Int J Adv Manuf Technol (2015) 78:525–536
DOI 10.1007/s00170-014-6643-4

ORIGINAL ARTICLE

Multiobjective optimization of process parameters for plastic injection molding via soft computing and grey correlation analysis

Gang Xu · Zhitao Yang

Received: 3 July 2014 / Accepted: 23 November 2014 / Published online: 7 December 2014

Abstract In the plastic injection molding (PIM), the optimization of the process parameters is a complex task. The objective of this study is to propose an intelligent approach for efficiently optimizing PIM parameters when multiple objectives are involved, where different objectives, such as minimizing part weight, flash, or volumetric shrinkage, present trade-off behaviors. Multiple objective functions reflecting the product quality are constructed for the optimization model of PIM parameters. The proposed approach integrates Taguchi's parameter design method, back-propagation neural network (BPNN), grey correlation analysis (GCA), particle swarm optimization (PSO) and multiobjective particle swarm optimization (MOPSO) to locate the Pareto optimal solution for multiobjective optimization problem. PSO and GCA are applied to optimize the network structure of BPNN to establish multiobjective mathematical model (PSO-GCANN) that finely maps the relationship between the input process parameters and output multiresponse. MOPSO is used to fine-tune the Pareto optimal solutions while the approximate PSO-GCANN is utilized to efficiently compute the fitness of every individual during the evolution of MOPSO. The illustrative application and comparison of results show that the proposed methodology outperforms the existing methods and can help mold designers to efficiently and effectively identify optimal process parameters.

Keywords Plastic injection molding · Back-propagation neural networks · Grey correlation analysis · Particle swarm optimization algorithm · Multiobjective optimization

1 Introduction

Injection molding is the most widely used for producing plastic products. In the process, many parameters such as melt temperature, mold temperature, hold pressure, cooling time, etc. are very important; they have direct influence on the product quality and manufacturing cost. The optimization of process parameters is a complex and difficult task. Traditionally, the process parameters are often determined by experienced engineers or based on reference handbooks. Then the process parameters are improved and fine-tuned by trial and error or Taguchi's parameter design method. These methods depend greatly on the experience of molding operators. Especially, they could potentially be costly and time-consuming in new resins or new applications. So it is not suitable for complex manufacturing processes [1]. Taguchi's parameter design method can only find the best specified process parameter level combination which includes the discrete setting values of process parameters [2]. Furthermore, when engineers deal with a multiresponse process parameter design problem, the conventional Taguchi parameter design method runs into difficulties. Because the optimization of the process parameters can be considered to be a "black art," some surrogate models are employed, such as response surface methodology, artificial neural network (ANN), support vector regression, Gaussian process, etc. [3–9]. In these surrogate models, a mathematical approximation is constructed to improve conventional Taguchi's parameter design. These surrogate models are capable of effectively treating continuous parameter values and have the ability to learn arbitrary nonlinear mappings between noisy sets of input and output data. With the

G. Xu (✉)
Department of Mathematics, Nanchang University,
Nanchang 330031, China
e-mail: xgang_csu@163.com

Z. Yang
National Engineering Research Center of Novel Equipment for Polymer Processing, South China University of Technology,
Guangzhou 510641, China

development of artificial intelligence, soft computing has been applied widely to optimize process parameters [10–13]. Combining artificial neural network and genetic algorithm (ANN/GA) method is proposed to optimize the injection molding process [12]. Chen [11] developed a self-organizing map plus a back-propagation neural network model for predicting product quality. Azlan [10] proposed a new approach integrating ANN and GA for estimating a minimum value of machining performance. However, such single response requirement rarely exists in practical production process. Generally, there are multiresponse requirements in product production.

An increase in injection temperature causes a decrease in melt viscosity, which results in reduced cavity pressure and shear stress. On the other hand, high melt temperature increases cooling time which lowers productivity. Too short injection time increases the cavity pressure and shear stress although it can reduce temperature difference. On the contrary, a long injection time will lead to a decrease in the flow front temperature, as well as an increase in melt viscosity and cavity pressure. It is clear that there are multiresponse requirements in production. An optimization algorithm must trade off these conflicting process parameters to obtain optimum parameters which produce a high-quality part at minimum cost. For the process parameter design problem of a multiple-input multiple-output (MIMO) production process, many researchers have developed and employed different optimization schemes for determining the optimal process parameters for polymer processing [14–16]. Huang [15] presented an approach for determining parameter values in melt spinning processes to yield optimal qualities of denier. Castro [14] used an approach comprising computer simulation, ANN, and data envelopment analysis (DEA) to determine the proper operating parameters for finding the best compromise among several conflicting performance measures. Chen [16] presented an approach in a soft computing paradigm for the process parameter optimization of MIMO plastic injection molding process. Usually, for many MIMO production processes, the researchers transform the multiobjective optimization problems to single objective optimization problems and apply surrogate models or evolutionary algorithms to attain the final optimal process parameter settings.

Recently, Zhou [17] presented the development of an integrated simulation-based optimization system to adaptively search for the Pareto optimal solutions to different objective functions. Wei [18] discussed the combination of the design method to solve the complex multiobjective optimal performance design of large-scale injection molding machines to find a much better spread of design solutions and better convergence near the true Pareto optimal front. A triple-objective optimization model to determine the Pareto optimal solutions by eliminating the uncertainty in the artificial priority election are proposed [19]. Solimanpur [20] presented a new optimization technique based on GA to find multiple solutions along the Pareto optimal front in machining operations. Cheng [21] proposed an intelligent methodology for efficiently optimizing the injection molding parameters when multiple constraints and multiple objectives are involved. Although the abovementioned researches have achieved various levels of success, more efforts should be taken to search an intelligent optimization strategy for efficiently optimizing the plastic injection molding (PIM) parameters when multiple objectives are involved.

For MIMO plastic injection molding, this research proposes an intelligent process parameter optimization approach to help manufacturers to determine the final optimal process parameter settings of PIM to achieve a competitive advantage of product quality. The proposed approach integrates Taguchi's parameter design method, back-propagation neural network (BPNN), grey correlation analysis (GCA), particle swarm optimization (PSO), and multiobjective particle swarm optimization (MOPSO) to locate the Pareto optimal solutions. More specifically, the proposed approach has two phases. First, Taguchi's parameter design method is used to effectively provide the training and testing data, the PSO plus GCA are applied to optimize the network structure of BPNN to establish PSO-GCANN model that finely map the relationship between the input process parameters and output multiresponse. Second, the finished PSO-GCANN model is employed to compute the values of multiresponse. Then MOPSO is applied to search for the Pareto optimal set based on the PSO-GCANN model. We have conducted comparison experiments to demonstrate the efficacy of the proposed intelligent approach. The final optimal process parameter settings are selected from the Pareto optimal set according to fuzzy sets theory [22] for setting up the process parameters and are not limited discrete value as in Taguchi's parameter design method.

