Finite Elements in Analysis and Design 162 (2019) 51–66

Contents lists available at ScienceDirect

Finite Elements in Analysis and Design

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

Robust low cost meta-modeling optimization algorithm based on meta-heuristic and knowledge databases approach: Application to polymer extrusion die design

N. Lebaal

Institut Interdisciplinaire Carnot de Bourgogne, UMR 6303 CNRS, Université Bourgogne Franche Comté (UBFC), UTBM, Site de Sévenans, 90010, Belfort Cedex, France

ARTICLE INFO

Keywords:
Extrusion process
Metamodel optimization
PSO
Sampling
Nonlinear
Kriging interpolation

ABSTRACT

The new method presented in this paper falls into the category of sampling methods and model management in the optimization process of surrogate related methods. This method was introduced in order to reach the global optimum with a limited number of computer experiments. During these developments, the Particle Swarm Optimization (PSO) was used as a smart sampling tool to construct the metamodel. These methods with their stochastic nature can also overcome the problems of local minima.

In order to improve the efficiency and accuracy of the metamodel (Kriging), a knowledge database with smart sampling methods has been integrated into the optimization model management, to avoid unnecessary finite elements calculations and enrich the collection (sampling) in each optimization iteration. This method makes it possible to reduce the sampling size and at the same time increases the accuracy of the metamodel.

For validation of the developed method, different benchmark functions were chosen in terms of features and has successfully then minimized. Finally, a practical engineering optimization problem for polymer extrusion was implemented with suggested Kriging Swarm Optimization algorithm (KSO). In this procedure, the Finite Element Analysis (FEA) was combined for simulation procedures to resolve non-isothermal non-Newtonian flow. Polymer extrusion results were applied for gathering information from design space samples and Kriging.

1. Introduction

The most challenging aspect in the polymer extrusion industry is to eliminate or reduce die correction. Nevertheless, the complex die design is generally due to the nonlinear material behavior of the polymer. The exit velocity distribution through an extrusion die is a function of the shear-rate, temperature and the heat dissipation of the polymer melt. Numerical simulation of the extrusion process must take into account the nonlinear relationship between the viscosities, temperature and shear rate to predict accurately the velocity, pressure and temperature distributions within the die.

The extrusion die performance depends on the design of the manifold geometry and on the operating conditions adopted during extrusion [1–5]. Considerable gains can be obtained from the use of adequate numerical computational analysis for the prediction of polymer flows through extrusion dies [2–4,6]. However, the result of the optimal die is frequently obtained from very high numbers of numerical trial and error corrections, during which, the various numerical solutions are tested [7].

The high CPU time of such analysis limits the use of numerical computational analysis in engineering design optimization. Consequently, metamodel methods such as Design and Analysis of Computer Experiments (DACE) combined with Response Surface Models (RSM) are commonly used in engineering design optimization to minimize the CPU's running time for such analysis and simulations.

The aim of this method is to construct a simplified approximation of the numerical simulation and facilitate the design space exploration and optimization. Since the approximation model is considered as surrogate for the numerical simulation, or an approximation model of the numerical model, it is referred to as a metamodel, or surrogate model.

In material forming optimization problems [8–13], different approximation methods have been developed, in particular, Response Surface Method using second-order polynomial approximation [11–16]. Other researchers are focused on the management of this RSM such as including an auto adaptive research space strategy, to obtain the global optimum. This strategy has been successfully used in e.g., polymer extrusion [17] and clinching optimization problems [18,19].

Other approximation based methods, including Kriging have also been applied in generating the response surfaces for system

E-mail address: nadhir.lebaal@utbm.fr.

https://doi.org/10.1016/j.finel.2019.05.004
Received 10 October 2018; Received in revised form 9 April 2019; Accepted 21 May 2019

approximation [20–23]. Other types of models include Radial Basis Functions (RBF) [24–26], Multivariate Adaptive Regression Splines (MARS) [27], Neural networks [28–30] and Support Vector Regression have also been reported [31].

Among various models, Kriging and Moving Least Square (MLS) approximation using a second-order polynomials are the most intensively studied. The computation of the RSM using second-order polynomial of Moving Least Square approximation is fast and can efficiently model low-order problems. However, this method is not good for high nonlinear problems [32], such as polymer extrusion. Nevertheless, Kriging model is more accurate for nonlinear problems but difficult to obtain and to use.

In general, the approximation methods offer much improved computational efficiency as fewer simulation samplings are required compared to GA based evolutionary methods and such, and at the same time without using derivative results, which is the case for gradient based algorithms.

On the other hand, the robustness and effectiveness of approximation based methods are depended upon how approximation is achieved and how sampling methods is used, as reference to DACE. Other study is focused on the approximation model management in the optimization process, such as screening [16,32,33] or generalized Karhunen-Loève expansion (KLE) [34] to reduce the dimensionality of the design space. Another active branch of research in metamodeling is the reduction of the design space that can gradually reduce and zoom the design space around the best solution to improve the accuracy and the convergence as shown [35,36]. In these branches of research in reduction space, several methods are also suggested in the literature [32,33]. Other investigation is focused on what optimization algorithms to be adopted in searching the optimal approached solution.

The objective of the meta-model optimization is to reduce the number of samples (computation cost) in the design and analysis of computer experiments while ensuring a good accuracy of approximate model and an improvements of the optimal solution through each optimization iteration. According to Kriging models, it is observed that the modeling efficiency and accuracy are directly related to the design space. Two of the important parameters are the sampling size and sampling distribution in the design space.

Thus, design of experiment (DOE), or space filling, has become the determining factor for accuracy and efficiency of metamodeling process. The accuracy and efficiency of meta-model largely depend on sampling methods and according model management in the optimization process [37–39].

The new method presented in this work concerns the sampling methods and model management in the optimization process of surrogate methods. The method is presented to achieve the global optimum with a smaller number of samples (finite element analysis) and to orient the approximate model towards the global optimum during the optimization iterations.

In these developments, the Particle Swarm Optimization (PSO) is used as a sampling tool to construct the metamodel, replacing the traditional method of DACE. This will allow the Kriging-based meta-model to fly over the search space based on the individual and collective knowledge of each particle (or sample). These methods with their stochastic nature also can overcome the problems of local minima.

Concerning the model management in the optimization process, in our previous development [14,17], an auto adaptive sampling scheme with a reduction of the search space that can gradually seek to reduce the search space around the optimum has been adapted and integrated as DACE using traditional Composite Design of Experiment (CDE). On the contrary of this space reduction schemes, in this development, all the design space is taken into account in each optimization iteration. This makes it possible for each sample to find other possible global optimum.

As very little is known a priori about the shape of the response function, one could argue that sample points should be chosen to fill all the design space. For this, a Central Composite Design (CCD) is used as initial sampling in order to fill all the design space with a known sampling number.

In the optimization process, a knowledge database, based on the history of movement of each particle, has also been integrated into the model management to avoid unnecessary finite element calculations and thus enrich the collection (sampling) in each optimization iteration. This method makes it possible to reduce the sampling size and at the same time increase the accuracy of the metamodel.

2. Optimization procedure

When we use FEA, the evaluation of the objective and constraint functions become implicitly related with the optimization parameters and need expensive numerical analysis. To obtain the optimum parameters at low cost yet good accuracy, the metamodeling approach using Kriging interpolation is adopted and coupled with a smart sampling and model management strategy.

The Kriging interpolation [22,33,40,41], is applied in this work to represent the metamodel in explicit form according to the optimization variables.

The approximate functions can be expressed as follow:

$$\tilde{J}(x) = P^T(x)a + Z(x) \tag{1}$$

where x is the design variables, $a = [\,a_1, \quad \ldots, \quad a_m\,]^T$ is the unknown parameters vector, $\tilde{J}(x)$ is the approximate function (objective or constraint function), and $Z(x)$ is the random fluctuation assumed to be a realization of a stochastic process with mean zero and spatial correlation function given by:

$$cov\left[Z(x_i),\ Z(x_j)\right] = \sigma^2 R(x_i, x_j) \tag{2}$$

where σ^2 is the process variance, and R is the correlation matrix, which is composed of the correlation function evaluated at each possible combination of simple of the design and analysis of computer experiments.

The value of the m basis function $p(x)$ evaluated at each sample are included in the vector P with $P(x) = [\,p_1(x), \quad \ldots, \quad p_m(x)\,]^T$.

The implicit output responses vector from the function is given as:

$$F(x) = [\,f_1(x), \quad \ldots, \quad f_n(x)\,]^T \tag{3}$$

The unknown coefficient a can be estimated using the fallowing equation:

$$a = \left(P^T R^{-1} P\right)^{-1} P^T R^{-1} F \tag{4}$$

The second part in Eq. (1) is used to model the deviation from of the regression to interpolate response data from the function and is given as:

$$Z(x) = r^T(x)\beta \tag{5}$$

where $r^T(x)$ is the correlation vector between an untried x and the sampled data points, given as:

$$r(x) = [\,R(x_1, x_1), \quad \ldots, \quad R(x_1, x_n)\,] \tag{6}$$

The parameters β are defined as fallow:

$$\beta = R^{-1}(F - Pa) \tag{7}$$

The accuracy and efficacy of metamodel largely depend on sampling methods and model management in the optimization process. In this development, Kriging Swarm Optimization (KSO) formulation is developed in order to reduce the sampling size and at the same time increase the accuracy of the metamodel.

2.1. Kriging Swarm Optimization (KSO) formulation

Design and analysis of computer experiments (DACE) have become the key to metamodeling process. In these developments, the PSO

algorithm [42–46] is used as a smart sampling tool to build the meta-model. For this, we replaced the traditional method of DACE by PSO algorithm. This will enable the meta-model based on Kriging to fly over the search space based on the individual and collective knowledge of each particle (or sample). Also, these methods with their stochastic nature overcome the problems of local minima and seek to reduce the search space around the global optimum. Knowledge databases, based on the history of the trajectory of each particle, has been integrated into the optimization algorithm to share the individual knowledge of each particle, and avoid unnecessary FEA calculations. In addition, this knowledge database makes possible to enrich the collection (sampling) on every optimization iteration.

The details of the proposed method are illustrated in the flowchart Fig. 1 and presented by the following steps:

Step 1 and 2: Definition of the search space, and initialization of the population array of particles and velocities in the search space. The initial sampling is distributed throughout the search space using a composite experimental design $X^{i,j\ (DOE)}$, $i = 1, 2, \ldots n\ j = 1, 2, \ldots, D$.where n is the size of the swarm, D is the dimension number.

Step 3: Finite elements evaluation of the problem for each sample (i) and implicit evaluation of the desired optimization fitness functions F_k^i at the iteration k.

Step 4: Creation & update the knowledge Swarm DataBases (SDB), containing information for each particle (fitness F_k^i, position fitness $x_k^{i,j\ (PSO)}$ and velocity $v_k^{i,j\ (PSO)}$ for all optimization iterations), for this, each particle remembers his trajectory and velocity. The SDB is then used to enrich the collection (memory of previous fitness and positions) for the construction of accurate metamodel using Kriging interpolation.

At the first iteration (k = 1):

The particle position stored in the databases $X_k^{i,j\ (SDB)}$, represent the particle position defined by the composite design of experiment.

$$\begin{cases} X_k^{i,j\ (SDB)} = X^{i,j\ (DOE)} \\ v_k^{i,j\ (SDB)} = 0 \\ F_k^{i\ (SDB)} = F_k^i \end{cases} \tag{8}$$

where, $v_k^{i,j}$ is the jth dimension velocity of particle i at iteration k.else:

The updated particle position, velocity and stiffness stored in the databases $X_k^{i,j\ (SDB)}$, represent a new set that contains all of elements that are in at last one of the two set, i.e. the previous (*k-1*) stored parameters ($X_{k-1}^{I_{k-1},j\ (SDB)}$, $v_{k-1}^{I_{k-1},j\ (SDB)}$,$F_{k-1}^{I_{k-1}\ (SDB)}$) and respectively the i particle position ($x_k^{i,j\ (PSO)}$), velocity ($v_k^{i,j\ (PSO)}$) and fitness obtained by PSO algorithm at iteration k.

$$\begin{cases} X_k^{I_k,j\ (SDB)} = X_{k-1}^{I_{k-1},j\ (SDB)} \cup x_k^{i,j\ (PSO)} \\ v_k^{I_k,j\ (SDB)} = v_{k-1}^{I_{k-1},j\ (SDB)} \cup v_k^{i,j\ (PSO)} \\ F_k^{I_k\ (SDB)} = F_{k-1}^{I_{k-1}\ (SDB)} \cup F_k^i \end{cases} \tag{9}$$

where $I_k = 1, 2, \ldots n_k$ are the new sampling size at the iteration k, $x_k^{i,j\ (PSO)}$ and $v_k^{i,j\ (PSO)}$ are respectively the i particle position and velocity obtained by PSO algorithm at iteration k.

