A proposed iteration optimization approach integrating backpropagation neural network with genetic algorithm

Han-Xiong Huang *, Jiong-Cheng Li, Cheng-Long Xiao

Lab for Micro Molding and Polymer Rheology, The Key Laboratory of Polymer Processing Engineering of the Ministry of Education, South China University of Technology, Guangzhou 510640, PR China

ARTICLE INFO

Article history:
Available online 4 August 2014

Keywords:
Iteration optimization
Backpropagation neural network
Genetic algorithm
Blow molding

ABSTRACT

An iteration optimization approach integrating backpropagation neural network (BPNN) with genetic algorithm (GA) is proposed. The main idea of the approach is that a BPNN model is first developed and trained using fewer learning samples, then the trained BPNN model is solved using GA in the feasible region to search the model optimum. The result of verification conducted based on this optimum is added as a new sample into the training pattern set to retrain the BPNN model. Four strategies are proposed in the approach to deal with the possible deficiency of prediction accuracy due to fewer training patterns used. Specifically, in training the BPNN model, the Bayesian regularization and modified Levenberg–Marquardt algorithms are applied to improve its generalization ability and convergence, respectively; elitist strategy is adopted and simulated annealing algorithm is embedded into the GA to improve its local searching ability. The proposed approach is then applied to optimize the thickness of blow molded polypropylene bellows used in cars. The results show that the optimal die gap profile can be obtained after three iterations. The thicknesses at nine teeth peaks of the bellow molded using the optimal gap profile fall into the desired range (0.7 ± 0.05 mm) and the usage of materials is reduced by 22%. More importantly, this optimal gap profile is obtained via only 23 times of experiments, which is far fewer than that needed in practical molding process. So the effectiveness of the proposed approach is demonstrated.

http://dx.doi.org/10.1016/j.eswa.2014.07.039
0957-4174/© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

Many engineering problems, such as system design, process control and prediction, and part manufacturing, are related to optimization. The purposes of the engineering optimizations can be generally summarized as follows: enhancing the system performance (Li & Yang, 2008; Wang, Zhao, Li, & Guan, 2011), increasing the process control and prediction precisions (Chang & Shih, 2010; Wang, Dong, & Sun, 2010), improving the product quality (Liu & Yang, 2008; Raja & Baskar, 2012), saving the cost (Lee & Lin, 2009; Wang, Wang, & Wang, 2013), etc. In engineering optimizations, a mathematical model is first developed for representing the quantitative relationship between the outputs and inputs of the investigated system or process, and then is solved in feasible region using an optimization algorithm to obtain the optimal process parameters.