The rest of this paper is organized as follows. Sect. 2 describes the optimization methodologies including neural networks, grey correlation analysis, particle swarm optimization, and multiobjective optimization. The intelligent process parameter optimization approach for locating the Pareto optimal solutions to the optimization problem and selecting an optimization solution according to fuzzy sets theory are put forward in Sect. 3. Then, Sect. 4 presents an illustrative case study that demonstrates the effectiveness of the proposed approach. The advantages of the proposed approach over the existing methods are also discussed in detail. Finally, the work is concluded in Sect. 5.

2 Optimization methodologies

The optimization methodologies including BPNN, grey correlation analysis, particle swarm optimization, and multiobjective particle swarm optimization for developing the proposed intelligent approach are briefly introduced below.

2.1 Back-propagation neural network

Neural network (NN) has been used in control, forecasting, manufacturing, optimization, etc. In numerous literatures, BPNN is adopted because it has the advantage of fast response and high learning accuracy [11]. Woll and Cooper [23] reported that all the nonlinear mappings could be approximated by BPNN with single hidden layer, so the BPNN used in this paper is composed of one input layer, one output layer, and one hidden layer. The number of input neurons in BPNN equals the number of injection molding parameters to be optimized, namely, N_I. The number of output neurons in BPNN equals the number of objective functions, N_O. The number of hidden neurons in BPNN can be determined by the experiential equation $N_\mathrm{H} = \sqrt{N_\mathrm{I} + N_\mathrm{O}} + \alpha \ (1 \le \alpha \le 10)$ [21]. The hyperbolic tangent function is utilized for all the transfer functions in BPNN in this paper. The superiority of a network's function approach depends on the network structure, parameters, and problem complexity. If inappropriate network structure and parameters are selected, the analysis results of the network may be undesirable. Conversely, the analysis results will be more significant if appropriate network structure and parameters are selected.

2.2 Grey correlation analysis

The principle of the GCA [24] is based on the macro- or microgeometric approach between the behavior factors. The more similar are the array curves, the closer connection they have. The detailed calculation formulas are as follows.

The arrays are as follows:

$$\begin{aligned} &\left\{X_0^{(0)}(r)\right\}, \quad r = 1,2,3,\ldots,N_0 \\ &\left\{X_1^{(0)}(r)\right\}, \quad r = 1,2,3,\ldots,N_1 \\ &\left\{X_2^{(0)}(r)\right\}, \quad r = 1,2,3,\ldots,N_2 \\ &\ldots \\ &\left\{X_k^{(0)}(r)\right\}, \quad r = 1,2,3,\ldots,N_k \end{aligned}$$

where $N_1, N_2, \ldots, N_k$ belong to natural number and may be not equal. The k arrays express k factors. The array $\{X_0^{(0)}(r)\}$ is assigned to main array and $\{X_m^{(0)}(r)\}$ $(m=1,2,\ldots,k)$ to sub-arrays. The average of $\{X_m^{(0)}(r)\}$, $r=1,2,\ldots,N_m$, $m=0,1,2,\ldots,k$ is $\overline{X_m} = \frac{1}{N_m}\left[\sum_{r=1}^{N_m} X_m^{(0)}(r)\right]$.

The conversion $Y_m(r) = X_m^{(0)}(r)/\overline{X_m}$ is made and the following arrays called inverted arrays can be obtained:

$$\begin{aligned} &\{Y_0(r)\}, \quad r = 1,2,3,\ldots,N_0 \\ &\{Y_1(r)\}, \quad r = 1,2,3,\ldots,N_1 \\ &\{Y_2(r)\}, \quad r = 1,2,3,\ldots,N_2 \\ &\ldots \\ &\{Y_k(r)\}, \quad r = 1,2,3,\ldots,N_k \end{aligned}$$

In fact, the transformation from $X_m^{(0)}(r)$ into $Y_m(r)$ can be regarded as a reflection. The grey correlation coefficient $\xi_{0m}(r)$ at $t=r$ is

$$\xi_{0m}(r) = \frac{\min_m \min_r \left|Y_0(r) - Y_m(r)\right| + \rho \max_m \max_r \left|Y_0(r) - Y_m(r)\right|}{\left|Y_0(r) - Y_m(r)\right| + \rho \max_m \max_r \left|Y_0(r) - Y_m(r)\right|} \tag{1}$$

ρ is recognition coefficient in the formula and $\rho \in [0,1]$, defined as 0.5 generally.

The grey correlation degree of Y_m and Y_0 is

$$\varsigma_{0m} = \left[\sum_{r=1}^{N} \xi_{0m}(r)\right] / N \tag{2}$$

The grey correlation degree is a quantitative value of the correlation between the factors. If the value of grey correlation degree is higher, the main factor and sub-factor are more relevant.

2.3 Particle swarm optimization algorithm

The PSO algorithm [25] works by initializing a flock of birds randomly over the searching space. Each particle successively adjusts its position toward the global optimum according to the two factors: the best position encountered by itself (pBest) and the best position found so far by the whole swarm (gBest). Supposing searching in a D-dimensional hyperspace, the position of the ith particle can be presented by a vector $X_i=(x_{i1}, x_{i2}, \ldots, x_{iD})$, and its velocity is represented as a vector $V_i=(v_{i1}, v_{i2}, \ldots, v_{iD})$. The velocity and position of particle i at next iteration are calculated according to the following equations:

$$v_{id}^{t+1} = w v_{id}^t + c_1 r_{1d}\left(\mathrm{pBest}_{id}^t - x_{id}^t\right) + c_2 r_{2d}\left(\mathrm{gBest}_d^t - x_{id}^t\right) \tag{3}$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{4}$$

where t is the index of the current generation; w is the inertia weight and usually decreases linearly from 0.9 to 0.4 during the run time [26]; c_1 and c_2 are two positive acceleration coefficients; r_{1d} and r_{2d} are two uniformly distributed random numbers in the interval [0, 1] for the dth dimension separately.