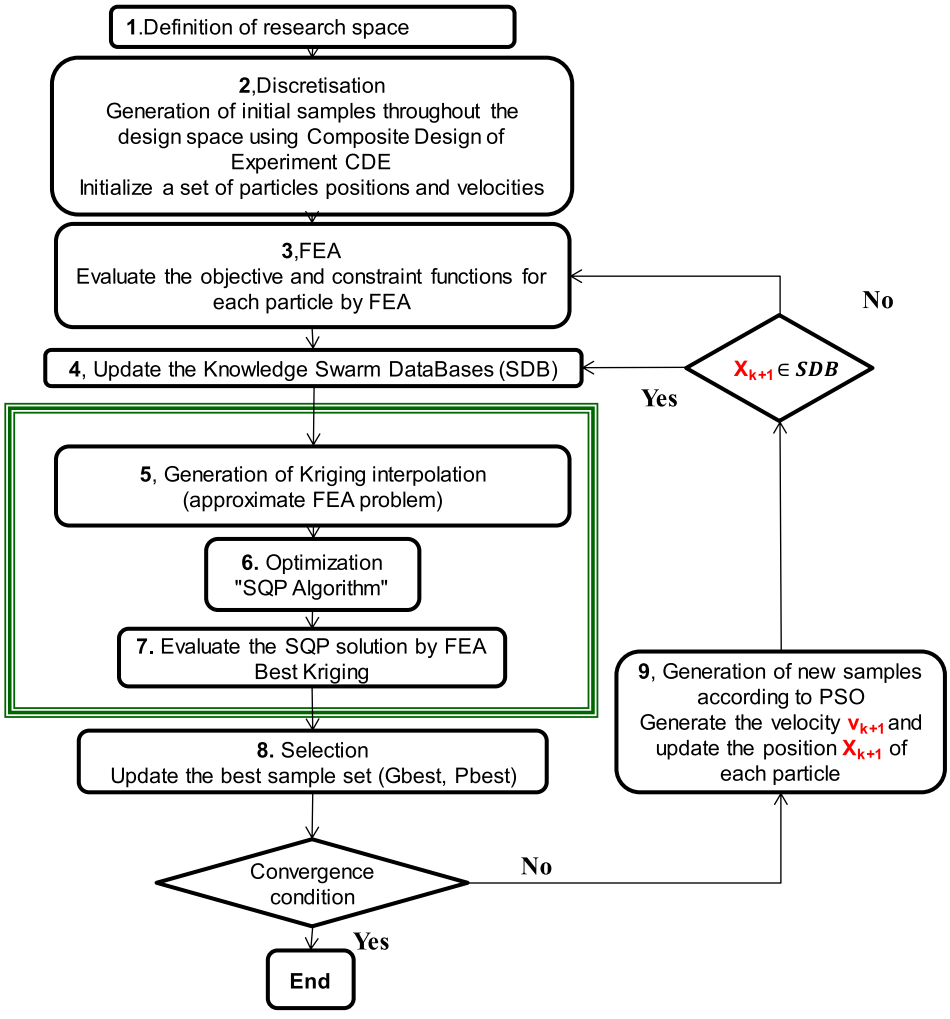

Fig. 1. Flow-chart of KSO algorithm.

Step 5: Construction of the metamodel using Kriging. The sampling used to construct this model is enriched by all samples stored in the swarm databases $X_k^{I_k j\ (SDB)}$. This can considerably improve the accuracy of the metamodel.

Step 6: Optimization of this explicit problem using Sequential Quadratic Programming (SQP) algorithm. To avoid falling into a local optimum, an automatic procedure is used which allows resolving the optimization problem using SQP algorithms, starting from each sample, then the best approximate solution among those obtained by the various optimizations is considered in the optimization procedure.

Step 7: Implicit evaluation (Finite elements evaluation) of the optimum at iteration k.

For more details of step 5, 6 and 7, the readers can consult the paper of Lebaal et al. [47].

Step 8: Select and updating of best samples located in the search space. The quality of the position of each particle (sample) is determined by the value of the fitness in this point F_k^i. Also, the optimum value implicitly evaluated from the optimization results using the SQP and Kriging is included in the swarm. The swarm size becomes (i = i+1). Each particle remembers the best position where it has already passed, which is denoted $P^{best^i}_k$. Compare particle's fitness evaluation with its $F^{best^i}_{k-1}$. If its current value F_k^i is better than $F^{best^i}_{k-1}$, then set equal to the current value of the objective function $F^{best^i}_k = F_k^i$, and $P^{best^i}_k$ equal to the current design value (position) X_k^{ij} in the design space. Else, $F^{best^i}_k = F^{best^i}_{k-1}$ *and* $P^{best^i}_k = P^{best^i}_{k-1}$. Then the best position of the neighborhood, i.e. the best position reached by the particles of the rated swarm G^{best}_k is equal to the best of all swarm with including Kriging-SQP solution of step 5,6 and 7.

Step 9: Generating new samples using the PSO algorithm. At each iteration k, each particle (sampling point) moves, by linearly combine the three components illustrated in Fig. 2.

These are calculated as follows :

$$v_{k+1}^{i,j} = w v_k^{i,j} + c_1 r_{1k}^{i,j}\left(P^{best^{i,j}}_k - x_k^{i,j\ (PSO)}\right) + c_2 r_{2k}^{i,j}\left(G^{best^j}_k - x_k^{i,j\ (PSO)}\right) \tag{10}$$

$$x_{k+1}^{i,j\ (PSO)} = x_k^{i,j\ (PSO)} + \chi\, v_{k+1}^{i,j},\quad i = 1,2,\ldots.n\ \ j = 1,2,\ldots,D \tag{11}$$

where χ is a constriction factor which is used to control and constrict velocities; w is an inertia weight factor, c_1 is a cognition weight factor, c_2 is a social weight factor, $r_{1k}^{i,j}$ and $r_{2k}^{i,j}$ are two random numbers varies between 0 and 1. Vector $P^{best^{i,j}}_k$ is the jth dimension of the own best position of particle i and $G^{best^j}_k$is the jth dimension of the best sample in the swarm.

Following weighting inertia function is used in above eq.

$$w = w_{max} - \frac{w_{max} - w_{min}}{k_{max}} k \tag{12}$$

where.

w_{max} = initial weight
w_{min} = final weight
k_{max} = maximum iteration number
k = current iteration number
with $w_{max} = 1$; $w_{min} = 0.8$; $C_1 = 0.5$ and $C_2 = 2$.

Step 10: Using the knowledge database: If the location of the new sample (particle) is very close to a particle stored in the database, the value of the objective and constraint(s) functions of current particles are replaced by the values of the corresponding particle stored in the database. The positions are also restored. However, the particles velocities of the considered sample are not restored; each particle keeps its velocity in order to follow its trajectory.

If $\left\| x_{k+1}^{i\ (PSO)} - X_k^{I_k j\ (SDB)} \right\| < \varepsilon \qquad \forall\, i \in 1,\ldots,n\ \&\ \forall\, I_k \in 1,\ldots,n_k$

$$\begin{cases} x_{k+1}^{i\ (PSO)} = X_k^{I_k j\ (SDB)} \\ F_{k+1}^{i} = F_k^{I_k\ (SDB)} \\ v_{k+1}^{i,j} = v_{k+1}^{i,j} \end{cases} \tag{13}$$

Then go back to step 4.

Otherwise, implicitly evaluate the objective functions and constraint(s) for each new particle's position using finite element calculations (step 3).

3. Application of KSO to benchmark functions

Before testing and solving an optimization problem; it is important to know what function aspects can make the optimization process become difficult. For this, and to compare and validate the performance of the proposed optimization algorithm, three benchmark functions with diverse properties are selected. These functions are chosen in terms of features like modality, basins, and dimensionality. All test functions are inseparable to increase the difficulty of the optimization.

To explore the robustness and efficacy of the KSO algorithm, an example of the Rastrigin's function was considered for the first illustration (section 3.1). To illustrate more clearly the workings of this algorithm, the Rastrigin's function was considered in two-dimensions with a restrictive search space in order to visualize the convergence of the algorithm. In section 3.2 other examples are given to explore other parameters like modality for larger search space, basins, and dimensionality.

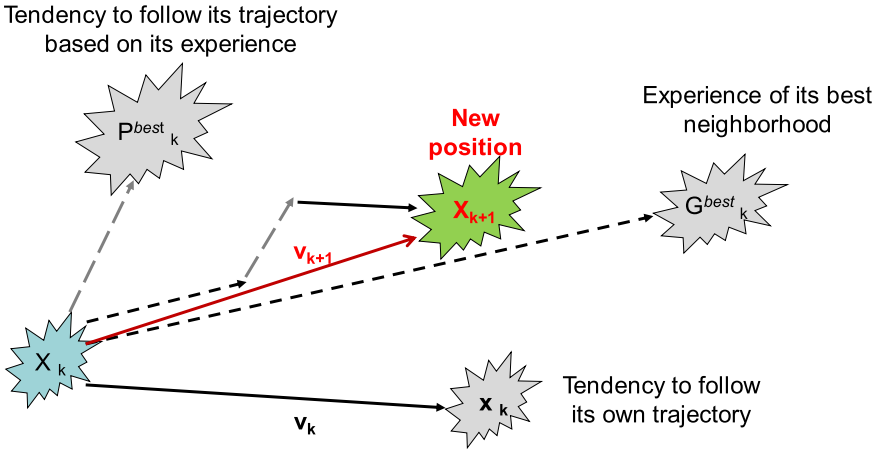

Fig. 2. The displacement strategy of each sample (PSO).

3.1. Working illustration of KSO for 2D Rastrigin's function

This highly non-convex function with many local minima would present considerable difficulty with gradient-based methods. The global minimum of the two-dimensional (d = 2) Rastringin's function evaluated on the bounded space (x,y) ∈ [-1, 2], was located at (x = y = 0). Notice that the lower and upper boundaries of the optimization are not the same. The function was not symmetric around the solution parameters to avoid that at the initialization of the KSO, the central point of the DOE is localized in the global optimum, forcing the algorithms to search over multiple orders of magnitude and effectively increase the difficulty of the optimization problem.

The function to be minimized was given by:

$$f = 20 + x^2 + y^2 - 10[\cos(2\pi x) + 10\cos(2\pi y)] \quad (14)$$

For this example, two solution errors were defined to cut-of the optimization solution, and were defined as $\mathscr{R} = \left\|X^{*j} - X^{sol,j}\right\|$, and. $\mathscr{F} = (f^* - f^{sol})^2$

wher j = 1,2 for two-dimensional problem.

The known solutions of the optimization problem $X^{sol,j}$ and f^{sol} are respectively the parameters value and the objective function value of the global optimum.

The minimum value of $\mathscr{F}$ and $\mathscr{R}$ are 0. X^j is the vector of the two optimization parameters (x,y).

Fig. 3 shows a graphical representation of Eq. (7).

For this example, a maximum of 10 iterations was used for the optimization algorithms. The initial populations are imposed as a CDE with 9 individuals (trial solutions).

