.org/10.1163/9789004275928_005.
[45] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, 7, MIT Press, 2006 Issue 5.
[46] A. Rogers, M. Ierapetritou, Challenges and opportunities in modeling pharmaceutical manufacturing processes, Comput. Chem. Eng. 81 (2015) 32–39, https://doi.org/10.1016/j.compchemeng.2015.03.018.
[47] D. Schulze, Powders and Bulk Solids: Behavior, Characterization, Storage and Flow, 2008.
[48] T.W. Simpson, Kriging models for global approximation in simulation-based multidisciplinary design optimization, AIAA J. 39 (12) (2001)https://doi.org/10.2514/2.1234.
[49] Søren N. Lophaven, J.S. Hans Bruun Nielsen, DACE - A Matlab Kriging Toolbox [Software]. Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2002http://www2.imm.dtu.dk/pubdb/pubs/1460-full.html.
[50] R.O. Uñac, A.M. Vidales, O.A. Benegas, I. Ippolito, Experimental study of discharge rate fl uctuations in a silo with different hopper geometries, Powder Technol. 225 (2012) 214–220, https://doi.org/10.1016/j.powtec.2012.04.013.
[51] M. Urquhart, E. Ljungskog, S. Sebben, Surrogate-based optimisation using adaptively scaled radial basis functions, Appl. Soft Comput. J. 88 (2020) 1–17, https://doi.org/10.1016/j.asoc.2019.106050.
[52] G.G. Wang, S. Shan, Review of metamodeling techniques in support of engineering design optimization, J. Mech. Des. 129 (4) (2007) 370, https://doi.org/10.1115/1.2429697.
[53] T. Weinhart, L. Orefice, M. Post, M.P. van Schrojenstein Lantman, I.F.C. Denissen, D.R. Tunuguntla, J.M.F. Tsang, H. Cheng, M.Y. Shaheen, H. Shi, P. Rapino, E. Grannonio, N. Losacco, J. Barbosa, L. Jing, J.E. Alvarez Naranjo, S. Roy, W.K. den Otter, A.R. Thornton, Fast, flexible particle simulations — an introduction to MercuryDPM, Comput. Phys. Commun. 249 (2020) 107129, https://doi.org/10.1016/j.cpc.2019.107129.
[54] J.S. Yoon, J.S. Park, C.O. Ahn, J.H. Choi, Cosimulation of MBD (multi body dynamics) and dem of many spheres using GPU technology, International Conference on Particle-Based Methods II - Fundamentals and Applications 2011, pp. 778–785.
[55] F. Zamponi, Mathematical physics: Packings close and loose, Nature 453 (7195) (2008) 606–607, https://doi.org/10.1038/453606a.
[56] Q.J. Zheng, M.H. Xu, K.W. Chu, R.H. Pan, A.B. Yu, A coupled FEM / DEM model for pipe conveyor systems : analysis of the contact forces on belt, Powder Technol. 314 (2017) 480–489, https://doi.org/10.1016/j.powtec.2016.09.070.
[57] R. Furukawa, Y. Shiosaka, K. Kadota, K. Takagaki, T. Noguchi, A. Shimosaka, Y. Shirakawa, Size-induced segregation during pharmaceutical particle die filling assessed by response surface methodology using discrete element method, Journal of Drug Delivery Science and Technology 35 (2016) 284–293, https://doi.org/10.1016/j.jddst.2016.08.004.
[58] P.M. Pardalos, A. Zhigljavsky, J. Žilinskas, Springer Optimization and Its Applications 107 Advances in Stochastic and Deterministic Global Optimization, 2016http://www.springer.com/series/7393.