However, precise explicit functions mapping the outputs and inputs of a system in engineering problems are often complex and nonlinear, and so quite difficult or even impossible to be deduced from some physical laws. For this reason, some approximation-based process modeling methods, including response surface methodology, radial basis function, Kriging model, and neural network (NN), are usually used to approximate the explicit functions in many engineering applications (Elsayed & Lacor, 2012; Gao & Wang, 2008; Huang, Li, Li, & Huang, 2011; Huang & Lu, 2005; Mirmohseni & Zavareh, 2011). Among these methods, NN is a synergistic representation of mathematical methods helpful in the modeling of nonlinear multivariate systems. The feature of the NN is its ability to capture complex nonlinear relationships between output and input patterns through appropriate learning. Among NN approaches, backpropagation neural network (BPNN) is the most classically and generally used training algorithm, and can provide effective solutions to industrial applications. BPNN is multilayer feed-forward neural network that is trained by the error BP algorithms. Although the BPNN is successful, it has some disadvantages. The algorithm is not guaranteed to find a global optimum and the convergence rate tends to be extremely low. In addition, the selection of the learning factor and inertial factor, which is usually determined by experience, affects its convergence. Genetic algorithm (GA) is a heuristic and stochastic optimization algorithm based on evolution theory and genetic principles. It is an aggressive...
Various investigations have demonstrated that combining BPNN and GA is a helpful methodology to obtain desirable solutions to optimization problems (Ahmad, Jeenanunta, Chanvarasuth, & Komolavanij, 2014; Chatterjee & Bandopadhyay, 2012; Chen, Lai, Wang, & Hung, 2011; Chen et al., 2014; Cho, Moon, Kim, & Yun, 2012; Cook, Ragsdale, & Major, 2000; Dehghani, Sefri, Ameri, & Kaveh, 2008; Esmaeili & Dashtbayazi, 2014; Gossard, Larigue, & Thellier, 2013; Ho & Chang, 2011; Huang & Huang, 2007; Irani & Nasimi, 2011; Kim & Han, 2003; Ko et al., 2009; Krishna, Rangajanardhaa, Hanumantha, & Sreenivasas, 2009; Mirarab, Sharifi, Ghayyem, & Mirarab, 2014; Nasser, Ashghari, & Abedini, 2008; Singh, Cooper, Blundell, Pratihar, & Gibbons, 2014; Sinha, Sikdar (Dey), Chatterpadhyay, & Datta, 2013; Solenimani, Shoushtari, Mirza, & Salahi, 2013; Su, Yang, & Huang, 2011; Versace, Bhatt, Hinds, & Shiffer, 2004; Wang et al., 2010; Yuen, Wong, Qian, Chan, & Fung, 2009). For example, Kim and Han (2003) proposed a hybrid model composed of BPNN and GA, in which the GA globally searches and seeks an optimal or near-optimal BPNN topology. Huang and Huang (2007) proposed a hybrid method consisting of finite element method, BPNN and GA to optimize the parison thickness distribution for an extrusion blow molded plastics part with required thickness distribution. The results showed that the proposed method can be used to effectively obtain the optimal parison thickness distribution. Dehghani et al. (2008) used GA to optimize the connection weights, network architecture and learning rules of BPNN model. Ko et al. (2009) investigated the process modeling for the growth rate in pulsed laser deposition-grown ZnO thin films using BPNN and GA. The results showed that this modeling methodology can explain the characteristics of the thin film growth mechanism varying with process conditions. Wang et al. (2010) coupled a BPNN with a GA to predict the saturates of sour vacuum gas oil. The study showed that the GA can find the optimal architecture of the NN and the parameters of the BP algorithm. Ho and Chang (2011) used a GA in the BPNN to find the optimal parameters to investigate the promoted effectiveness of predicting platelet transfusion requirements for acute myeloblastic leukemia patients. Irani and Nasimi (2011) presented a GA evolved BPNN, which can improve the reliability and predictability of BPNN. Each initial weight of the gradient descent-based BPNN was selected by a standard GA and the fitness of the GA was determined by the BPNN. The genetic operators and parameters were carefully designed and set, avoiding premature convergence and permutation problems. The methodology combines the local searching ability of the gradient decent method with the global searching ability of the GA. Su et al. (2011) found that better initial weight/bias for the NN can be calculated by the GA. Chatterjee and Bandopadhyay (2012) used a GA for the selection of BPNN parameters to forecast the reliability of a load–haul–dump machine. In the work of Esmaeili and Dashabayazi (2014), a BPNN model was used for predicting the characteristics of the prepared Al/SiC nanocomposite, and then GA was applied to optimize the process parameters. The results showed that the combination of the BPNN and GA would make good on appropriate use of data for predicting and optimizing preferred parameters in materials processing technology. Singh et al. (2014) applied a GA to enhance the prediction accuracy of BPNN by altering its topology. Mirarab et al. (2014) performed an optimization procedure based on GA to select the best BPNN architecture and determine the optimum neuron numbers in the hidden layer of the BPNN.

As a combined prediction-optimization approach, the hybrid BPNN–GA model processes excellent performance since it combines the inherent merits of BPNN (i.e. accurate nonlinear data fitting or regression capabilities) and GA (i.e. efficient and parallel global searching ability). From the foregoing, the optimization strategy coupling BPNN with GA is a very effective approach to solve engineering optimization problems. Moreover, following two main aspects are covered in applying GA into BPNN. One is to optimize the topological structure of the network, and the other is to optimize some parameters of the network. However, there are