Fig. 3a shows the Rastrigins function with the initial sampling (CDE) and the correspondent metamodel based on kriging. The 9 distributed samples were generated now as shown in Fig. 4a. For the first optimization iteration, it is observed that the Best Swarm solution is not the same as the Kriging solution. The Global best (KSO) converges to the Kriging solution. In the second iteration (Fig. 3b), each particle (sampling) determines its trajectory by combining certain aspects of the history of its own trajectory and best positions with those of the swarm, using some random perturbations. Additionally, using the neighborhood relation, each particle communicates with some other particles and is affected by the best point found by any member of its topological neighborhood, and also by the best Kriging position. This best solution seeks to reduce the search space of particles around the global optimum (KSO) and the PSO algorithm updates its best Swarm position to refresh the metamodel. In this iteration, it is observed that the swarm size decreases (Step 10) as illustrated in Fig. 4b. Furthermore, the metamodel is more accurate around the global optimum due to the position of each

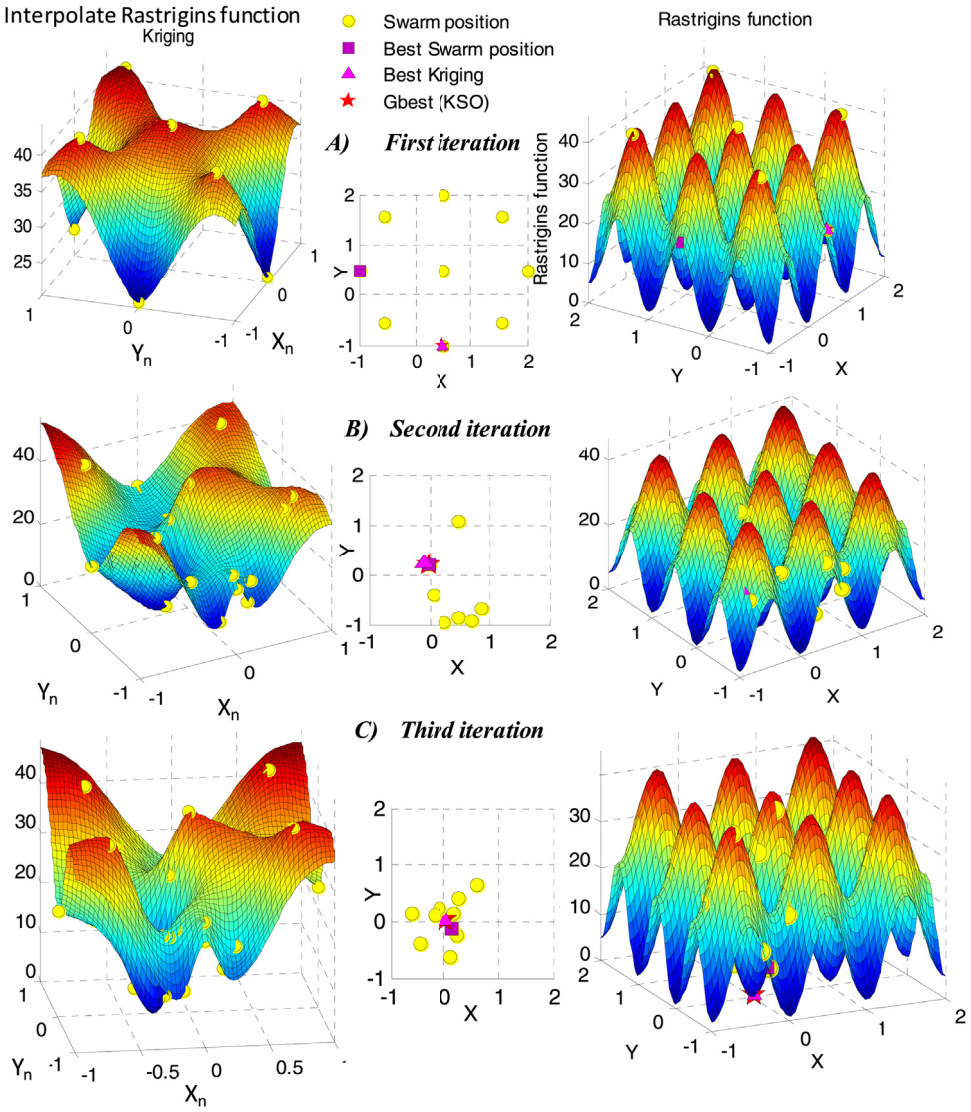

Fig. 3. An illustration of metamodels according to Kriging method based on PSO and SDB for each iteration.

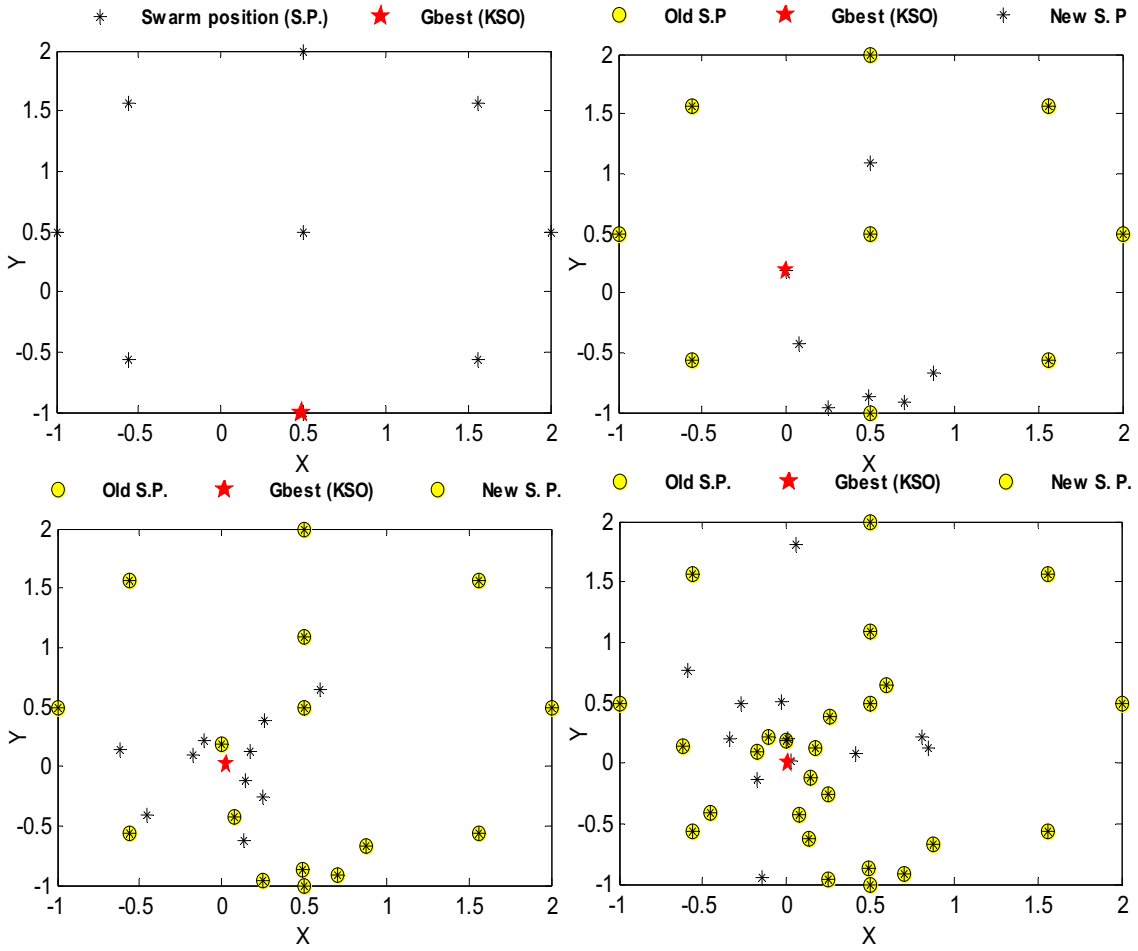

Fig. 4. An illustration of samples distribution generated by PSO and SDB for each iteration. The convergence story of the optimization run is illustrated in Fig. 5, and the summary of the optimization of the Rastrigin function is listed in Table 1.

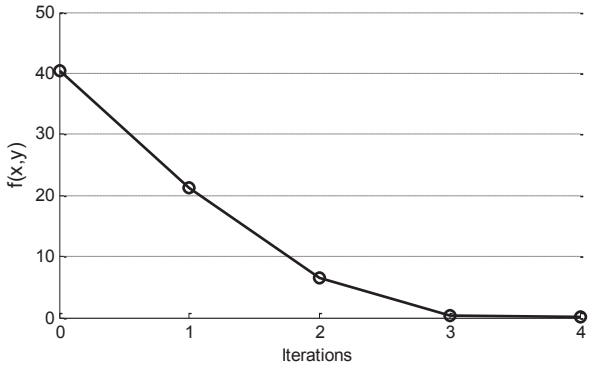

Fig. 5. Convergence history of the optimization of Rastrigin function.

particle. Additionally, an enrichment of the collection (Step 4) by using the memory of previous fitness and positions (Old Swarm Position "Old S.P″) through means of Knowledge Swarm DataBases (SDB) as illustrated in Fig. 4b, c, and d. This will improve the accuracy of the metamodel in all design spaces as illustrated in Fig. 3b and c. At the same time, due to the stochastic characteristic of the sampling method, the optimization algorithm can avoid the local optimum. For this second iteration, it is observed that the optimal solution of the sampling methods (PSO) and Kriging are close.

At the third iterations (Fig. 3c), the optimization algorithm converges to the optimal solution. The metamodel is more and more accurate around the global optimum due to the sampling methods and the use of SDB (Figs. 3 and 4). The best results (KSO) are different from the best swarm solution (PSO). However, the Best Kriging solution represents the better results.

It is observed that the optimization algorithm avoids the local minima and converges well with a limited number of functions evaluations. However, this is still a simple function of only two variables with a restrictive search space. It is interesting to study the behavior of the optimization algorithm for other terms of features, in order to prove the effectiveness of the proposed method. In the next section, the KSO algorithm is applied to more complex functions involving other parameters like modality for larger search space, basins, and dimensionality to explore its effectiveness.

3.2. *Advanced demonstration and validation*

There have been many test or benchmark functions reported in the literature; however, there is no standard set of benchmark functions. Preferably, test functions should have diverse properties so that they can be truly useful to test new algorithms. In this section, different benchmark functions are chosen in terms of features like modality for larger search space, dimensionality, the presence of a large hole and basins. To explore the effectiveness of the proposed algorithm, a standard PSO algorithm is also used and compared with the KSO algorithm.

There are many measures of merit for optimization algorithms. These measures can be defined as the number of function evaluation and accuracy.

To take into account the impact of the random number seeds, each

algorithm was executed 20 times for each problem ($N\_opt = 20$). The value of the mean μ and standard deviation σ value of each merit measure is used as a comparison basis.

For these examples, to cut-off the optimization solution, the following parameters are defined: maximum iteration of 10, $\left\|X^{*j} - X^{solj}\right\| \leq 1e^{-3}$ or. $\left\|f^{*} - f^{sol}\right\| \leq 1e^{-3}$

The focus of the analysis is to use, as a comparison basis of accuracy, the objective function values found by the algorithms and the locations of the solutions in the domain space. This is because locations must distinguish between multiple optima, whereas using only objective function values, the solution can include local optima.