2.4 Multiobjective optimization of unitary performance

2.4.1 Mathematical modeling of optimization problem

Supposing that there are N_I design variables, N_O objective functions, and N_C constraints to be considered in the optimization of injection molding, the multiobjective optimization problem can be expressed as

$$\begin{cases} \min\ y = \min\ F(x) = \min\left(f_1(x),\ f_2(x), \ldots, f_{N_\mathrm{O}}(x)\right) \\ s.t.\ x \in X = \left\{x \in R^n \middle| g_i(x) \le 0,\ i = 1, 2, \cdots, N_\mathrm{C}\right\} \end{cases} \tag{5}$$

Where $x = (x_1, x_2, \ldots, x_{N_\mathrm{I}})$ is the vector of design variables with its elements being process parameters; $X = \{(x_1, x_2, \ldots, x_{N_\mathrm{I}}) | l_i \le x_i \le u_i, i = 1, 2, \cdots, N_\mathrm{I}\}$ is the feasible domain of vector of design variables x with $L = (l_1, l_2, \ldots, l_{N_\mathrm{I}})$ and $U = (u_1, u_2, \ldots, u_{N_\mathrm{I}})$ being lower and upper limits, respectively; $y = (y_1, y_2, \ldots, y_{N_\mathrm{O}}) = (f_1(x),\ f_2(x), \ldots, f_{N_\mathrm{O}}(x)) \in Y$ is the objective vector and Y is the objective space composed of multiple subspaces corresponding to different objective functions; $g_i(x) \le 0$ (i=1,2,⋯,N_C) are constraint functions.

2.4.2 MOPSO for locating Pareto optimal solutions

The optimization problem of injection molding expressed in Eq. (5) has multiple objective functions and constraints. There is no single optimal solution but rather a set of compromise solutions named Pareto optimal or nondominated solutions to such an optimization problem with multiple conflicting objective functions. Among the Pareto optimal solutions, one solution is worse with regard to at least one other objective function if it is better with regard to an objective function. Thus, the final injection molding scheme should be determined by designers on the basis of the moldability evaluations of various Pareto optimal solutions or fuzzy sets theory shown in Sect. 2.4.3. For the constrained multiobjective optimization problem defined in Eq. (5), a solution $x^* \in X$ is said to be Pareto optimal if there is no solution $x \in X$ such that $f_i(x) \le f_i(x^*)$ for all i=1,2,⋯,N_O with strict inequality for at least one i. Any other feasible solution $x \in X$ with $f_i(x) \ge f_i(x^*)$ for all i=1,2,⋯,N_O is an inferior solution. The Pareto front of the optimization problem in Eq. (5) can be obtained by plotting all its Pareto optimal solutions according to their objective values, which is a $N_\mathrm{O}-1$ dimensional surface.

In the general case, it is difficult to get an analytical expression of the line or the surface that contains all points of the Pareto front. The MOPSO [27] is applied to resolve the constrained multiobjective optimization problem in Eq. (5), where MOPSO integrates a powerful PSO with the concept of Pareto optimality to automatically find out solutions illustrative of the nondominated set. Algorithm process of MOPSO can be described as follows: Initial population POP including N individual values are random in the bounding range. First, according to optimization goal and constraint condition, population ranking will be done and the nondominated individuals of the population are stored in external repository. The population will be evolved through the update of velocities and positions of particles. The population combines with the nondominated individuals in external repository and the sorting will be calculated. Then all the nondominated individuals of the population are stored in external repository again. It is a completed process of MOPSO algorithm. When the previously set maximum generation is got by circulation, the algorithm will be ended and the Pareto optimal set will be achieved.

2.4.3 Pareto optimizing based on fuzzy sets theory

In the final step, an optimization solution will be selected out from the Pareto sets which are calculated by MOPSO. Because manual Pareto-optimizing contains several uncertain subjective factors, a Pareto sets optimization method based on fuzzy sets theory [22] is used. Member function f_m is defined as a proportion of number 1 target in one solution:

$$f_\mathrm{m} = \begin{cases} 0 & f_i > f_i^{\max} \\ \dfrac{f_i^{\max} - f_i}{f_i^{\max} - f_i^{\min}} & f_i^{\min} < f_i < f_i^{\max} \\ 1 & f_i > f_i^{\min} \end{cases} \tag{6}$$

For each nondomination solution R in Pareto sets, domination function f^R could be defined as below:

$$f^R = \sum_{i=1}^{N} f_i^R \Big/ \sum_{j=1}^{N} \sum_{i=1}^{N} f_j^R \tag{7}$$

N is the number of solution. The larger value of f^R is the better unitary performance of that solution. Therefore, the solution with maximum f^R would be chosen as an optimal solution from the Pareto optimal set. By sorting the Pareto optimal set into a depending order according to the value of f^R, optimization sequence of feasible solution can be achieved.

3 Optimization methodology for injection molding parameters

This research proposes an intelligent approach to effectively assist engineers in the process parameter optimization for MIMO plastic injection molding. The proposed approach integrates Taguchi's parameter design method, BPNN, GCA, PSO, and MOPSO to locate the Pareto optimal solutions for the multiobjective optimization problem. Taguchi's parameter design method is used to arrange an orthogonal array experiment and to reduce the number of experiments. Subsequently, the signal-to-noise ratio (*S*/*N* ratio) is employed to determine the process parameter settings that have minimal sensitivity of noise under the consideration of most major quality characteristics. In this research, the GCA is used to determine the hidden neuron number of BPNN to realize the optimization of BPNN structure. Generally, the PSO algorithm is utilized to obtain the appropriate connection weights and threshold values for BPNN [28]. Moreover, compared with other algorithms such as the steepest descent algorithm, PSO converges rapidly with less training cycles when the error energy is minimized during the training phase of BPNN. The training of the network will be terminated either when the maximum training time is reached or when the mean square error (MSE) between the desired values and network outputs is reduced to a given level. By means of the functions of PSO and GCA, the PSO and GCA are simultaneously applied to optimize BPNN to establish a PSO-GCANN model which finely maps the relationship between the input process parameters and output multiresponse. In this process, the experimental data of Taguchi's parameter design method are used to effectively train and test the PSO-GCANN model. Subsequently, MOPSO is applied to the finished PSO-GCANN model to search solutions illustrative of the nondominated set of process parameters. Finally, the confirmation experiments are performed to confirm the effectiveness of Pareto optimal process parameter settings based on the PSO-GCANN model, and the final optimal process parameter settings from Pareto optimal set are determined according to fuzzy sets theory. The multiobjective optimization procedure for process parameters of injection molding includes the following two stages: the training stage shown in Fig. 1 and the iteration optimization stage shown in Fig. 2.