---

**Fig. 1.** Flow chart of proposed iteration optimization approach integrating BPNN with GA in this work.
still some drawbacks of this combined approach needed to be dealt with. For example, a great number of learning samples acquired by costly experiments or time-consuming numerical simulations should be provided as the training patterns to train the BPNN model with accurate generality. When trained using the standard training algorithm such as standard Levenberg–Marquardt algorithm, the convergence of the BPNN is relatively slow, especially for the networks with large scale. Moreover, the standard GA tends to be trapped into a local optimum rather than a global one in the search process. To address these problems encountered in engineering optimizations, a novel iteration optimization approach combining BPNN with GA is proposed in this work, in which the standard Levenberg–Marquardt algorithm and GA are modified and improved. By using the proposed approach, the optimization results can be easily and quickly obtained without carrying out too many experiments or numerical simulations. The effectiveness of the approach is confirmed by its application in the wall thickness optimization for a complex-shaped industrial plastics part.

2. Iteration optimization approach integrating backpropagation neural network with genetic algorithm

Fig. 1 shows the flow chart of the proposed iteration optimization approach integrating BPNN with GA. The main idea of the proposed optimization approach is described as follows. A BPNN model is first established and trained using fewer learning samples, which can be acquired by experiments or numerical simulations according to the design of experiment (DOE), such as orthogonal design method, Latin hypercube sampling technique, and Box–Behnken or center composite experimental designs. Thereafter, using the trained BPNN model as the fitness function of the GA, the model is solved in the feasible region to search the model optimum. The verification experiment or simulation based on the derived optimum is conducted. Then the optimum and corresponding experimental or simulated results are added into the training pattern set as a new learning sample to retrain the BPNN model. Therefore, only one additional experiment or simulation is required in each iteration. The training and searching processes repeat until the optimization process is converged.

To deal with the possible deficiency of prediction accuracy due to fewer training patterns used in the proposed approach, the following four strategies are adopted in this work. (i) The Bayesian regularization algorithm (MacKay, 1992, 1995) is used during the training of the BPNN model to improve its generalization ability. (ii) Standard Levenberg–Marquardt training algorithm is modified to accelerate the BPNN convergence. (iii) Simulated annealing algorithm (SAA) is embedded into the GA to enhance its local searching ability, which facilitates to effectively improve the quality of solutions. (iv) For each iteration, the currently obtained best solution is always selected as an individual in the initial population of the GA; moreover, an elitist strategy is applied, in which the elite individual (namely the current best individual during evolution) is directly copied to the population of the next generation without performing any genetic operations. By adopting the four strategies, the solution obtained using the proposed iteration optimization approach is better and better with the iteration. In addition, the gradual increase of the learning samples number in the training pattern set can also improve the generalization ability and prediction accuracy of the BPNN model and thus facilitate to improve the

<table>
<thead>
<tr>
<th>Exp. No.</th>
<th>$O_1$</th>
<th>$O_2$</th>
<th>$O_3$</th>
<th>$O_4$</th>
<th>$O_5$</th>
<th>$O_6$</th>
<th>$O_7$</th>
<th>$y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>0.0144</td>
</tr>
<tr>
<td>2</td>
<td>65</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>0.0120</td>
</tr>
<tr>
<td>3</td>
<td>65</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>0.1455</td>
</tr>
<tr>
<td>4</td>
<td>80</td>
<td>65</td>
<td>65</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>0.0355</td>
</tr>
<tr>
<td>5</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>0.0988</td>
</tr>
<tr>
<td>6</td>
<td>80</td>
<td>95</td>
<td>95</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>0.0135</td>
</tr>
<tr>
<td>7</td>
<td>95</td>
<td>65</td>
<td>65</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>0.0322</td>
</tr>
<tr>
<td>8</td>
<td>95</td>
<td>80</td>
<td>95</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>0.0307</td>
</tr>
<tr>
<td>9</td>
<td>95</td>
<td>95</td>
<td>65</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>0.1631</td>
</tr>
<tr>
<td>10</td>
<td>65</td>
<td>65</td>
<td>95</td>
<td>95</td>
<td>80</td>
<td>80</td>
<td>80</td>
<td>0.2451</td>
</tr>
<tr>
<td>11</td>
<td>65</td>
<td>80</td>
<td>65</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>95</td>
<td>0.0787</td>
</tr>
<tr>
<td>12</td>
<td>65</td>
<td>95</td>
<td>80</td>
<td>80</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>0.0397</td>
</tr>
<tr>
<td>13</td>
<td>80</td>
<td>65</td>
<td>80</td>
<td>95</td>
<td>65</td>
<td>65</td>
<td>95</td>
<td>0.0082</td>
</tr>
<tr>
<td>14</td>
<td>80</td>
<td>80</td>
<td>95</td>
<td>65</td>
<td>80</td>
<td>80</td>
<td>95</td>
<td>0.0556</td>
</tr>
<tr>
<td>15</td>
<td>80</td>
<td>95</td>
<td>65</td>
<td>80</td>
<td>95</td>
<td>80</td>
<td>80</td>
<td>0.0851</td>
</tr>
<tr>
<td>16</td>
<td>95</td>
<td>65</td>
<td>95</td>
<td>80</td>
<td>95</td>
<td>65</td>
<td>80</td>
<td>0.0992</td>
</tr>
<tr>
<td>17</td>
<td>95</td>
<td>80</td>
<td>65</td>
<td>95</td>
<td>65</td>
<td>80</td>
<td>95</td>
<td>0.0152</td>
</tr>
<tr>
<td>18</td>
<td>95</td>
<td>95</td>
<td>80</td>
<td>65</td>
<td>95</td>
<td>80</td>
<td>95</td>
<td>0.0191</td>
</tr>
</tbody>
</table>