The mean and standard deviation value of the objective function values and the Euclidean norm between the optimal parameter value and the locations of the solutions are defined as follows:

$$\mu_{\mathscr{R}} = \frac{1}{N\_opt}\sum_{i=1}^{N\_opt}\left\|X^{*j} - X^{solj}\right\|,\ \sigma_{\mathscr{R}} = \frac{1}{N\_opt}\sum_{i=1}^{N\_opt}\left(\left\|X_i^{*j} - X^{solj}\right\| - \mu_{\mathscr{R}}\right)^2 \tag{15}$$

$$\mu_f = \frac{1}{N\_opt}\sum_{i=1}^{N\_opt} f_i^{*},\ \sigma_f = \frac{1}{N\_opt}\sum_{i=1}^{N\_opt}\left(f_i^{*} - \mu_f\right)^2 \tag{16}$$

Instead of CPU time the maximum numbers of function evaluations (f_{eval}) are used to measure the effectiveness of the optimization algorithm. The mean and standard deviation values are defined as follows:

$$\mu_{f_{eval}} = \frac{1}{N\_opt}\sum_{i=1}^{N\_opt} f_{eval_i}^{*},\ \sigma_{f_{eval}} = \frac{1}{N\_opt}\sum_{i=1}^{N\_opt}\left(f_{eval_i}^{*} - \mu_{f_{eval}}\right)^2 \tag{17}$$

a) Modality for larger search space

The modality represents the number of peaks in the function, which is referred to the multiple local optima. If the optimization algorithms encounter these peaks, there is a tendency to be trapped during a search process and can direct the search away from the true optimal solutions. In order to study the capability of the KSO algorithm to not be trapped in one of such peaks, which will have a negative impact on the search process, two-test function with multimodality are selected. The Ackley and Rastregin function.

The Ackley function is characterized by a nearly flat outer region with many local minima and a long single-funnel at the coordinate $x_j = 0$ as it is illustrated in its two-dimensional form, as shown in Fig. 6. This function poses a risk for optimization algorithms, to be trapped in one of its many local minima especially if the local minimum is detected in the nearly flat region. The large hole is also difficult to detect for the approximation based algorithm.

The function to be minimized for 'd' dimension was defined as follow.

$$f_{\text{Ackley}}(x) = -20e^{\left(-0.2\sqrt{\frac{1}{d}\sum_{j=1}^{d}x_j^2}\right)} - e^{\left(\frac{1}{d}\sum_{j=1}^{d}\cos(2\pi\, x_j)\right)} + 20 + e^{1} \tag{18}$$

Here, d represents the number of dimensions, and this function is

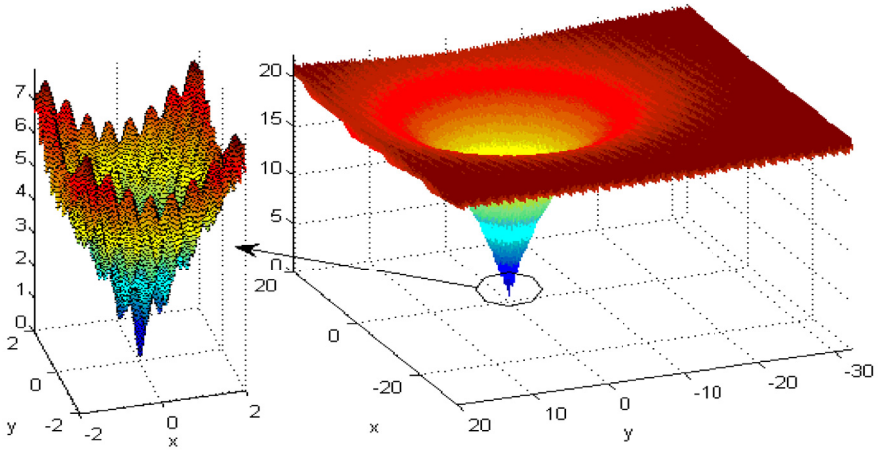

Fig. 6. Two-dimensional illustration of Ackley function.

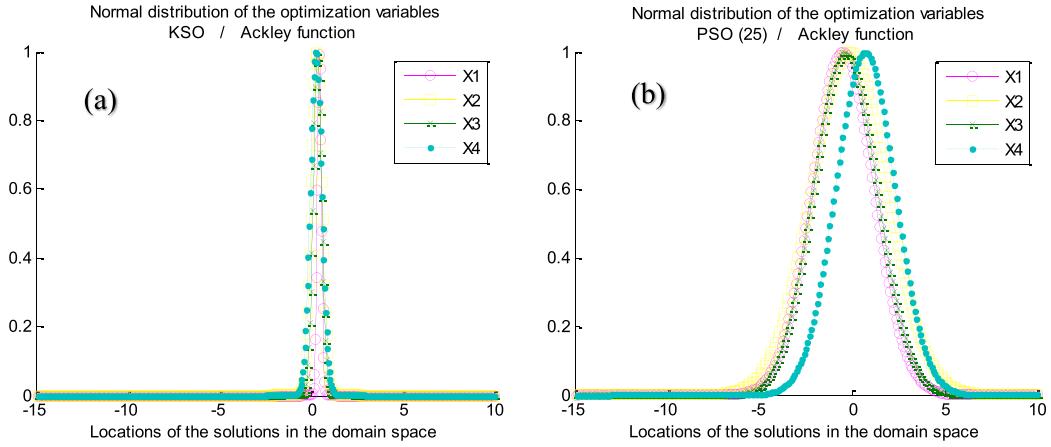

Fig. 7. Normal distribution plot based on the best solution values of the optimization variables for both KSO (a) and PSO (b) algorithm for 4d Ackley function.

evaluated, in order to increase the optimization difficulty, on a large asymmetric search space defined by the hypercube $x_j \in [-32, 20]$, for all $j = 1, \ldots, d$.

The normal distribution of the optimal values of the optimization variables is plotted in Fig. 7, to analyze the accuracy of the optimization algorithms and to localize the solutions in the domain space in each replication. The optimal solution is located at $x_j = 0$ for all $j = 1, \ldots, d$. It is noted that the mean value of the normal distribution of illustrated optimal variables plots is closer to the optimal solution of zero, and the standard deviations are not spaced for KSO Algorithm. This indicates that the KSO Algorithm often finds the best and the same localization at the replication runs.

For PSO algorithm, the standard deviation is evenly spaced. This involves that the solutions value at the replication runs are frequently different. However, the solutions are not far from the optimal localization.

Fig. 8 shows the performance profiles of the algorithms when the maximum number of iteration is 10. These figures also illustrate the effect of the swarm size on the performance of the PSO algorithm, by examining two different levels of population size. The function evaluations (eval) are also given.

Furthermore, we construct normal distribution plots, as shown in Fig. 8, to capture the efficiency of the algorithms in terms of mean and standard deviation values of the variation of the optimal solution found in each replication for both objective functions (Fig. 8a) and the Euclidean norm value (Fig. 8b). It is noted that the standard deviation of the normal distribution of illustrated objective function plots is intentionally not evenly spaced for KSO Algorithm. This implies that the solutions value at the replication runs are often the same. The mean value is closer to the optimal solution of zero, indicates that the KSO Algorithm often finds a lot of improving function values at the replication runs.

For PSO algorithm, it is observed that the standard deviation is evenly spaced. This involves that the solutions value at the replication runs are often different. However, when the population size increases the mean value approaches the optimal solution, but the number of maximal iteration is not sufficient to converge. The same results are observed for the Euclidean norm which represents the position accuracy of the optimization variables.

The second highly multimodal test function is Rastrigin function (Fig. 9), it has several local minima with regularity distribution. In d dimension this function is defined as follows:

$$f_{\text{Rastrigin}}(x) = 10d + \sum_{j=1}^{d} \left[x_j^2 - 10\cos(2\pi x_j) \right] \tag{19}$$

In a similar plot as the Ackley problem, Fig. 10 illustrates the performance profiles of both algorithms for 4 D Rastrigin function. The normal distribution plots, as shown in Fig. 10, illustrate an extremely tight standard deviation for the optimal solutions of both objective function and Euclidean norm for KSO Algorithm, which denote that the

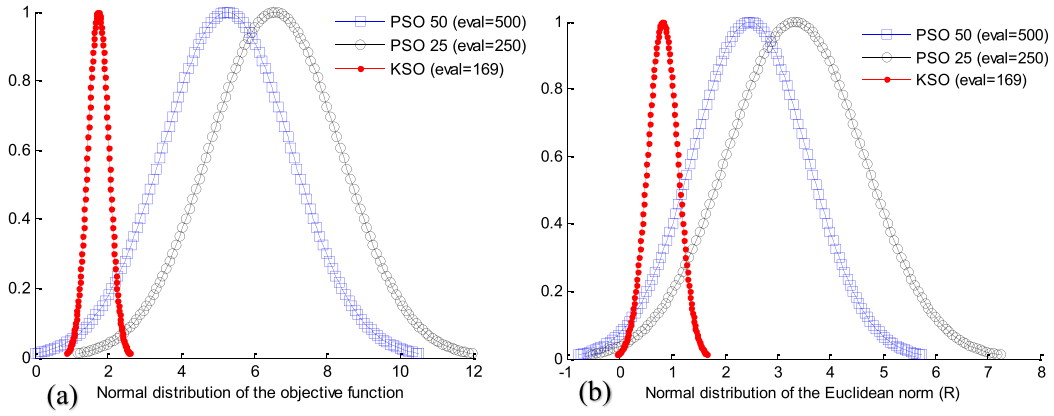

Fig. 8. Normal distribution plot based on the best objective function (a) and Euclidean norm (b) values for 4d Ackley function.

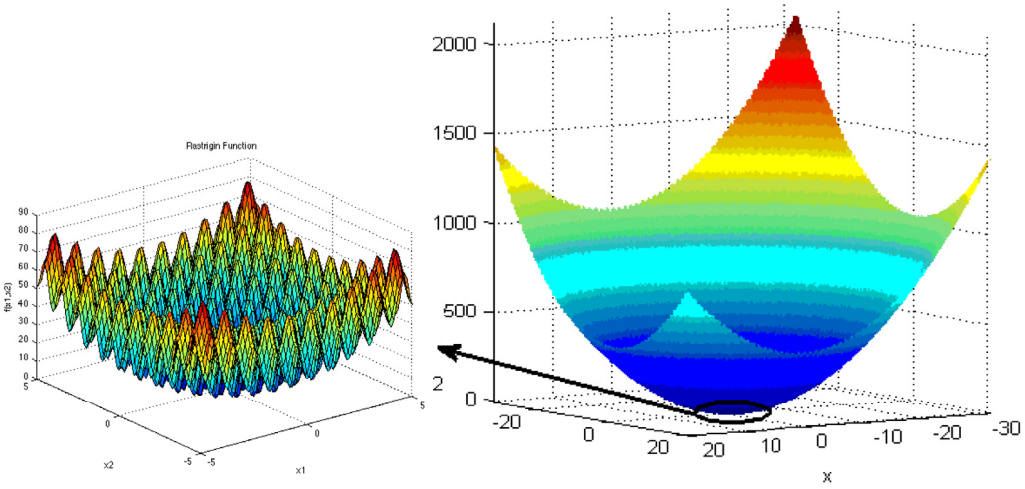

Fig. 9. Two-dimensional illustration of Rastrigin function.

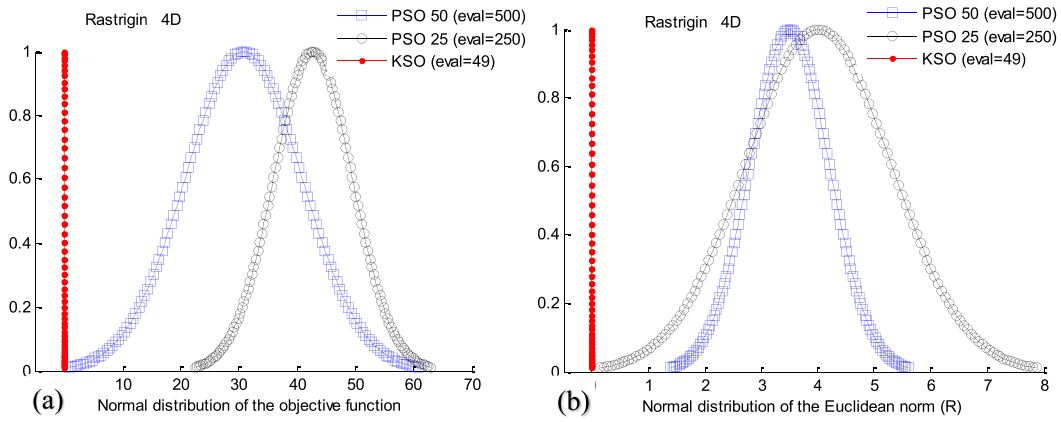

Fig. 10. Normal distribution plot based on the best objective function (f) and Euclidean norm (R) values for 4d Rastrigin function.