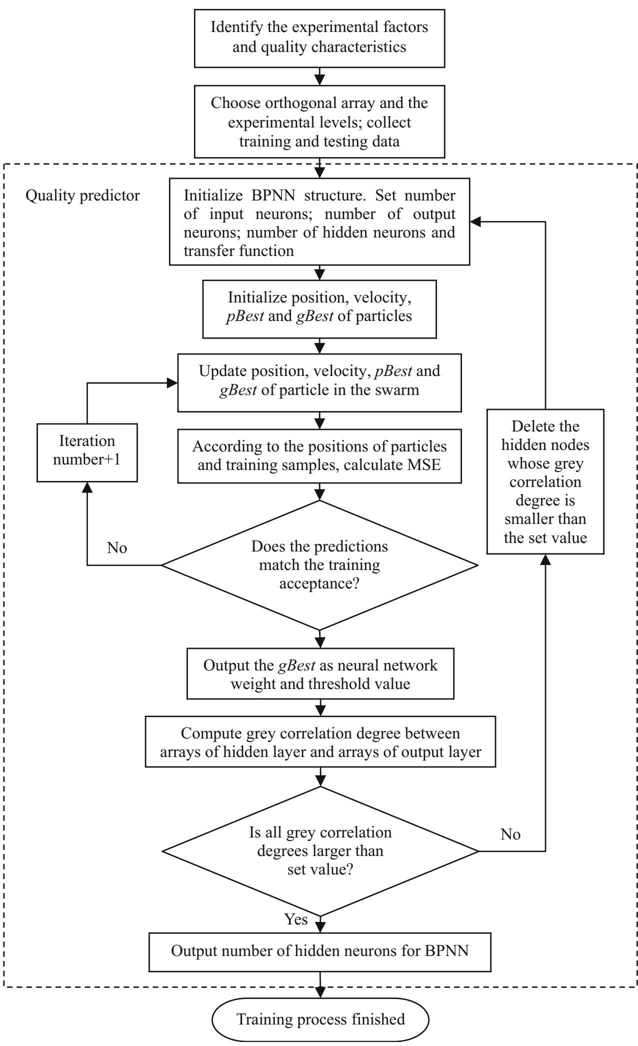

Fig. 1 Training stage of optimization procedure using PSO plus GCA

3.1 Training stage

Step 1. Identify feasible responses (quality characteristics) as the target requirements of the experiment. The responses which have significant influence on final product quality must be confirmed. Moreover, engineers need to choose the most major responses from all responses just identified by expert opinion or experience.

Step 2. Determine the feasible and tractable process parameters and levels that influence the performance of responses. Select an appropriate orthogonal array for arranging the experiment and acquiring the experimental treatments.

Step 3. Perform experiments for each treatment and collect the performance measurement of the responses.

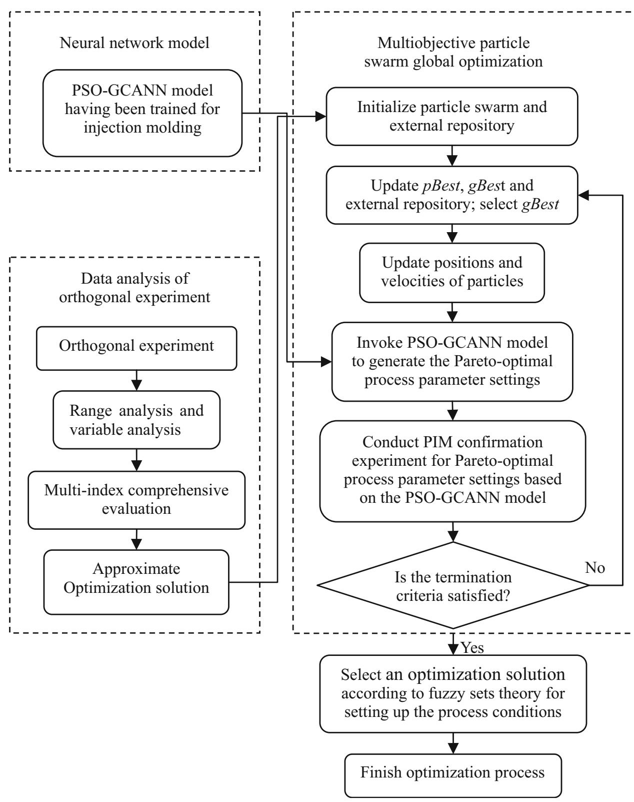

Fig. 2 Iteration stage of optimization procedure

Step 4. Use PSO plus GCA based on BPNN to develop PSO-GCANN model that finely maps the relationship between the input process parameters and output multiresponse.

Step 5. Formulate a multiobjective fitness function used in MOPSO search approach.

3.2 Iteration optimization stage

Step 1. Formulate the fitness function of MOPSO with ranges of the process parameters and set the MOPSO parameters.

Step 2. Determine the Pareto optimal process parameter settings via soft computing. Use MOPSO to search the Pareto optimal solutions to the current PSO-GCANN model. The detail of MOPSO-based optimization approach can be referred in [27].

Step 3. Discretize the objective space into archives and then represent the current Pareto optimal front with a collection of archives. Perform a confirmation experiment to verify the effectiveness of Pareto optimal process parameter settings based on the PSO-GCAN N model predictions.

Step 4. Select an optimization solution from the Pareto optimal set according to fuzzy sets theory for setting up the process parameters.

4 Illustrative example

4.1 Description of optimization problem

In this example, the proposed intelligent approach will be used to perform the multiobjective process parameter optimization for injection molding of a thin-walled part shown in Fig. 3. The plane of the part has a thickness of 3 mm, and the sides have the length of 80 mm. Melt temperature, mold temperature, injection pressure, injection time, holding pressure, holding time, and cooling time are selected as process parameters. Moreover, each process parameter has four levels. Table 1 shows the seven process parameters and their level setting values. In this illustrative case, there are three concerns regarding part quality: (1) part weight, which should be kept as light as possible in order to decrease manufacturing cost; (2) flash, which is a critical quality characteristic to be minimized to keep product quality; and (3) volumetric shrinkage, which should be minimized to improve molded part quality. Three outputs from experimental results (part weight, flash, and volumetric shrinkage) are selected as the objective values to represent the above criteria, respectively. Thus, the multiobjective optimization problem with the ranges of process parameters to be optimized is defined as follows:

Fig. 3 Thin-walled part molded with injection molding process

$$\begin{cases} \text{find}: & Var = (T_m,\ T_w,\ P_i,\ t_i,\ P_h,\ t_h,\ t_c) \\ \text{Minimize}: & \text{part weight} \\ \text{Minimize}: & \text{flash} \\ \text{Minimize}: & \text{volumetric shrinkage} \\ \text{Subject to}: & 180^\circ\text{C} \le T_m \le 210^\circ\text{C};\ 35^\circ\text{C} \le T_w \le 80^\circ\text{C} \\ & 30 \le P_i \le 60;\quad 2\text{s} \le t_i \le 5\text{s};\ 30 \le P_h \le 60; \\ & 2\text{s} \le t_h \le 5\text{s};\quad 5\text{s} \le t_c \le 20\text{s} \end{cases} \tag{8}$$

where seven independent process parameters, namely, melt temperature T_m, mold temperature T_w, injection pressure P_i, injection time t_i, holding pressure P_h, holding time t_h, and cooling time t_c are optimized to achieve the desired objectives.