| Experimental design with $L_{18}$ ($3^7$) orthogonal array and objective function values $y$. | | |

![Fig. 2. Schematics of extrusion blow molding process.](image-url)
quality of solutions. It is noteworthy that, to the authors’ best knowledge, no literature reports such hybrid iteration optimization approach. In the following, aforementioned strategies (ii) and (iii) are described in detail.

2.1. Accelerate BPNN convergence using modified Levenberg–Marquardt algorithm

Conventional backpropagation algorithms (e.g., the steepest descent method) are the first-order training algorithm for they require only the first derivatives of the error function. The second-order algorithms (e.g., Levenberg–Marquardt algorithm), in which the second derivatives of the error function need to be estimated, are demonstrated to be more effective for training NN model with a relatively small number of weight and bias (Su et al., 2011). The update rule for network weight values using the Levenberg–Marquardt algorithm is defined as:

$$\Delta w_k = -\eta [H(w_{k-1}) + \lambda I]^{-1} \nabla E(w_{k-1})$$  

(1)

where $\Delta w_k$ is the weight value increment of the neural network in the $k$th update step, $\eta$ is the network learning rate, $H$ is the Hessian matrix, $\lambda$ is a positive constant, and $\nabla E$ is the gradient of the network error function. The calculation of the Hessian matrix inversion at each update step is required in Eq. (1), which is a quite time-consuming process and significantly lowers the network convergence rate. In light of this, a modified Levenberg–Marquardt algorithm is proposed to avoid computing the inverse matrix and thus to increase the network training efficiency in this work. Start with the initial weight given in the first stage and denote the term of $H(w_{k-1}) + \lambda I$ as matrix $G$, Eq. (1) can be rewritten as:

$$\Delta w_k = -\eta G^{-1}(w_{k-1}) \nabla E(w_{k-1})$$  

(2)

Left-multiplication by matrix $G$ on both sides of Eq. (2) results in:

$$G(w_{k-1}) \Delta w_k = -\eta \nabla E(w_{k-1})$$  

(3)

The $G$ is a nonsingular matrix, so it can be decomposed into the multiplication of a lower triangular matrix $L$ and an upper triangular matrix $U$, namely $G = LU$. In this way, Eq. (3) can be solved by readily performing the forward substitution and back substitution to find the $\Delta w_k$. This is a computationally efficient process in comparison to the direct calculation of $G^{-1}$. The latter takes about three times as many arithmetic operations as the former. Moreover, the results obtained using the LU decomposition method may be more accurate.

