ELSEVIER

Computers in Industry 50 (2003) 113–122

COMPUTERS IN INDUSTRY

www.elsevier.com/locate/compind

Applying robust multi-response quality engineering for parameter selection using a novel neural–genetic algorithm

T.S. Li[a], C.T. Su[b,*], T.L. Chiang[c]

[a]Department of Industrial Engineering and Management, Ming Hsin University of Science and Technology, Hsinchu, Taiwan 304, ROC
[b]Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu, Taiwan 300, ROC
[c]Department of Business Administration, Ming Hsin University of Science and Technology, Hsinchu, Taiwan 304, ROC

Received 22 September 2000; accepted 2 September 2002

Abstract

This study presents a neural–genetic algorithm to solve the selection problem of manufacturing process parameters. The proposed algorithm is a combination of artificial neural network (ANN) and genetic algorithms (GAs). In addition, the neural network is used to formulate a fitness function for predicting the value of the response based on the parameter settings. GAs then take the fitness function from the trained neural network to search for the optimal parameter combination. Owing to the most of manufactured products have more than one quality characteristic and the quality characteristics are generally correlated with each other, this study also proposes a desirability function to obtain a compromise, composite solution. A case study of how the silicon manufacturing process parameters are selected offline demonstrates the effectiveness of the proposed approach.

Keywords: Artificial neural network; Genetic algorithm; Multi-response optimization; Desirability function; Parameter selection

1. Introduction

A group of responses often characterize the performance of a manufactured product. These responses are generally correlated and measured by a different measurement scale. Therefore, a decision-maker must resolve the parameter selection problem to optimize each response. This problem is regarded as a multi-response optimization problem, subject to different response requirements. Most of the conventional methods are incomplete in that a response variable is selected as the primary one and is optimized by adhering to the other constraints set by the criteria [1].

Many heuristic methodologies have been developed to resolve the multi-response problem. Cornell and Khuri [2] explored the multi-response problem using a response surface method. Tai et al. [3] assigned a weight for each response to resolve the problem. Pignatiello [4] utilized a squared deviation-from-target and a variance to form an expected loss function for optimizing a multiple response problem. Layne [5] presented a procedure capable of simultaneously considering three functions: weighted loss function, desirability function, and distance function. While providing a multi-response example in which Taguchi methods are used, Byrne and Taguchi [6] discussed an example involving a connector and a tube. Logothetis

* Corresponding author. Tel.: +886-3-5731857; fax: +886-3-5722392.
E-mail address: ctsu@cc.nctu.edut.tw (C.T. Su).

0166-3615/02/$ – see front matter
PII: S0166-3615(02)00140-9

and Haigh [7] also discussed a manufacturing process characterized by five responses. In doing so, they selected one of the five response variables as primary and optimized the objective function sequentially while ignoring possible correlations among the responses. Optimizing the process with respect to any single response leads to nonoptimum values for the remaining characteristics.

Optimizing the parameter selection problem requires developing a model capable of accurately describing the input–output behavior and capturing the range of these input–output parameters. Therefore, this work presents a neural–genetic algorithm that combines the neural network and genetic algorithm to identify the nonlinear relationship between input and output parameter and obtain a near-optimal parameter combination. The neural networks have been extensively used to model the engineering process. Briefly, the neural network maps the input–output observed data and, in doing so, defines the fitness function of the parameter selection. Consequently, the genetic algorithm utilizes the fitness function to identify the optimal solution of the problem. In addition, this study proposes a concurrent optimization performance index to obtain a preferable solution by using a desirability function developed by Derrienger and Suich [8]. Capable of concurrently optimizing several responses and allowing the user to weigh the responses by their importance, the desirability function is easily understood and intuitive [6]. A case study demonstrates the effectiveness of the proposed approach.

The rest of this paper is organized as follows. Section 2 describes neural networks, genetic algorithms and the hybrid neural–genetic algorithm. Section 3 provides details of the multi-response optimization technique. Section 4 describes a case study of the silicon manufacturing process in Taiwanto show how the proposed algorithm is implemented. Concluding remarks are finally drawn in Section 5.