4.2 Experimental data

The polymer material used for molding the thin-walled part is polypropylene (trademark T30S, density 0.955 g/cm^3, MFI= 3.2 g/10 min), which is provided by Wuhan Phoenix Co. Ltd., China. Experimental data are collected from a hydraulic plastic injection molding machine. The part weight is measured by an electric balance which has a precision of 0.01 g. The flash is measured by an electric caliper which has a precision of 0.01 mm. Furthermore, the shrinkage can be calculated as follow:

$$S = \{(D - M)/D\} \times 100\% \tag{9}$$

where S is shrinkage, D is the cavity dimension of mold, and M is the dimension of plastic part. The cavity dimension of mold and the dimension of plastic part are measured by the above electric caliper.

In order to establish the PSO-GCANN model, according to the seven selected process parameters and their level setting values which are shown in Table 1, an $L_{32}(4^9)$ orthogonal array is selected for arranging the factors and carrying out the experiment. In this experiment, there are 32 treatments with different level combinations of the seven factors and five replications are taken to increase the sensitivity of the statistical analysis. Therefore, 160 sample data are collected. During the collection of samples, first, it takes time for the injection molding machine and mold base to reach a steady state. Second, ten shots of each treatment are conducted to

Table 1 Process parameters and settings of the various levels

	Melt temperature (°C)	Mold temperature (°C)	Injection pressure (%)	Injection time (s)	Holding pressure (%)	Holding time (s)	Cooling time (s)
Level 1	180	35	30	2	30	2	5
Level 2	190	50	40	3	40	3	10
Level 3	200	65	50	4	50	4	15
Level 4	210	80	60	5	60	5	20

ensure that the plastic injection molding is stable, and the ten plastic parts obtained in ten shots of each treatment are discarded. Finally, the official samples used for training back-propagation neural network are collected.

4.3 Optimal structure of BPNN based on PSO and GCA

BPNN training is a high-dimensional optimization problem with many local minima. The mostly used training algorithm is the back-propagation (BP) algorithm. However, there are inherent defects in BP. First, the BP is easily trapped in local minima especially for those nonlinearly separable pattern classification problems or complex function approximation problem [29, 30]. Second, the convergence speed of the BP algorithm is too slow in complex problems. Recently, evolution algorithm is used to train BPNN because the evolution algorithm can improve training performance, and many efficient results are derived. However, neural network's primary target is to ensure network generalization ability, and the generalization ability of neural network depends on network structure (network structure is mainly shown by the number of hidden layers, the number of hidden neurons, and the function characteristics of hidden neurons) and characteristics of training samples. Especially, the determination of the number of hidden neurons is a difficult problem in the study of BPNN.

In this section, BPNN is optimized to obtain PSO-GCANN model by applying GCA to determine the number of hidden neurons and employing PSO to train BPNN. Since better generalization ability and prediction performance are achieved, PSO-GCANN model can finely map the relationship between the input process parameters and output multiresponse. For training BPNN, the objective is to minimize the MSE over all training patterns. The variables consist of BPNN connection weights and threshold values. Suppose a 3-layer BPNN structure with 7 input neurons, 13 hidden neurons, and 3 output neurons for PIM quality indicator as shown in Eq.(8), a 7-13-3 structure of BPNN is constructed according to Sect. 2.1.

The BPNN is first trained using randomly selected 120 samples. By applying PSO plus GCA, the final optimized 7-9-3 structure of BPNN, named as PSO-GCANN model, is obtained. Then the rest 40 samples of verifying data are used to make predictions. The network performance is obtained by calculating the MSE. In order to verify the PSO-GCANN model performance, the experiment of PSO-GCANN is compared with those of PSONN model and BPNN model which are trained based on BPNN by PSO and BP algorithms, respectively. For each algorithm, the results of all experiments are averaged over 50 independent runs to eliminate random discrepancy. The network

Table 2 Comparison of the training and checking results for three model quality predictors

Model	Item	Relative error for training samples			Relative error for checking samples		
		Largest	Smallest	Average	Largest	Smallest	Average
PSO-GCANN	Part weight	0.943	0.116	0.627	1.026	0.137	0.774
	Flash	2.169	0.187	1.042	2.738	0.226	1.68
	Volumetric shrinkage	2.573	0.429	1.843	2.851	0.656	1.942
PSONN	Part weight	1.231	0.185	0.835	1.458	0.194	0.988
	Flash	2.845	0.235	1.127	2.937	0.391	1.913
	Volumetric shrinkage	3.326	0.573	2.563	3.252	0.914	2.853
BPNN	Part weight	1.312	0.192	0.938	1.631	0.348	1.052
	Flash	3.032	0.181	1.421	3.467	0.429	2.47
	Volumetric shrinkage	3.724	0.577	2.951	4.571	0.836	3.417