2.2. Embed simulated annealing algorithm into genetic algorithm

SAA is a heuristic, stochastic, and global optimization algorithm for nonlinear programming problems proposed by simulating the slow cooling process of molten metal (Kirapatrick, Gelatt, &
Vecchi, 1983). Its basic idea is to generate a random point to avoid getting trapped at a local optimum. The random search not only accepts the changes that decrease the fitness function $f$ (make it better), but also the changes that increase the $f$ (make it worse) with a probability $p = \exp(-\Delta/T)$, where $\Delta$ is the increment of the $f$ and $T$ is a control parameter known as system absolute temperature. The detailed implementation of the SAA can be found in Abbasi and Mahlooji (2012).

Considering that the standard GA tends to be trapped in a local optimum rather than global one in the searching process, the SAA is embedded into the standard GA to improve its local searching ability in this work. This integrated algorithm can effectively combine the merits of both algorithms. That is, it possesses not only excellent global searching ability of the GA, but also powerful local searching ability of the SAA. The detailed procedure of the integrated algorithm is described as follows.

Step 1: Initialization. Initialize the population size $N$ of the GA, the initial annealing temperature $T_0$, the final annealing temperature $T_s$, and the number of repetitions $L$ allowed at each temperature level of the SAA.

Step 2: Genetic operations.

Step 2.1: Set iteration number $t = 1$. Randomly generate $N$ individuals in feasible region to form the first population $P_1$,

Fig. 5. Flow chart of optimization process based on proposed approach for case study.
molding (EBM) is one of main polymer manufacturing technologies and has a wide product range from packaging containers to industrial complex parts such as those supplied to the automobile, office automation equipment, and pharmaceutical sectors. Recently, more and more automobile parts are gradually turning to be produced using the EBM technology with the rapid development of automobile industry. This is attributed to its advantages of design flexibility, lower production cost, double-wall hollow part construction, and high part strength-to-weight ratio. Unsuitable process parameter settings in EBM may cause production problems, reduce price competitive advantage, and decrease a company’s profitability. So in this section, the wall thickness optimization of a polypropylene bellow molded using the EBM technology is taken as an example to illustrate the engineering application of the proposed iteration optimization approach.

In the EBM process as schematically shown in Fig. 2, the polymer is melted and mixed in an extruder, and then the molten polymer is extruded through a parison die to form an annular parison, which is located between two mold halves. When the extruded parison reaches the predetermined length, the mold is closed and then the clamped parison is inflated with compressed air to take the shape of the mold cavity, followed by the part cooling and ejection. Strictly controlling the wall thickness distribution of EBM parts is usually required to achieve required mechanical properties of parts and to minimize the usage of materials. The part wall thickness distribution depends mainly on the thickness distribution along the parison just prior to inflation, which can be controlled by manipulating the die gap as a function of time during parison extrusion, known as parison programming. Fig. 3 schematically illustrates the parison thickness adjustment using the die gap programming.

### 3.1. Objection function establishment

Fig. 4 schematically shows the EBM bellow investigated in this work. This bellow used in cars has nine triangular teeth, and the distance between adjacent two peaks of five teeth near the small and big ends is 10 and 15 mm, respectively. The durability of the bellow in applications is mainly dependent on the thicknesses at the teeth peaks, which should not be less than a certain value. Considering the material usage, the peaks should not be too thick. In this work, the thicknesses at the nine teeth peaks are controlled by manipulating the die gap as a function of time during parison extrusion, known as parison programming. Fig. 3 schematically illustrates the parison thickness adjustment using the die gap programming.

3. Application of proposed approach in wall thickness optimization of extrusion blow molded part

Determination of optimal process parameter settings, which affects productivity, quality, production cost and delivery time, is crucial work and regarded as the most challenging part in manufacturing industries. To ascertain the optimal process parameter settings, numerous process trials are generally required to evaluate the process variables and their interactions. Extrusion blow
difficult to be achieved using the conventional strategy based on the experience and intuition of operators. To achieve this goal, it is necessary to optimize the thickness distribution along the parison, and so seven programming points are set to manipulate the die opening in the parison extrusion stage. The die openings at the seven discrete points, denoted as $O_1$, $O_2$, ..., $O_7$, are identified as the design variables.