2. Optimization approach

2.1. Neural networks

Describing a manufacturing process precisely is generally too difficult by a mathematical function. A recent work adopted neural networks to elucidate the ability to learn complex relationships between parameters and responses, usually for process and quality control [9]. These models are frequently used to identify optimal process settings. An approximated model can be constructed using a neural network. Although statistical regression methods and neural network method both can effectively correct the dimensional measurements of geometric features on a part profile, Chang et al. [10] indicated that neural network methods will be a very powerful alternative for precision measurement using computer vision system.

Neural networks have been successfully applied to diverse areas such as speech synthesis and pattern recognition [11]. Once trained, a neural network can be evaluated very quickly, particularly during the optimization phase. Recent review of neural network applications in manufacturing, Zhang and Huang [12] cited such diverse venues as milling, metal cutting, injection modeling, arc welding and spray painting. Details regarding further applications can be found in [13–17].

Neural networks are formed by processing parallel units called neurons, which closely resemble the structure of a human neurological system. The elementary processors are interconnected so that knowledge pertaining to the relationship between input and output parameters are stored in the weights of the connections between them. Each neuron except the first layer contains the weighted sum of previous input neuron by an exponential function. This function allows neural networks to be generalized with a wide range of application.

Neural networks can be categorized into network structures such as multilayer perceptron, the feedback model of Hopfield [18] and Hopfield and Tank [19], the adaptive resonance technique (ART) networks and Kohonen network etc., and the learning methods such as back-probagation. The ability to learn is one of the main advantages that makes the neural networks so attractive. They also have the capability of performing parallel processing and possess significant fault tolerance. Since the BP neural network can be used to approximately realize continuous mapping [20], this work adopts the BP neural network owing to its ability to map the complex relationship between input data and corresponding outputs.

2.2. Genetic algorithms

Genetic algorithms (GAs), an optimization methodology based on a direct analogy to Darwinian natural selection and genetics in biological systems, is a promising alternative to conventional heuristic methods [21]. GAs differ from conventional search techniques that conduct a point-to-point search in the solution space. GAs work with a set of candidate solutions called population and, based on the Darwinian principle of "survival of the fittest", obtain the optimal solution after a series of iterative computations. This characteristic, associated with their stochastic nature, enables GAs to deal with large search spaces randomly and efficiently.

Genetic algorithms (GAs) have been extensively used to optimize complicated production systems. GAs are known for their robustness and effective overall search capabilities [22]. Hung and Adeli [23], and Hsu et al. [24] demonstrated the superiority of GAs over other networks capability in terms of its optimum search. Highly promising for obtaining near optimal solutions to complex problems, GAs have been extensively applied to diverse areas such as scheduling and sequencing [25–28], cellular manufacturing [29], PCB layout design [30], and process control strategies [31].

GA, a local search technique, can find solutions for a wide range of application. To achieve the desired response, GAs generate a successive population of alternate solutions which are represented by a chromosome, i.e. a solution to the problem, until acceptable results are obtained. In this manner, a GA can quickly yield a successful outcome without examining all possible solutions to the problem. The procedure using the fitness function is to assess the performance of the solution. The reproduction, crossover, and mutation are the main operators that randomly impact the fitness value. Chromosomes are selected for reproduction by evaluating the fitness value. The fittest chromosomes are then saved and copied into the next generation. Crossover, the critical genetic operator that allows new solution regions in the search space to be explored, is a random mechanism for exchanging genes between two chromosomes. The probability of crossover is generally set between 0.5 and 0.9. Mutation, in which the genes may occasionally be altered, i.e. a "0" becomes an "1" or vice versa. During the search, the mutation must avoid the premature loss important information although they are typically set at an extremely low value, 0.01 to 0.05.