Table 1
Summary of the results.

Iterations	x	y	$f(x,y)$	$\mathscr{T}(x,y)$	$\mathscr{R}(x,y)$	$F_{obj} = f/f_0$
0	0,5	0,5	40,5	1640,250	0,7071	1
1	0,4943	−1	21,238	451,053	1,1155	0,5244
2	−0,006	0,1927	6,5201	42,512	0,1928	0,1610
3	0,0314	0,0167	0,2499	0,062	0,0356	0,0062
4	0,0111	0,0078	0,0365	0,001	0,0136	0,0009

solutions value at the replication runs are very close.

The mean value is nearer to the global optimum (solution of zero), which indicates that the KSO Algorithm found a lot of improving function values at the replication runs with only 49 function evaluations.

For PSO algorithm, the solutions value at the replication runs are often different, and the mean value is far from the optimal value. When the population size increases, the mean value approaches the optimal solution but the number of iteration is not sufficient to converge. Nevertheless, the number of function evaluation increases considerably (500 function evaluations).

In order to illustrate the effect of the maximum number of iteration, the performance of the PSO algorithms is given in Table 2 by examining three different levels of the maximum number of iterations. This table shows the mean and standard deviation values of many measures of merit for 2D Rastrigin function optimization. It can be noted that the PSO algorithm highly depends on the maximum number of iteration allowed. The mean of objective function value decreases as the number of iteration increases. Furthermore, the number of function evaluation increases proportionally in function of the iterations. This effect on algorithm performance is not surprising; however, for the KSO algorithm is it not necessary to increase the number of iteration because the solution is reached before. With a reduced time cost varying between 81% and 96,8% compared respectively to the examples of low and high evaluations.

KSO algorithm illustrates promising results by dominating the PSO algorithm throughout the accuracy and the number of functions evaluations. In addition, it demonstrated that it could avoid the risk to be trapped in one of the many local minima presented in both test functions and can detect the global minimum even if it is located in a large hole such as for Ackley problem.

Now it is interesting to study the performance of the proposed optimization algorithm for other terms of features, such as the presence of Basin.

b) Basins

The Basins are the flat regions to which the optimization algorithms can be easily attracted. Once in these regions, the search process become severely troubled. This is due to the lack of information to direct the search process towards the minimum. To study the capability of the KSO algorithm to avoid such problem, a Rosenbrock function as illustrated in its 2D form in Fig. 11 is selected. This multimodal function is defined in 'd' dimension as follows:

$$f_{\text{Rosenbrock}}(x) = \sum_{j=1}^{d-1}\left[100\left(x_j^2 - x_{j+1}\right)^2 + \left(x_j - 1\right)^2\right] \tag{20}$$

And $x_j \in [-2\ 2]$, for all $j = 1, \ldots, d$.

Two examples are selected for illustration. The normal distribution plots, for both 2d and 6d Rosenbrock optimization problem, are given in Fig. 12. Fig. 12a, illustrates the mean value of zero which represents the optimal solution of the objective function with an extremely tight standard deviation for the optimal solutions for KSO Algorithm with a very small number of function evaluation (55 evaluations) compared to PSO algorithm (90–180). A similar plot is illustrated in Fig. 12b for 6d Rosenbrock problem. It can be observed that the KSO algorithms settle at about the same function value, each with an extremely tight value around the mean at the last iteration. A similar result is obtained with the PSO algorithm using high population size. Furthermore, the number of function evaluation increases (770 evaluation).

In the same way as for highly multimodal problem, KSO algorithm illustrates promising results throughout the accuracy, and the number of

Table 2
Summary of the mean and standard deviation values of many measures of merit for 2D Rastrigin function optimization.

	PSO population 9		PSO population 18		PSO population 18		PSO population 18		KSO	
Iteration max	10		10		20		30		10	
2D	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
Objective	12,38	7,62	6,32	4,42	3,17	1,97	1,53	1,14	3,9E-04	1,2E-03
Evaluations	90	0	180	0	360	0	540	0	17	0
Norme (X)	2,58	1,29	1,31	0,93	1,18	0,73	0,67	0,61	4,9E-04	1,4E-03

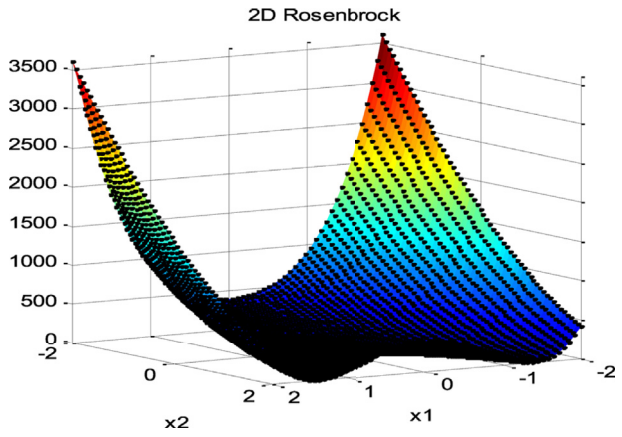

Fig. 11. Two-dimensional illustration of Rosenbrock function.

functions evaluations compared to PSO to obtain the best result for many local minima problem with the presence of flat region such as the basin. Especially if the local minimum is detected in the nearly flat region as in the case of Rosenbrock problem.

c) Dimensionality

Generally, the difficulty of an optimization problem increases with its dimensionality. As the number of parameters or dimension increases, the search space also increases exponentially. For highly nonlinear problems, this dimensionality may be a significant barrier for almost all optimization algorithms. To demonstrate the capability of the KSO algorithm to converge to the best solution for different dimensionality and not be trapped in one of the multi-local minima, the Ackley and Rastreging functions are used as test functions with different dimensions varying from 2 to 10. The maximum number of iteration of 10 is used for the comparison results. For PSO algorithm two (high and low) populations size for each dimension are used.

Fig. 13 represents the results of optimal values of the objective function by the means of both the average and the standard deviation of all the replications at the last optimization iteration with different dimensions varying from 2 to 10. The results of the KSO algorithm are presented and compared with the PSO algorithm for both high and low population size. Each color bar indicates mean values and the vertical bar represents the standard deviation. A large standard deviation indicates that the solutions can spread far from the mean and a small standard deviation indicates that they are clustered closely around the mean. It is observed that the KSO algorithm converge to the global solution for all dimensions and the values of the objective function are less than the cut-off criteria of 10^{-3} for all replications. For PSO algorithm, it is noticed that the mean value increases with the dimensionality of the optimization problem. These higher values indicate that the convergence becomes very difficult when the dimensionality increases. The standard deviation values are also relatively large and decrease when the dimensionality decrease. For the effect of population size, in the PSO algorithm, it is observed a better solution for all dimensions when the population size increase. However, the number of function evaluation increases to a considerable level such as illustrated in Fig. 14. It is also observed the

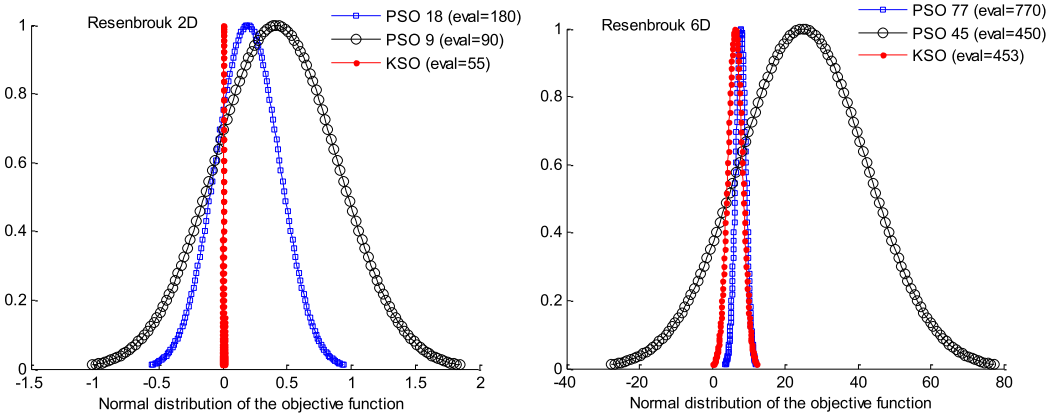

Fig. 12. Normal distribution plot of objective function for 2d and 6d Rosenbrock problem.

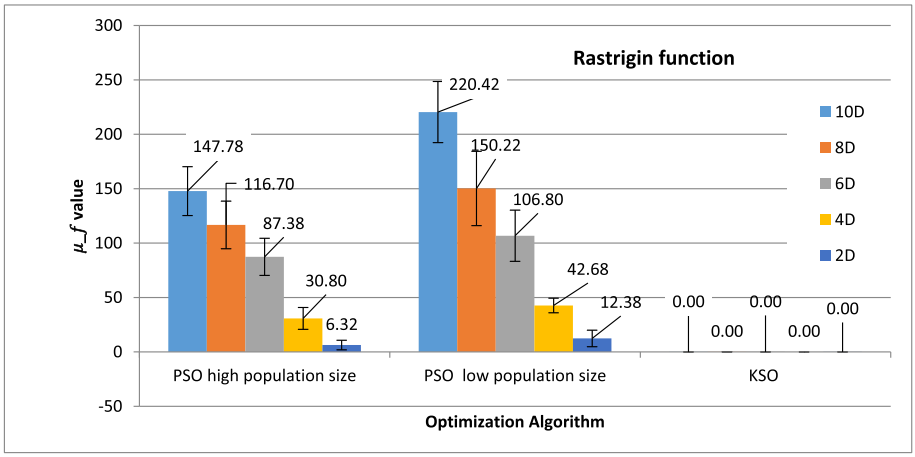

Fig. 13. Mean and quartile plot of objective function values for d dimension Rastrigin problem.

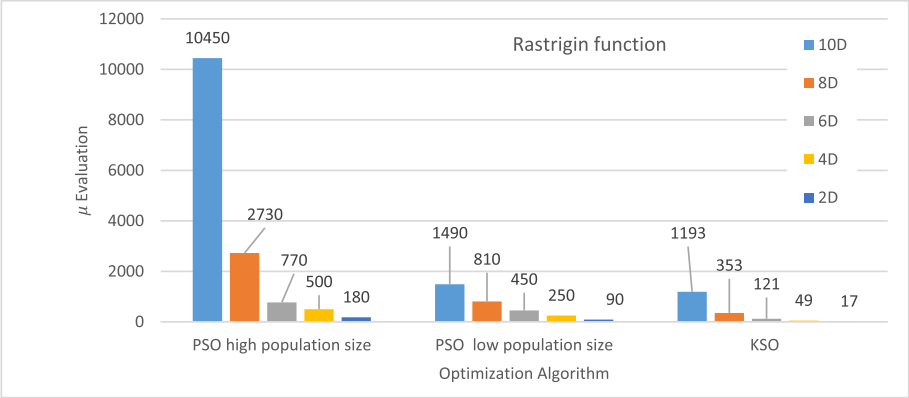

Fig. 14. Mean plot of function evaluations for d dimension Rastrigin problem.

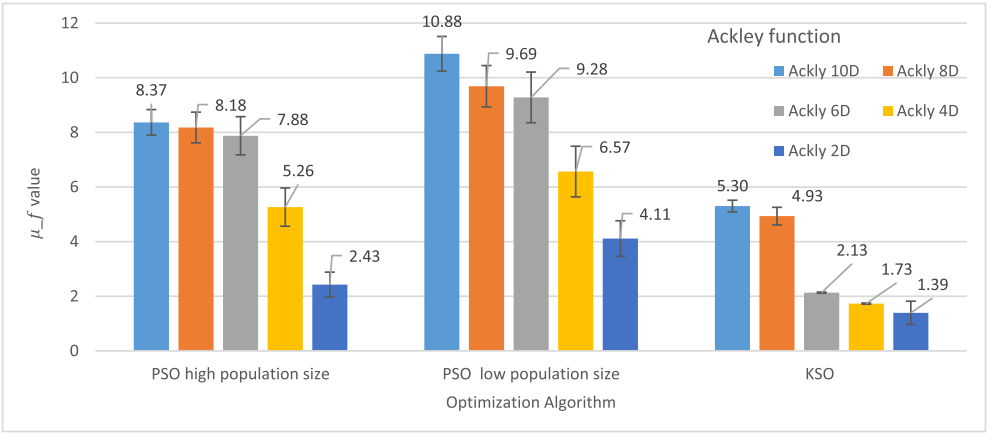

Fig. 15. Mean and quartile plot of objective function values for d dimension Ackley problem.