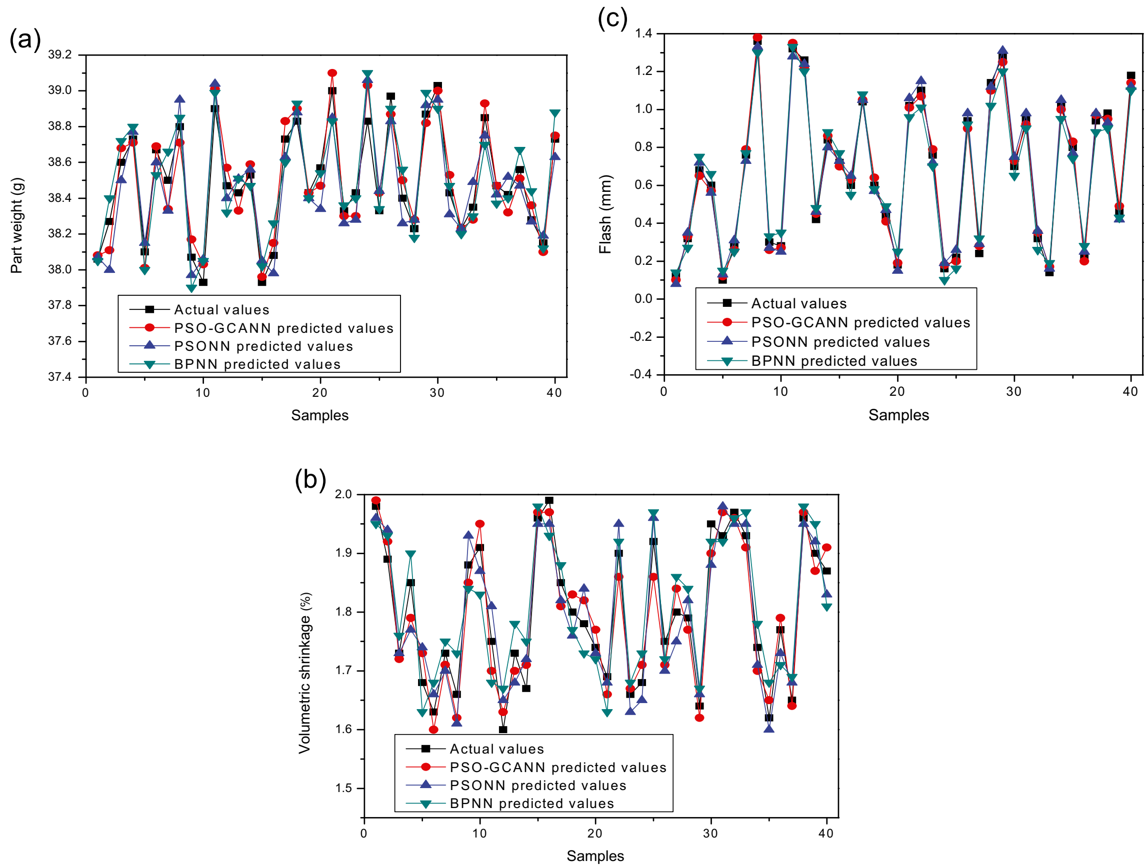

Fig. 4 Comparison between the predicted and actual values of **a** part weight, **b** volumetric shrinkage, and **c** flash via three models

performances of PSO-GCANN, PSONN, and BPNN are shown in Table 2. The comparison results between the experimental values and the predicted values of PSO-GCANN, PSONN, and BPNN under part weight, flash, and volumetric shrinkage are presented in Fig. 4.

From Table 2 and Fig. 4, it can be seen that the extrapolation of the three models all are good and the three models all have high prediction accuracy, but the expansible error of network trained by PSO plus GCA is the smallest, and the expansible error of network trained by BP is larger than that trained by PSO. It can be concluded that the prediction of the PSO-GCAN N agrees well with the data from the experiments. Obviously, the performance of PSO-GCANN is better than those of PSONN and BPNN, respectively, which suggests that the PSO-GCANN can indeed benefit from the optimization of BPNN structure. The PSO-GCANN provides better generalization ability and prediction performance, so that PSO-GCANN can finely map the relationship between the input process parameters and output multiresponse. Thus, it confirms that the prediction ability of PSO-GCANN model further optimized by MOPSO is adequate enough to achieve the Pareto optimal solutions for this application.

4.4 Optimization results and discussion

To use MOPSO to evaluate the PSO-GCANN model to obtain the current Pareto optimal solutions, the parameters for

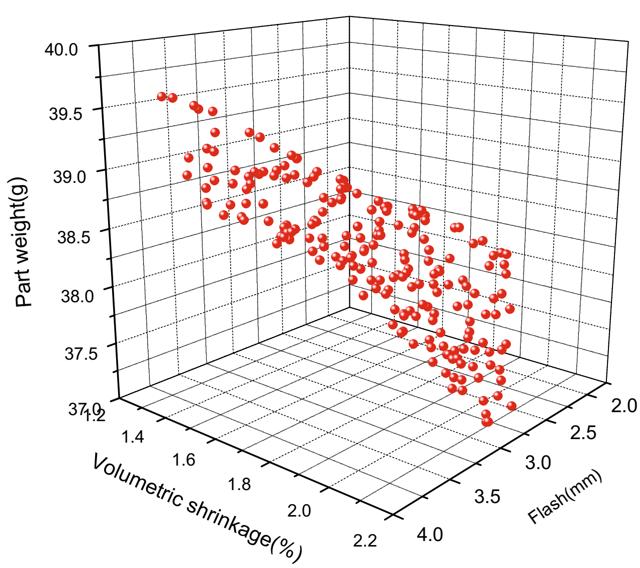

Fig. 5 Pareto optimal front achieved by MOPSO for PSO-GCANN model

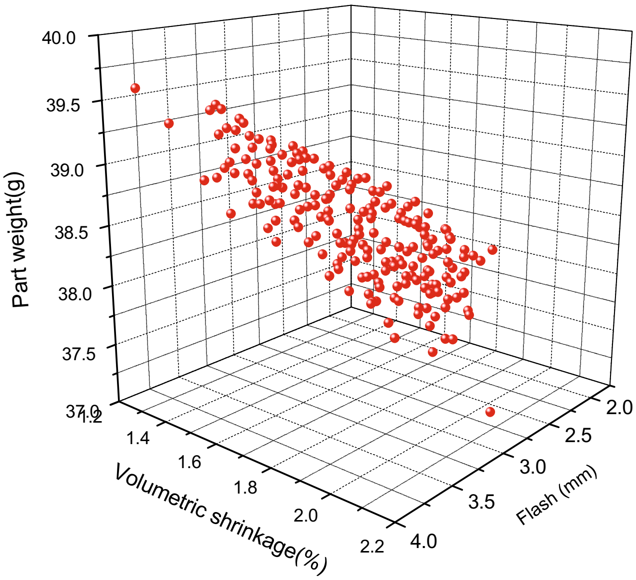

Fig. 6 Pareto optimal front achieved by MOGA for GP surrogate model

MOPSO are selected as suggested in [27]. The number of nondominated solutions to be get is set to 200. After establishing the finalized PSO-GCANN model, the MOPSO is employed in conjunction with the PSO-GCANN model to search for the current Pareto optimal solutions. The three-dimensional Pareto front for Eq. (8) is displayed in Fig. 5.

The predicted responses are nonlinearly correlated to the process parameters, which features complex interactions. From Fig. 5, on the basis of the achieved PSO-GCANN model, it shows that in this specific application there exists direct trade-off behaviors among the part weight, flash, and volumetric shrinkage as expected, in which the volumetric shrinkage decreases with an increase in the part weight and flash, and vice versa. After employing MOPSO to achieve the Pareto optimal solutions for PSO-GCANN model, different combinations of optimal trade-off solutions can be selected from these solutions according to the designer's preference for setting up the process parameters. In order to achieve rationality and practicality, the designer can also select an optimization solution from the Pareto optimal solutions according to fuzzy sets theory for setting up the process parameters.