The peaks of the nine teeth (denoted as 1#, 2#, ..., 9# shown in Fig. 4) of the bellow are selected as the measured positions of wall thicknesses, and the following objective function is developed to evaluate the wall thickness distribution:

$$
\min_y = \frac{\sum_{j=1}^{m} (T_{h_j} - T_{h_{obj}})^2}{m}
$$

where $n = f(O_1, O_2, ..., O_7)$ is the vector of design variables, $T_{h_j}$ is the peak thickness at the $j$th measured position, $T_{h_{obj}}$ denotes the objective wall thickness for the nine teeth peaks, and $m$ is the total number of the measured positions. Obviously, the smaller the value of the objective function $y$, the smaller the deviation between the actual and objective wall thicknesses at the nine teeth peaks of the molded bellows is.

### 3.2. Optimization process

The purpose of the optimization for this case study is to employ the proposed approach to model and acquire the optimal die gaps corresponding to the seven parison sections for obtaining the desired wall thicknesses at the nine teeth peaks in the molded bellows. Fig. 5 illustrates the flow chart of the optimization process in detail. The process includes two steps, that is, the design and analysis of experiment and the iteration optimization for die gap profile.

In the design and analysis of experiment, the orthogonal design method is used to collect the representative samples for training the BPNN model more effectively. The levels of the design variables are determined based on the initial die gap profile with a fixed opening of 80% (denoted as "gap 0"). Three levels (65%, 80%, and 95% die openings) for each design variable are set, so eighteen experimental runs depended on $L_{18}(3^7)$ orthogonal array as shown in Table 1 are performed.

The orthogonal experiment results are analyzed using the sensitivity analysis method, in which the sensitivities of the design variables at each level are calculated. Then, the optimal level of each design variable is obtained according to the analyzed results.
and the corresponding die gap profile is denoted as “gap 1”. The verification experiment with “gap 1” is conducted and the thickness at the nine measured positions on the molded bellow are detected. If the bellow molded using “gap 1” meets the desired thickness requirement, output “gap 1” as the optimal die gap profile for blow molding process; else, totally 20 samples obtained so far by conducting the EBM experiments according to “gap 0”, $L_{18}$ orthogonal array and “gap 1” are used as the training patterns of the BPNN model in the following iteration optimization step.

In the iteration optimization for die gap profile, all the currently obtained samples are used as the training patterns to train the BPNN model. A three-layer (namely, an input layer, a hidden layer and an output layer) BPNN model is developed to build the quantitative relationship between the objective function (Eq. (4)) and die openings. Sigmoid and linear transfer functions are used for the hidden and output layers, respectively. The seven die openings in all the samples are selected as the input parameters of the BPNN model, their corresponding objective function values are the output parameters. Because there is no definite rule to determine the appropriate number of neurons in the hidden layer, it is determined by trail-and-error method in this work. The results showed that the BPNN model with 10 neurons in the hidden layer can give better generalization and effectively avoid the overfitting of neural network. So a BPNN model with 7–10–1 architecture is constructed, as shown in Fig. 6. In training the BPNN process, the aforementioned Bayesian regularization algorithm and modified Levenberg–Marquardt algorithm are applied. Moreover, the tolerance of the mean square error (MSE) is set at 0.0001. Once the MSE of the network is reduced within the given tolerance, the training pattern set to retrain the BPNN model. The training and searching processes repeat until the iteration optimization process is converged.

### 3.3. Results and discussion

The molded bellows are cut open along their axial plane of symmetry. The wall thicknesses at the nine peaks (as shown in Fig. 4) are measured using a digimatic caliper with a resolution of 0.001 mm. Fig. 7 gives the measured thicknesses of the nine peaks of the bellow molded using the initial die gap profile (gap 0). As can be seen, only four peaks exhibit the thicknesses falling into the desired range (0.7 ± 0.05 mm), and the difference between maximum and minimum thicknesses among the nine peaks is about 0.23 mm. For each set of design variables in the orthogonal array (shown in Table 1), a corresponding EBM experiment is conducted. The measured thicknesses are used to estimate the objective function values $y$ shown in Eq. (4), which are listed in the corresponding column in Table 1. The sensitivity analyses on the orthogonal experiment results show that the optimum set of design variable levels is a combination of $O_12$, $O_22$, $O_32$, $O_41$, $O_52$, $O_61$, and $O_71$, which is denoted as gap 1. Gap 1 corresponds to 80% die opening for 1#, 2#, 3#, and 5# programming points, and 65% opening for 4#, 6#, and 7# points. Fig. 8 shows the measured thicknesses at the nine peaks of the bellow molded using gap 1. It can be clearly observed that the uniformity of the thickness distribution is greatly improved, the difference between maximum and minimum thicknesses decreases from initial 0.23 to 0.12 mm. The $y$ value obtained using gap 1 is 0.0029, which is much less than the minimum $y$ value obtained in the orthogonal experiment (0.0082, Table 1). However, still four peaks exhibit the thicknesses out of the desired range. So the thickness optimization procedure goes into the iteration optimization step.