2.3. A hybrid neural–genetic algorithm

This study proposes a novel hybrid neural–genetic algorithm to determine the parameter settings in a manufacturing process. The proposed approach combines the neural network and GA to the problem. The proposed approach consists of two stages. The first stage in a hybrid procedure involves identifying the desirability function deriving from the multiple responses. A BP network is trained to derive the relationship between input parameters and output responses. Notably, the trained network can accurately predict the behavior of possible parameter combinations. Thus, tuning the input parameters in the trained network allow us to obtain the corresponding response. The trained network is used as the fitness function in the GA. During the second stage, GA is directly used to solve the problem. Herein, the chromosome is used to represent the possible solution. Each gene in the chromosome represents the value of the input parameter. For example, a manufacturing process has three input parameters P, Q, and R. A chromosome can represent the value of the three parameters (P, Q, R), respectively. The essential genetic operators during the iterative procedure can be found in the previous section. These operations are conducted to obtain the optimal response, which is evaluated by the fitness function. Therefore, the optimal parameter of the problem can be obtained. Fig. 1 schematically depicts the proposed hybrid procedure. The detailed procedure is summarized as given further.

Step 1. Collect the input parameters and corresponding responses.

Step 2. Develop a BP network model to obtain the relationship between the input parameters and output responses. The trained network is referred to as a fitness function.

Step 3. Set the GA operating condition (e.g. population size, maximum number of generation, parameter number, crossover rate, and mutation rate).

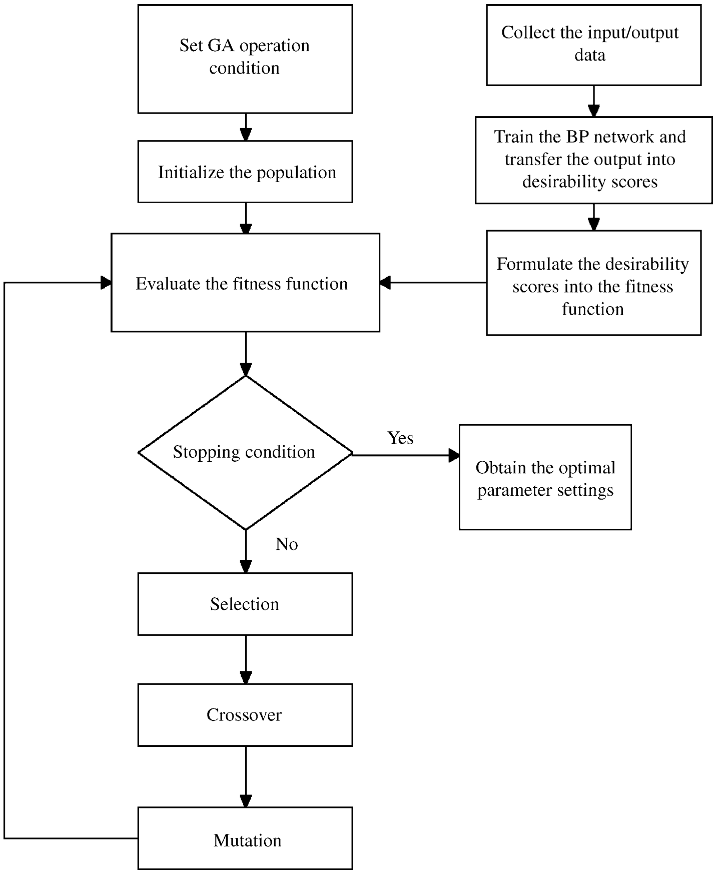

Fig. 1. The Schematic diagram of the hybrid procedure.

Step 4. Create an initial population by randomly selecting the value of the input parameters.

Step 5. Repeat steps 6–10 until the stopping condition is reached.

Step 6. Calculate the responses, by inputting the parameter values to the fitness function (responses are taken from Step 2).

Step 7. Select the parameter values according to the computed response.

Step 8. Crossover the fitness parameter values.

Step 9. Mutate the parameter values to yield the next generation.

Step 10. Obtain the current optimal parameter values.