In order to evaluate the effectiveness and efficiency of the proposed approach, experiments are conducted to compare the performance of Gaussian process approach [17] with the performance of our proposed approach. The optimization result for the same application with Gaussian process approach is shown in Fig. 6. From Figs. 5 and 6, it can be seen that employing MOPSO on the PSO-GCANN model can get better Pareto front with good distribution than the results of employing multiobjective genetic algorithm (MOGA) on the GP surrogate model [17]. To verify the accuracy of this intelligent optimization approach, four Pareto optimal

Table 3 Comparison between PSO-GCANN model predictions and GP surrogate model predictions with experimental results under the corresponding process parameters

		PSO-GCANN model				GP surrogate model			
		1	2	3	4	1	2	3	4
Optimal process parameters	Melt temperature	199.9	187.8	189.9	197.3	195.7	201.2	188.4	184.5
	Mold temperature	36.9	44.85	38.2	37.6	46.99	49.1	51.1	38.4
	Injection pressure	39.9	50.99	52.05	45.42	49.7	56.97	53.56	40.2
	Injection time	4.5	5	4.8	3.67	4.5	3.9	4.9	4.8
	Holding pressure	54.59	46.1	51.27	38.1	50.58	48.86	48.24	39.9
	Holding time	4.4	5	4.7	3.4	4.8	4.4	4.6	2.9
	Cooling time	20	15.9	15.6	6	18.8	16.8	12.5	6.7
Part weight (g)	Model's prediction	39.2	38.5	38.38	37.55	39.05	38.52	38.33	37.82
	Experimental results	38.53	39.38	39.01	36.92	37.96	37.55	39.4	36.95
	Difference (%)	1.74	−2.24	−1.62	1.71	2.87	2.58	−2.71	2.35
Flash (mm)	Model's prediction	1.12	0.98	0.55	0.28	1.48	0.85	0.62	0.23
	Experimental results	1.08	1.02	0.54	0.27	1.39	0.91	0.67	0.21
	Difference (%)	2.78	−3.92	1.85	3.71	6.47	−6.59	−7.46	9.52
Volumetric shrinkage (%)	Model's prediction	1.34	1.42	1.47	1.89	1.34	1.45	1.46	1.93
	Experimental results	1.31	1.46	1.51	1.81	1.26	1.36	1.59	1.82
	Difference (%)	2.29	−2.74	−2.65	3.31	6.35	7.35.	−8.18	6.04
	Average difference (%)	Volumetric shrinkage=2.75; Part weight=1.83; Flash=3.07				Volumetric shrinkage=6.86; Part weight=2.63; Flash=7.51			

Table 4 The final optimal process parameter settings according to Pareto optimizing based on fuzzy sets theory

	Melt temperature	Mold temperature	Injection pressure	Injection time	Holding pressure	Holding time	Cooling time
MOPSO search result	191.03	42.95	48.12	4.05	48.21	3.96	16.16
After tuning	191	43	48	4	48	4	16

solutions based on the PSO-GCANN model and GP surrogate model predictions are selected randomly, respectively. This research conducts conformation experiments with the above seven selected corresponding optimal process parameters. The replication of above conformation experiments is 30 and the results of conformation experiments are averaged for eliminating random disturbance. The comparison between confirmation experimental results and predicted results of the two models is listed in Table 3. Difference percentages between confirmation experimental results and predication results of part weight, flash, and volumetric shrinkage are denoted as difference (%). From Table 3, the average difference percentages of part weight, flash, and volumetric shrinkage are 2.63, 7.51, and 6.86 % for GP surrogate model, respectively; the average difference percentages of part weight, flash, and volumetric shrinkage are 1.83, 3.07, and 2.75 % for PSO-GCAN N model, respectively. It is clear that the average difference percentages of PSO-GCANN model are much smaller than those of GP surrogate model, respectively. Especially, the maximum difference percentage of GP surrogate model under flash is 9.52 %. However, the maximum difference percentage of PSO-GCANN model under flash is only 3.92 %. It can be concluded that the PSO-GCANN model has superior predictive ability than GP surrogate model, and the proposed optimization scheme is effective. It can be explained that PSO-GCANN model realizes the optimization of BPNN structure, which may be attributed to optimization of the number of hidden neurons of BPNN by PSO plus GCA algorithm. Hence, PSO-GCANN model provides better generalization ability and prediction performance than GP surrogate model or other surrogate models. Meanwhile, MOPSO has better global search ability and can be in conjunction with the PSO-GCANN model to achieve Pareto front with good distribution than the results of MOGA or other multiobjective optimization algorithms.

The current approach executes fewer experiments for objective function evaluations and achieves better solutions. Therefore, with the help of this multiobjective optimization approach based on PSO-GCANN model, the optimization task specified in this application can yield reasonable results with a reasonably small amount of computing resources. Although the procedure needs a relatively long time to execute experiments for obtaining the initial training data, the subsequent optimization process could realize lots of benefits from the trained PSO-GCANN model.

After executing a soft computing model, the Pareto optimal solutions are determined. According to Pareto optimizing based on the fuzzy sets theory shown in Sect. 2.4.3, the final optimal process parameter settings from Pareto optimal set are determined after the minimum unit tuning and are shown in Table 4. To demonstrate the effectiveness of the proposed approach, the experimental results are also analyzed using conventional Taguchi method. This research conducts an extra confirmation experiment that has process parameter settings determined by the Taguchi's parameter design method under volumetric shrinkage and flash response consideration. Volumetric shrinkage and flash are used as responses, because weight does not have a target value. The research follows the two-stage approach of Taguchi's parameter design method to determine the optimal process parameter settings under volumetric shrinkage and flash responses consideration.

Two statistics, standard deviation and mean absolute deviation ($\mathrm{MAD} = \frac{1}{n}\sum_{i=1}^{n}|P_i - \mathrm{TV}|$, where is P_i the specific response value of ith confirmation sample, TV is the target value of the specific response, and n is the number of confirmation samples), are compared to show the effectiveness of the proposed approach. To calculate the MAD of weight response, it assumes that 38.51 (the weight average of 160 Taguchi experimental data) is the weight's target just for comparison implementation purpose. The comparison results are showed

Table 5 Response performance comparison under different approaches

	Proposed approach			Taguchi method		
	Part weight	Volume shrinkage	Flash	Part weight	Volume shrinkage	Flash
Average	38.122	1.415	0.202	38.563	1.732	0.421
Standard deviation	0.0126	0.0053	0.0034	0.0181	0.0114	0.0063
MAD	0.0483	0.0071	0.0024	0.0672	0.0122	0.0046

in Table 5. The comparison results reveal that the improvement rates of standard deviation under part weight, flash, and volume shrinkage are 30, 46, and 53 %, respectively, when using the proposed approach. Moreover, the improvement rates of MAD under part weight, flash, and volume shrinkage are 28, 48, and 41 %, respectively, when using proposed approach.