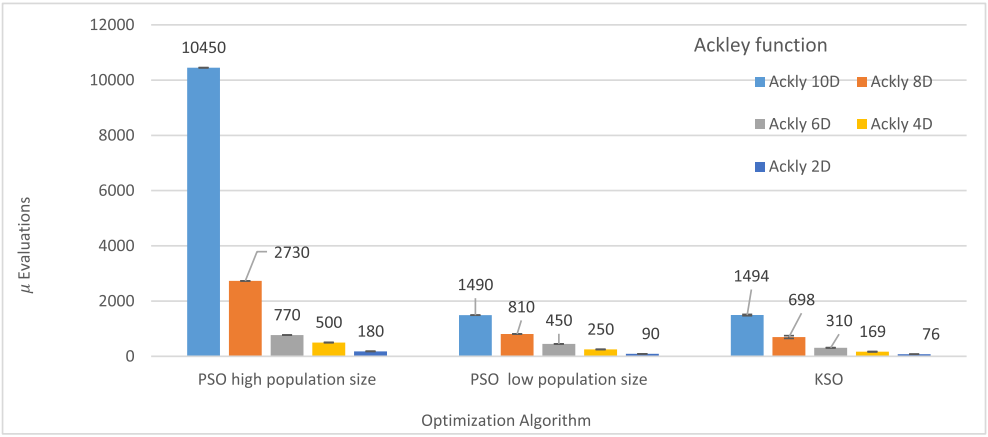

Fig. 16. Mean plot of function evaluations for d dimension Ackley problem.

lower evaluation number for proposed optimization algorithm compared to PSO.

In a similar plot of the Rastrigin problem, Figs. 15 and 16 represent the optimization results of Ackley problem for the objective function values and the number of function evaluations respectively. Both the mean and standard deviation of all the replications at the last iteration with different dimensionalities are presented for each figure. The result of the KSO algorithm is also presented and compared with the PSO algorithm for both high and low population size. As shown in Fig. 15, KSO algorithm gives a lot of improvement of the objective function values found in each replication. This is illustrated by the lower mean values observed for each dimension. These values are less than those obtained by PSO for the two-population size. These results imply that the KSO algorithms often find a lot of improving function values at the very beginning of replication runs and it settles at about the same function value, each with an extremely tight range at the last function evaluation. Fig. 16 shows that the dimensionality of the optimization problem highly depends on the number of function evaluations allowed. It can be noted also that the KSO algorithm gives the best improvement with a good performance in term of a number of function evaluations. This is valid for all dimensions. However, it can be mentioned that the optimization difficulties increase for all algorithms when the dimensionality increases.

These results make in evidence the robustness and the performance effectiveness of the proposed algorithm to resolve different benchmark

function chosen in terms of features, such as the high modality, larger search space, presence of hole, presence of Basin and for high dimensionality.

4. Application of KSO to polymer extrusion problem

4.1. Modeling and simulation

In the extrusion process, the flow of polymer melt through an extrusion die is characterized by a low Reynolds number. This low Reynolds number is due to the higher viscous forces compared to inertial forces. This typically occurs, when the viscosity is very high and the velocities are very low.

For all this reason, the polymer flow can be considered as a laminar flow, and the external forces can be neglected.

The extrusion simulation is carried out using COMSOL Multiphysics.

The mass, momentum, and energy conservation equations, are used to follow the material behavior, from which the velocity, pressure, and temperature fields are determined. The final governing equation is given as:

$$\begin{cases} \nabla.\left(2\eta\left(\dot{\bar{\gamma}}\right)\dot{\varepsilon}(v)\right) - \nabla \mathrm{P} = 0 \\ \nabla.v = 0 \\ \rho C_p \dfrac{dT}{dt} = -\nabla.\mathrm{q} + \sigma : \dot{\varepsilon}(v) \end{cases} \tag{21}$$

The behaviors laws used give an expression of the viscosity in function of the shear rate and temperature. In this paper, the geometry of a flat die is optimized for an LDPE 22H760. Carreau Yasuda/Arrhenius viscosity model is used to characterize the temperature and shear rate dependence. It is written as:

$$\eta(v,T) = \eta_0(T)\left[1 + \left(\eta_0(T)\frac{\dot{\bar{\gamma}}}{\tau_s}\right)^{\alpha}\right]^{\frac{m-1}{\alpha}} \tag{22}$$

In this model, α, m, $\eta_0(T)$ and τ_s are material constants, whereas $\eta_0(T)$ establishes the thermal dependency, given by the Arrhenius model:

$$\eta_0(T) = \eta_0(T_{ref}).e^{\left[\beta\left(\frac{1}{T}-\frac{1}{T_{ref}}\right)\right]} \tag{23}$$

where $\beta = \frac{E}{R}$ are material constants, and T_{ref} is the references temperature. The rheological parameters are given in Table 3.

Table 3
Rheological parameters.

η_0 Pa.s	m	α_0	τ_s Pa	B [K]	T_{ref} [K]
3328	0.02376	0.2648	12,648	4591	473

A flow rate of 50,000 mm^3/s was imposed on the entry with a temperature of 200 °C, and the temperature of the die is constant and equals to 210 °C. Due to the symmetry of the die, only ¼ of the part is modeled with two symmetry plans are used.

4.2. Formulation of the optimization problem

Balancing the distribution of flow through a die to achieve a uniform velocity distribution across the die exit is one of the most difficult tasks of extrusion die design. This optimization problem consists in determining an optimal geometry to homogenize the velocity distribution through the die exit, which corresponds to the minimum of the velocity dispersion E(x).

There exist a whole variety of criteria which are known to influence the correctness of the velocity distribution, but they are only known in a loose and hard-to-measure sense [5]. The relative importance of the homogeneous velocity distribution suggests the use of the velocity magnitude as a basis for the objective function. Lebaal et al. [17] propose to subdivide the outflow of the extrusion die into subsections and compute the dispersion. Siegbert et al. [48] subdivide the outflow of the extrusion die into several sections and compute the variance of the local maximal velocity compared to the average maximal velocity over all sections.

In this work, since the thickness of the entire outflow geometry is uniform, to analyze the flow distribution at the outlet section, the velocity distribution at the mid-plane at the die exit is used to evaluate the velocity dispersion. The optimization problem is defined as follows:

$$\begin{cases} \min \quad J(x) = \dfrac{E(x)}{E_0(x)} \\ \textit{Such that} \quad x^l \le x \le x^u \end{cases} \tag{24}$$

where $J$is the normalized objective function, the quality of the flow distribution was evaluated by the velocity dispersion E(x), that is always positive and becomes zero when all the exit velocity vector reaches the bulk average velocity, that is given by:

$$E(x) = \left(\frac{1}{N}\sum_{i=1}^{N}\left(\frac{|v_i(x) - \bar{v}(x)|}{\bar{v}(x)}\right)\right) \tag{25}$$

where$E_0$is the velocity dispersion in the initial die. N is the total number of nodes at the die exit in the middle plane and v_i is the velocity at an exit node. The critical spot as far as the velocity distribution is concerned is the die border. Here, the velocity is severely reduced due to the wall adhesion effect. For this, the velocities near the border are not taken into account to calculate the velocity dispersion and $\bar{v}$is the average exit velocity, which is defined as:

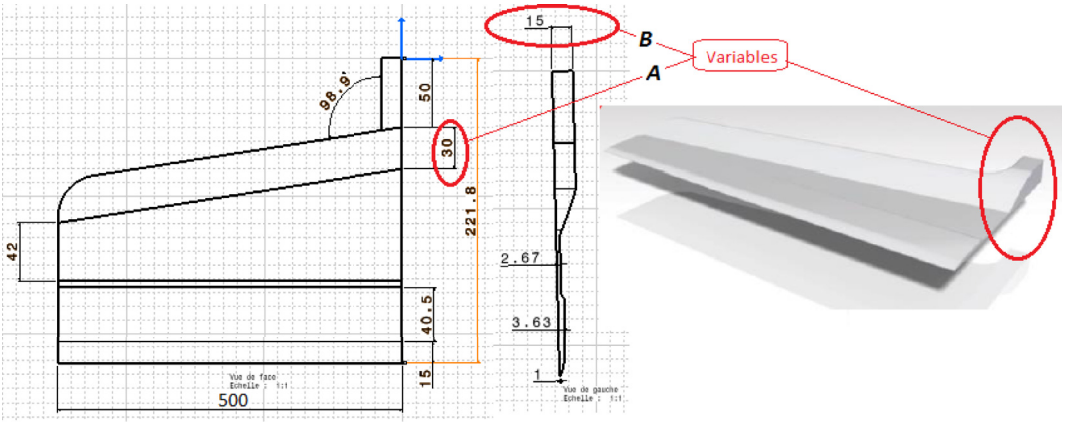

Fig. 17. Geometry of the flat die.

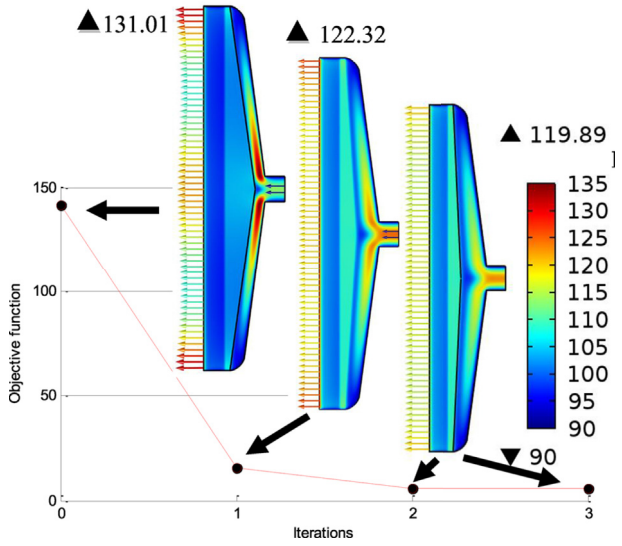

Fig. 18. Convergence history during an optimization process on the objective functions.

$$\bar{v} = \frac{1}{N}\sum_{i=1}^{N} v_i \tag{26}$$

The design of a flat die is based on the values of various geometrical parameters; the optimization variables used in this work are (A and B) correspond respectively to the depth of the channel repartition (A), and the second variable B represents the channel thickness (Fig. 17). During the optimization process, the variables A, B, vary in a limited field. Each variable has its respective geometrical limitations which are $30 \leq A \leq 110$ mm and $5 \leq B \leq 45$mm.

5. Result and discussion

The optimization strategy is applied to optimize the melt distributor geometry in order to achieve a uniform velocity distribution across the die exit.

A representative convergence history during an optimization process on the objective functions (J) is represented in Fig. 18. After 2 iterations, the objective function decreased significantly and the third iteration, the cut-off criteria fell below 10-3 ($\mathscr{F} = f^{*} - f^{sol} < 10^{-3}$) and the simulation stopped. The velocity distribution at the outflow is qualitatively shown in function of the convergence story of the optimization run in the same Figure (Fig. 18) and plotted over the die exit at the mid plane in Fig. 19.

Given the rheological parameters of the LDPE resin, and the process conditions, the optimal solution is obtained after two iterations. The objective function is then reduced to 96% of its initial value. The optimization algorithm used in this work clearly showed its ability to obtain the global optimal solution, with fast convergence and fewer function evaluations. A summary of the results is reported in Table 4.