Fig. 9 shows the evolution of the measured wall thicknesses at the nine peaks with iteration in the iteration optimization process. It can be clearly seen that the deviation between the actual and desired thickness distributions is gradually decreased with iteration. After three iterations, the thicknesses at all measured positions fall into the desired range and the average thickness is 0.701 mm, then the iteration optimization process is terminated. Table 2 summarizes the objective function values $y$ and weights for the bellows molded using the aforementioned gap 0 and gap 1 and the die gap profiles obtained in the three iterations (denoted as gap 2, gap 3, gap 4). As can be seen, the $y$ value is very small (0.0006) in the final iteration. Significantly, the bellow weights are reduced from the initial 61.39 g to the optimized 47.98 g, saving about 22% materials. This indicates that the proposed iteration optimization approach in this work is feasible and very effective to obtain the optimal die gap profile for achieving the desired thickness distribution in final parts. More importantly, this optimal gap profile is obtained via only 23 experiment times, which is far fewer than that needed in practical molding process. The corresponding die gap profile (gap 4) is taken as the optimal one input into the controller of the EBM machine for the continuous blow molding process.

### 4. Conclusions

The contributions of the current work include both methodology and application aspects. In the aspect of methodology, a hybrid optimization approach integrating both BPNN and GA is proposed. Four strategies are adopted to overcome the possible deficiency of prediction accuracy due to fewer training patterns used in the approach. Specifically, in training the BPNN model, the Bayesian regularization and modified Levenberg–Marquardt algorithms are applied to improve its generalization ability and convergence, respectively; the elitist strategy is adopted and the SAA is embedded into the GA to improve its local searching ability. The proposed approach can avoid getting trapped at a local optimum. More
it can avoid using a great number of learning samples, which are acquired by expensive experiments or numerical simulations, for training the BPN model.

The hybrid optimization approach proposed in this work shows great potential in complicated industrial applications. The wall thickness optimization of the polypropylene bellows molded using the blow molding technology is taken as an example to illustrate the application of the proposed approach. The trained BPN model is used as the fitness function of the GA for solving the model to search the optimal die gap profile. The results show that the proposed approach is feasible and very effective to find the optimal die gap profile for achieving the desired thickness distribution in final bellows without doing too many experiments. After three iteration optimizations, the thickness at all the nine peaks fall into the desired range (0.7 ± 0.05 mm) and the average thickness is 0.701 mm. Moreover, the usage of materials is reduced by 22%. That is, satisfactory result is obtained via fewer experiment number, which is only 23 and far fewer than that needed in practical molding process for such complex part. The proposed approach can effectively help to determine optimal process parameter settings and achieve competitive advantages on product quality and costs in manufacturing industries.

Work is still needed to be performed in three major aspects. (1) This work focuses on the optimal die gap profile in blow molding. For future work the main concern may be towards integration of more design variables. The proposed approach may be extended to other complicated engineering optimization problems for obtaining optimal solutions via less cost and time spent on experiments and analyses. (2) The trial and error method is used for determining the parameters in the GA, including the population size, crossover rate, and mutation rate. In the future, GA parameters may be considered to improve the predictive performance of the model. (3) Other heuristic optimization algorithms, such as particle swarm optimization, may be used to deal with similar optimization problems, and their performances are compared to that of the GA.

Acknowledgments

Financial support provided by National Natural Science Foundation of China and National High-Tech Research and Development Program is greatly acknowledged.

References