5 Conclusions

Determination of optimal process parameter settings in PIM is complex work that influences product quality. Engineers have conventionally used trial-and-error processes or Taguchi's process parameter design method to determine the optimal process parameter settings. However, the application of these methods has some shortcomings and may cause engineers to make undesirable optimal process parameter settings. In this study, an intelligent approach based on soft computing and grey correlation analysis is put forward for efficiently optimizing MIMO plastic injection molding parameters when multiple objectives are involved. The proposed approach integrates Taguchi's parameter design method, BPNN, GCA, PSO, and MOPSO to locate the Pareto optimal solutions for the multiobjective optimization problem. According to the implementation results obtained in the illustrative example, the Pareto optimal solutions determined by the proposed approach definitely produce better performance compared with the methods shown in previous literature. Moreover, the final optimal process parameter settings from Pareto optimal set are determined by fuzzy sets theory definitely produce better performance in the PIM production process than those of Taguchi's approach. Therefore, the proposed methodology is feasible and effective for process parameter optimization in MIMO plastic injection molding and can assist the manufacturing industry in achieving competitive advantages on quality and costs.

Acknowledgments The authors wish to acknowledge the National Nature Science Foundation of China (Grant 61175127, 11101204), the Nature Science Foundation of Jiangxi Province China (Grant 20142BAB211021), and the Science and Technology Plan Project of Jiangxi Provincial Education Department China (Grant GJJ12093) for the financial support.

References

1. Chen WC, Liou PH, Chou SC (2014) An integrated parameter optimization system for MIMO plastic injection molding using soft computing. Int J Adv Manuf Technol 73(9–12):1465–1474
2. Öktem H (2012) Optimum process conditions on shrinkage of an injected-molded part of DVD-ROM cover using Taguchi robust method. Int J Adv Manuf Technol 61(5–8):518–528
3. Chen CC, Su PL, Lin YC (2009) Analysis and modeling of effective parameters for dimension shrinkage variation of injection molded part with thin shell feature using response surface methodology. Int J Adv Manuf Technol 45(11–12):1087–1095
4. Yin F, Mao H, Hua L, Guo W, Shu M (2011) Back-propagation neural network modeling for warpage prediction and optimization of plastic products during injection molding. Mater Des 32:1844–1850
5. Gao YH, Wang XC (2009) Surrogate-based process optimization for reducing warpage in injection molding. J Mater Process Tech 209(3):1302–1309
6. Kitayama S, Onuki R, Yamazaki K (2014) Warpage reduction with variable pressure profile in plastic injection molding via sequential approximate optimization. Int J Adv Manuf Techno 72(5–8):827–838
7. Zhao P, Zhou H, Li Y, Li D (2010) Process parameters optimization of injection molding using a fast strip analysis as a surrogate model. Int J Adv Manuf Technol 49:949–959
8. Li XL, Hu B, Du RX (2008) Predicting the parts weight in plastic injection molding using least squares support vector regression. IEEE T Syst Man Cy C 38(6):827–833
9. Wei XW, Luo B, Liao XP (2011) An enhanced optimization approach based on Gaussian process surrogate model for process control in injection molding. Int J Adv Manuf Technol 56(9–12):929–942
10. Azlan MZ, Habibollah H, Safian S (2012) Integrated ANN-GA for estimating the minimum value for machining performance. Int J Prod Res 50(1):191–213
11. Chen WC, Tai PH, Wang MW, Deng WJ, Chen CT (2008) A neural network-based approach for dynamic quality prediction in a plastic injection molding process. Expert Syst Appl 35(3):843–849
12. Shen CY, Wang LX, Li Q (2007) Optimization of injection molding process parameters using combination of artificial neural network and genetic algorithm method. J Mater Process Tech 183(23):412–418
13. Guo W, Hua L, Mao HJ (2014) Minimization of sink mark depth in injection-molded thermoplastic through design of experiments and genetic algorithm. Int J Adv Manuf Technol 72(1–4):365–375
14. Castro CE, Rios MC, Lilly B, Castro JM (2005) Simultaneous optimization of mold design and processing conditions in injection molding. J Polym Eng 25(6):459–485
15. Huang CC, Tang TT (2006) Parameter optimization in melt spinning by neural networks and genetic algorithms. Int J Adv Manuf Technol 27(11):1113–1118
16. Chen WC, Fu GL, Tai PH, Deng WJ (2009) Process parameter optimization for MIMO plastic injection molding via soft computing. Expert Syst Appl 36(2):1114–1122
17. Zhou J, Turng LS (2007) Adaptive multiobjective optimization of process conditions for injection molding using a Gaussian process approach. Adv Ploym Tech 26(2):71–85
18. Wei Z, Feng YX, Tan JR, Wang JL, Li ZK (2009) Multi-objective performance optimal design of large-scale injection molding machine. Int J Adv Manuf Technol 41(3/4):242–249
19. Wei Z, Yang DD, Wang XY, Wang JL (2009) Multi-objectives optimal model of heavy equipment using improved Strength Pareto Evolutionary Algorithm. Int J Adv Manuf Technol 45(3/4):389–396
20. Solimanpur M, Ranjdoostfard F (2009) Optimisation of cutting parameters using a multi-objective genetic algorithm. Int J Prod Res 47(21):6019–6036
21. Cheng J, Liu ZY, Tan JG (2013) Multiobjective optimization of injection molding parameters based on soft computing and variable complexity method. Int J Adv Manuf Technol 66(5–8):907–916
22. Luo YB, Xiao TY, Chen DF (2003) Key technologies for sharing NC machine tool based on multi-agent. Comput-integr Manuf Syst 9(9):750–754
23. Woll SLB, Cooper DJ (1997) Pattern-based closed-loop quality control for the injection molding process. Polym Eng Sci 37(5):801–812
24. Zhang YJ, Zhang X (2007) Grey correlation analysis between strength of slag cement and particle fractions of slag powder. Cement Concrete Comp 29(6):498–504

25. Shi YH, Eberhart RC (1998) A modified particle swarm optimizer. In: Proceedings of the IEEE international conference on evolutionary computation, pp. 69–73
26. Shi YH, Eberhart RC (1999) Empirical study of particle swarm optimization, in: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 1945–1950.
27. Carlos A, Coello C, Gregorio TP, Maximino SL (2004) Handling multiple objectives with particle swarm optimization. IEEE T Evolut Comput 8(3):256–279
28. Zhang JR, Zhang J, Lok TM, Lyu MR (2007) A hybrid particle swarm optimization-back-propagation algorithm for feedforward neural network training. Appl Math Comput 185(2):1026–1037
29. Hsu CM (2004) Improving the electroforming process in optical recordable media manufacturing via integrated procedure. Eng Optimiz 36(6):659–675
30. Su CT, Chang HH (2000) Optimization of parameter design: an intelligent approach using neural network and simulated annealing. Int J Sys Sci 31(12):1543–1549