Table 4
Summary of the results.

Optimization results	Initial die	Result
"J"	1	**0.0397**
Improvement	–	**96%**
Objective function	141.6	**5.62**
Pressure[bar]	13.82	**8.45**
Variable A [mm]	31	**78**
Variable B [mm]	30	**32**

It is clear that the velocity distribution in the initial geometry (Fig. 20a) is not homogeneous (arrow line). Quantitatively, Fig. 19 illustrates that in the middle of the die, exit velocities are higher about 125 [mm/s] and in parts close to the border of the die, it is observed an increase of velocities until 132 [mm/s], except in the die border the velocity is severely reduced due to the no-slip condition. On the other hand, in the region between the middle and the border of the die the velocities decrease until 110 [mm/s].

Figs. 19 and 20b illustrate that the melt distribution became homogeneous after optimization, all over the die exit.

In the initial die, because the width of the repartition channel is small (Variable A = 31 mm), the velocity in the manifold is higher and increases until 181 mm/s (see velocities magnitude in the slice) and this low manifold width increases the pressure drop. The pressure drop in the initial die (Fig. 21) is about 13.8 [bar]. However, in the optimal die, when the width of the repartition channel increase (A = 78 mm), the pressure drop decreases which gives a pressure of 8.45[bar].

The temperature distribution in both initial and optimal die is illustrated in Fig. 22. For the boundary conditions, the Die temperature is higher than the melt temperature. For this, it is observed that the

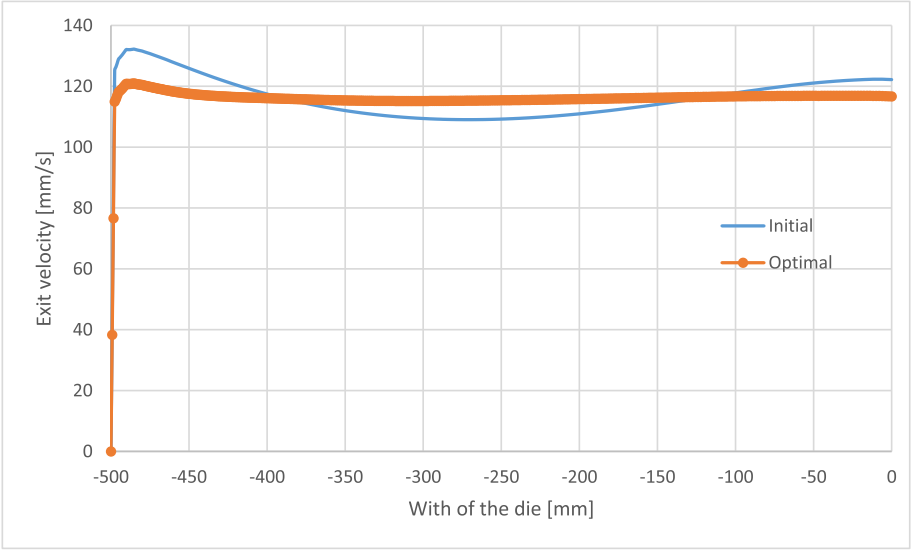

Fig. 19. Exit velocity distribution in mid plane for initial and optimal die.

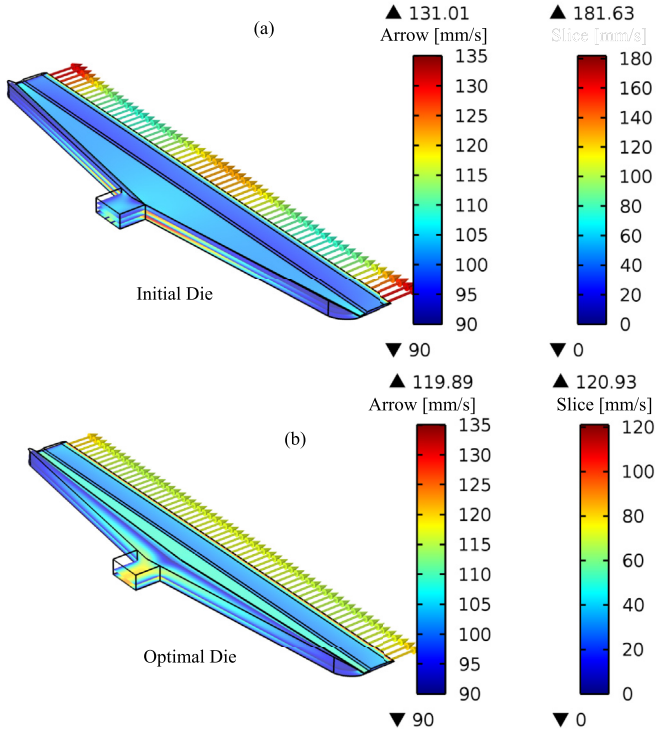

Fig. 20. Velocity distribution in initial and optimal dies.

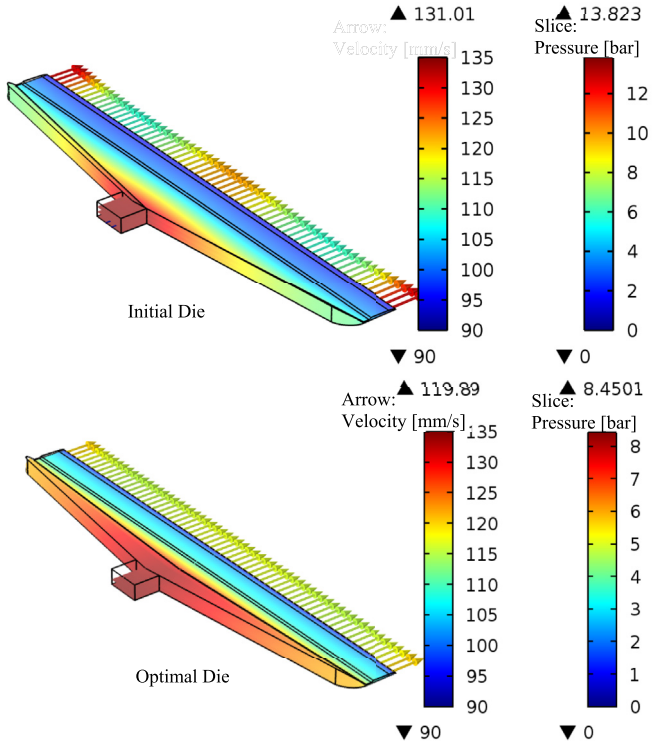

Fig. 21. Pressure distribution in initial and optimal dies.

temperature increases only in the small regions near the end of this die (in the border of the die). As a result of this increase in temperature, the viscosity decrease, which then affect the velocity. In Fig. 23 the velocity distribution is illustrated for two boundary conditions i.e., in the first case the die temperature (Tdie = 210 °C) is higher than the melt

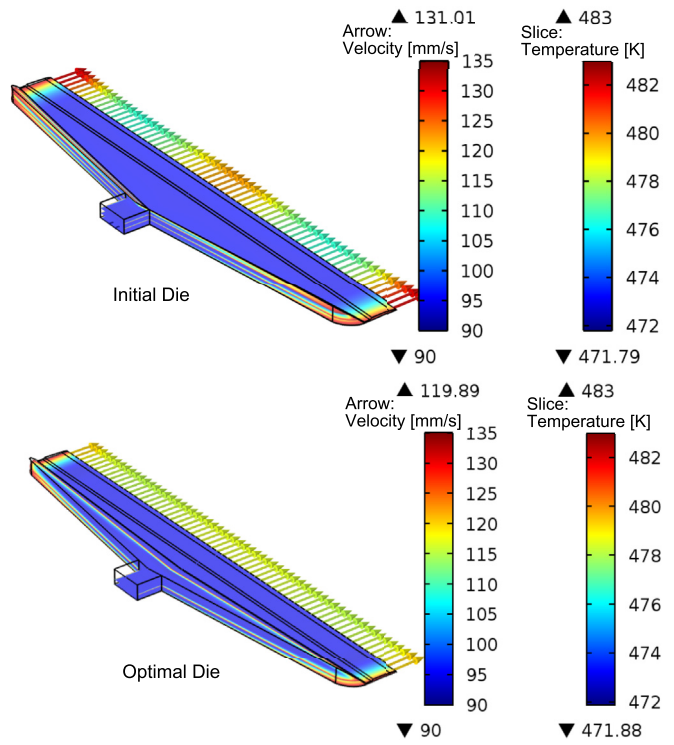

Fig. 22. Temperature distribution in initial and optimal dies.

temperature (Tmelt = 200 °C), for this case the velocity distribution is higher near the end of this die, but when both temperatures are constant, i.e., (Tdie = Tmelt = 200 °C), the velocity distribution becomes more homogeneous.

6. Conclusion

This study proposes a metamodel optimization system based on a smart sampling method and Knowledge Databases with Kriging interpolation. The aim of the proposed method called "Kriging Swarm Optimization" (KSO) is to combine two optimization algorithms and use a Knowledge Databases in order to decrease the number of functions evaluations needed to construct the metamodel.

The smart sampling method based on PSO and Knowledge Swarm Databases was adopted for the generation of samples. The essential of KSO is to combine two optimization algorithms. The first one is a metaheuristic algorithm where the solution is established on the displacement of all samples based on the best samples' information. The samples information is then used to construct the metamodel which is resolved by the gradient algorithm (SQP). The best solution is then used in the PSO algorithm such as the best solution to generate new samples for the new iteration of the KSO algorithm.

In order to increase the accuracy of the metamodel and at the same time reduce the sampling size, a knowledge database has been also integrated into the model management, for a share to avoid unnecessary FE calculations based on the history of the movement of each particle and also to enrich the collection (sampling) in each optimization iteration. This method makes it possible also to reduce the sampling size and at the same time increase the accuracy of the metamodel. At the same time, due to the stochastic characteristic of the sampling method, the optimization algorithm can avoid the local optimum.

Different benchmark functions chosen in terms of features such as modality for larger search space, dimensionality, the presence of a large hole and basins, was successfully minimized for validation of the

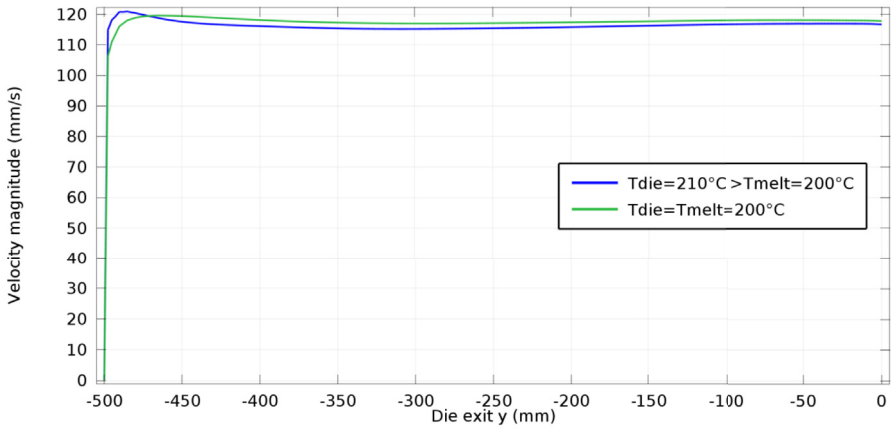

Fig. 23. Effect of die temperature on the optimal velocity distribution.

developed method. Finally, a practical engineering optimization problem for the design of die in polymer extrusion process was implemented with suggested Kriging Swarm Optimization algorithm (KSO). The optimal extrusion die is obtained starting from the second iteration and gives a very good result with a uniform exit velocity distribution.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.finel.2019.05.004.

References

[1] N. Lebaal, F. Schmidt, S. Puissant, D. Schläfli, Design of optimal extrusion die for a range of different materials, Polym. Eng. Sci. 49 (2009) 432–440, https://doi.org/10.1002/pen.21298.
[2] S. Elgeti, M. Probst, C. Windeck, M. Behr, W. Michaeli, C. Hopmann, Numerical shape optimization as an approach to extrusion die design, Finite Elem. Anal. Des. 61 (2012) 35–43, https://doi.org/10.1016/j.finel.2012.06.008.
[3] N. Lebaal, S. Puissant, F. Schmidt, Application of a response surface method to the optimal design of the wall temperature profiles in extrusion die, Int. J. Material Form. 3 (2009) 47–58, https://doi.org/10.1007/s12289-009-0416-x.
[4] C. Chen, P. Jen, F.S. Lai, Optimization of the coathanger manifold via computer simulation and an orthogonal array method, Polym. Eng. Sci. 37 (1997) 188–196.
[5] P. Liu, S. Xie, L. Cheng, Die structure optimization for a large, multi-cavity aluminum profile using numerical simulation and experiments, Mater. Des. 36 (2012) 152–160, https://doi.org/10.1016/j.matdes.2011.11.013.
[6] L. Pauli, M. Behr, S. Elgeti, Towards shape optimization of profile extrusion dies with respect to homogeneous die swell, J. Non-Newtonian Fluid Mech. 200 (2013) 79–87, https://doi.org/10.1016/j.jnnfm.2012.12.002.
[7] N.D. Gonçalves, O.S. Carneiro, J.M. Nóbrega, Design of complex profile extrusion dies through numerical modeling, J. Non-Newtonian Fluid Mech. 200 (2013) 103–110, https://doi.org/10.1016/j.jnnfm.2013.02.007.
[8] D. Wei, Z. Cui, J. Chen, Optimization and tolerance prediction of sheet metal forming process using response surface model, Comput. Mater. Sci. 42 (2008) 228–233, https://doi.org/10.1016/j.commatsci.2007.07.014.
[9] B. Lu, H. Ou, H. Long, Die shape optimisation for net-shape accuracy in metal forming revised manuscript , where j=using direct search and localised response surface methods, Struct. Multidiscip. Optim. 44 (2011) 529–545.
[10] H. Naceur, S. Ben-Elechi, J.L. Batoz, C. Knopf-Lenoir, Response surface methodology for the rapid design of aluminum sheet metal forming parameters, Mater. Des. 29 (2008) 781–790, https://doi.org/10.1016/j.matdes.2007.01.018.
[11] W. Hu, L. Enying, L.G. Yao, Optimization of drawbead design in sheet metal forming based on intelligent sampling by using response surface methodology, J. Mater. Process. Technol. 206 (2008) 45–55, https://doi.org/10.1016/j.jmatprotec.2007.12.002.
[12] G. Sun, G. Li, Z. Gong, X. Cui, X. Yang, Q. Li, Multiobjective robust optimization method for drawbead design in sheet metal forming, Mater. Des. 31 (2010) 1917–1929, https://doi.org/10.1016/j.matdes.2009.10.050.
[13] N. Lebaal, S. Puissant, F.M. Schmidt, Rheological parameters identification using in situ experimental data of a flat die extrusion, J. Mater. Process. Technol. 164–165 (2005) 1524–1529, https://doi.org/10.1016/j.jmatprotec.2005.02.218.
[14] H. Naceur, Y.Q. Guo, S. Ben-Elechi, Response surface methodology for design of sheet forming parameters to control springback effects, Comput. Struct. 84 (2006) 1651–1663.
[15] H. Wang, L. Tang, G.Y. Li, Adaptive MLS-HDMR metamodeling techniques for high dimensional problems, Expert Syst. Appl. 38 (2011) 14117–14126, https://doi.org/10.1016/j.eswa.2011.04.220.
[16] S. Chakraborty, A. Sen, Adaptive response surface based efficient finite element model updating, Finite Elem. Anal. Des. 80 (2014) 33–40, https://doi.org/10.1016/j.finel.2013.11.002.
[17] N. Lebaal, S. Puissant, F. m. Schmidt, D. Schläfli, An optimization method with experimental validation for the design of extrusion wire coating dies for a range of different materials and operating conditions, Polym. Eng. Sci. 52 (2012) 2675–2687, https://doi.org/10.1002/pen.23203.
[18] N. Lebaal, M. Oudjene, S. Roth, The optimal design of sheet metal forming processes: application to the clinching of thin sheets, Int. J. Comput. Appl. Technol. 43 (2012) 110–116.
[19] M. Oudjene, L. Ben-Ayed, A. Delamézière, J.-L. Batoz, Shape optimization of clinching tools using the response surface methodology with Moving Least-Square approximation, J. Mater. Process. Technol. 209 (2009) 289–296, https://doi.org/10.1016/j.jmatprotec.2008.02.030.
[20] S.J. Yakowitz, F. Szidarovszky, A comparison of kriging with nonparametric regression methods, J. Multivar. Anal. 16 (1985) 21–53, https://doi.org/10.1016/0047-259X(85)90050-8.
[21] W. Hao, W. Shaoping, M.M. Tomovic, Modified sequential kriging optimization for multidisciplinary complex product simulation, Chin. J. Aeronaut. 23 (2010) 616–622, https://doi.org/10.1016/S1000-9361(09)60262-4.
[22] E. Roux, P.-O. Bouchard, Kriging metamodel global optimization of clinching joining processes accounting for ductile damage, J. Mater. Process. Technol. 213 (2013) 1038–1047, https://doi.org/10.1016/j.jmatprotec.2013.01.018.
[23] M. Azaouzi, N. Lebaal, G. Rauchs, S. Belouettar, Optimal design of multi-step stamping tools based on response surface method, Simulat. Model. Pract. Theor. 24 (2012) 1–14, https://doi.org/10.1016/j.simpat.2012.01.006.
[24] N. Tanguy, N. Iassamen, M. Telescu, P. Cloastre, Parameter optimization of orthonormal basis functions for efficient rational approximations, Appl. Math. Model. 39 (2015) 4963–4970, https://doi.org/10.1016/j.apm.2015.04.017.
[25] S. Jakobsson, B. Andersson, F. Edelvik, Rational radial basis function interpolation with applications to antenna design, J. Comput. Appl. Math. 233 (2009) 889–904, https://doi.org/10.1016/j.cam.2009.08.058.
[26] G. Sun, G. Li, Q. Li, Variable fidelity design based surrogate and artificial bee colony algorithm for sheet metal forming process, Finite Elem. Anal. Des. 59 (2012) 76–90, https://doi.org/10.1016/j.finel.2012.04.012.
[27] J.H. Friedman, Multivariate adaptive regression Splines, Ann. Stat. 19 (1991) 1–67.
[28] S. Chen, C.F.N. Cowan, P.M. Grant, Orthogonal least squares learning algorithm for radial basis function networks, IEEE Trans. Neural Netw. 2 (1991) 302–309, https://doi.org/10.1109/72.80341.
[29] D.J. Fonseca, D.O. Navaresse, G.P. Moynihan, Simulation metamodeling through artificial neural networks, Eng. Appl. Artif. Intell. 16 (2003) 177–183.
[30] H.M. Gomes, A.M. Awruch, P.A.M. Lopes, Reliability based optimization of laminated composite structures using genetic algorithms and artificial neural networks, Struct. Saf. 33 (2011) 186–195, https://doi.org/10.1016/j.strusafe.2011.03.001.
[31] Y.F. Li, S.H. Ng, M. Xie, T.N. Goh, A systematic comparison of metamodeling techniques for simulation optimization in Decision Support Systems, Appl. Soft Comput. 10 (2010) 1257–1273, https://doi.org/10.1016/j.asoc.2009.11.034.
[32] T.W. Simpson, A.J. Booker, D. Ghosh, A.A. Giunta, P.N. Koch, R.-J. Yang, Approximation methods in multidisciplinary analysis and optimization: a panel discussion, Struct. Multidiscip. Optim. 27 (2004), https://doi.org/10.1007/s00158-004-0389-9.
[33] G.G. Wang, S. Shan, Review of metamodeling techniques for product design with computation-intensive processes, Proc. Can. Eng. Educ. Assoc. (2011). http://queens.scholarsportal.info/ojs/index.php/PCEEA/article/view/3940. (Accessed 13 May 2016).

[34] M. Diez, E.F. Campana, F. Stern, Design-space dimensionality reduction in shape optimization by Karhunen–Loève expansion, Comput. Methods Appl. Mech. Eng. 283 (2015) 1525–1544, https://doi.org/10.1016/j.cma.2014.10.042.
[35] G.G. Wang, Adaptive response surface method using inherited Latin hypercube design points, J. Mech. Des. 125 (2003) 210–220, https://doi.org/10.1115/1.1561044.
[36] M. Azaouzi, N. Lebaal, A. Makradi, S. Belouettar, Optimization based simulation of self-expanding Nitinol stent, Mater. Des. 50 (2013) 917–928.
[37] W. Hu, L. Enying, L.G. Yao, Optimization of drawbead design in sheet metal forming based on intelligent sampling by using response surface methodology, J. Mater. Process. Technol. 206 (2008) 45–55, https://doi.org/10.1016/j.jmatprotec.2007.12.002.
[38] H. Wang, E. Li, G.Y. Li, The least square support vector regression coupled with parallel sampling scheme metamodeling technique and application in sheet forming optimization, Mater. Des. 30 (2009) 1468–1479, https://doi.org/10.1016/j.matdes.2008.08.014.
[39] H. Wang, G. Li, E. Li, A comparative study of boundary-based intelligent sampling approaches for nonlinear optimization, Appl. Soft Comput. 11 (2011) 2227–2238, https://doi.org/10.1016/j.asoc.2010.08.002.
[40] N. Lebaal, F. Schmidt, S. Puissant, Optimisation of extrusion flat die design and die wall temperature distribution, using Kriging and response surface method, Int. J. Mater. Prod. Technol. 38 (2010) 307. Nos 23 2010 307. 38.
[41] G. Jia, A.A. Taflanidis, Kriging metamodeling for approximation of high-dimensional wave and surge responses in real-time storm/hurricane risk assessment, Comput, Methods Appl. Mech. Eng. 261–262 (2013) 24–38, https://doi.org/10.1016/j.cma.2013.03.012.
[42] J. Kennedy, R. Eberhart, Particle swarm optimization, in: IEEE Int. Conf. Neural Netw. 1995 Proc. 4, 1995, pp. 1942–1948, https://doi.org/10.1109/ICNN.1995.488968.
[43] N. Di Cesare, D. Chamoret, M. Domaszewski, A new hybrid PSO algorithm based on a stochastic Markov chain model, Adv. Eng. Software 90 (2015) 127–137, https://doi.org/10.1016/j.advengsoft.2015.08.005.
[44] Y. Zhang, Z. An, J. Zhou, Optimization design of flash structure for forging die based on Kriging-PSO strategy, in: Adv. Swarm Intell., Springer, 2010, pp. 373–381, accessed, http://link.springer.com/chapter/10.1007/978-3-642-13495-1_46. (Accessed 27 May 2016).
[45] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: IEEE Int. Conf. Evol. Comput. Proc. IEEE World Congr. Comput. Intell. Cat No98TH8360, 1998, 1998, pp. 69–73, https://doi.org/10.1109/ICEC.1998.699146.
[46] A. Kaveh, V.R. Mahdavi, Optimal domain decomposition using Colliding Bodies Optimization and k-median method, Finite Elem. Anal. Des. 98 (2015) 41–49, https://doi.org/10.1016/j.finel.2015.01.010.
[47] N. Lebaal, M. Nouari, A. Ginting, A new optimization approach based on Kriging interpolation and sequential quadratic programming algorithm for end milling refractory titanium alloys, Appl. Soft Comput. 11 (2011) 5110–5119, https://doi.org/10.1016/j.asoc.2011.05.048.
[48] R. Siegbert, N. Yesildag, M. Frings, F. Schmidt, S. Elgeti, H. Sauerland, M. Behr, C. Windeck, C. Hopmann, Y. Queudeville, U. Vroomen, A. Bührig-Polaczek, Individualized production in die-based manufacturing processes using numerical optimization, Int. J. Adv. Manuf. Technol. 80 (2015) 851–858.