Step 11. Obtain the optimal parameter settings and responses.

3. Multi-response problem

3.1. General scheme

Optimization of the multi-response problem is an attempt to optimize all output responses simultaneously. Among the concurrent optimization methods, most of the authors used the approaches that combine all the different response requirements into one composite requirement [1]. Hence, the compromise solution is obtained in a much simpler way. A simple weighting method was found in Ilhan et al. [32], as applied in an electrochemical grinding (ECG) process. Zadeh [33] normalized each response and then gave a simple weight for each response. The discussion regarding the assignments of weights can be found in [34].

3.2. Desirability function

The desirability function transforms each response to a corresponding desirability value $d_i (0 \leqq d \leqq 1)$. All the desirabilities are combined to form a composite desirability function:

$$D = f(d_1, d_2, d_3, \ldots d_n) \tag{1}$$

where n is number of responses. The value of D may be defined as the geometric mean of the d_i's and thus D lies between 0 and 1. Consequently, the desirability approach can convert a multi-response problem into a single-response one. The plant manager can easily determine the optimal parameters among a group of solutions. However, the user specifies the parameters "p" of Eqs. (2) and (3) based on technical, economical and other considerations. For two-sided specification limits with a target value T for the response Y, Derringer [33] used the following transformations

$$d_i = \begin{cases} \left(\dfrac{Y_i - \mathrm{LSL}_i}{T_i - \mathrm{LSL}_i}\right)^p & \mathrm{LSL}_i \leq Y_i \leq T_i \\ \left(\dfrac{Y_i - \mathrm{USL}_i}{T_i - \mathrm{USL}_i}\right)^p & T_i \leq Y_i \leq \mathrm{USL}_i \\ 0 & Y_i < \mathrm{LSL} \text{ or } Y_i > \\ & \mathrm{USL} \text{ otherwise} \end{cases} \tag{2}$$

where LSL_i is ith lower specification of limit; USL_i the ith upper specification of limit; T_i the ith target of the response; and Y_i is the ith response.

For a one-sided specification limit (higher-the better-type response), Derringer [8] suggested the following transformations:

$$d_i = \begin{cases} 0 & Y_i \leq \mathrm{LSL}_i \\ \left(\dfrac{Y_i - \mathrm{LSL}_i}{Y_i \max - \mathrm{LSL}_i}\right)^p & \mathrm{LSL}_i < Y_i < Y_i\max \\ 1 & Y_i \geq Y_i\max \text{ otherwise} \end{cases} \tag{3}$$

where $Y_i$max is the highest value which is practically attainable.

3.3. The proposed approach

This study proposes a desirability function to solve multi-response optimization problems. When the multi-response problem is transformed into a single-response problem, the single-response problem is divided into two problems: how to specify the weights and how to transform each response into a more "desirable" response. This work proposes a composite approach by using the desirability function [8,35] to determine the overall value of scalar function. The scalar function ranges between 0 and 1, and the larger the value implies a more stringent user requirement. This value can also be used as a performance index for the multiple responses.

Herein, the hybrid neural–genetic algorithm and the desirability function are combined. The neural network is first trained by using the process production data. The desirability function is then used to transform the multiple responses into a single response. Finally, GA is applied to obtain the best desirability value (i.e. fitness value). Consequently, the optimal parameter settings of the manufacturing process can be determined.

4. Illustrative example

4.1. Problem description

The silicon compound of RC50 is a critical part that is used in the computer peripheral and medical appliance assemblies. Fig. 2 shows a flow chart of RC50 silicon compound manufacturing process. The process starts by mixing two silicon raw materials: silicon filler and catalyst under a high temperature. The materials are then polymerized in a chemical chamber under parameter settings, which were originally assigned by an equipment provider in Japan. The process of polymerization in the chamber is the most complicated and critical process that strongly depends on parameter settings (e.g. N_2 flow, release agent, conductivity, and oil absorption). Filtering, water cleaning and purification are then applied to remove small amounts of contamination and improve the

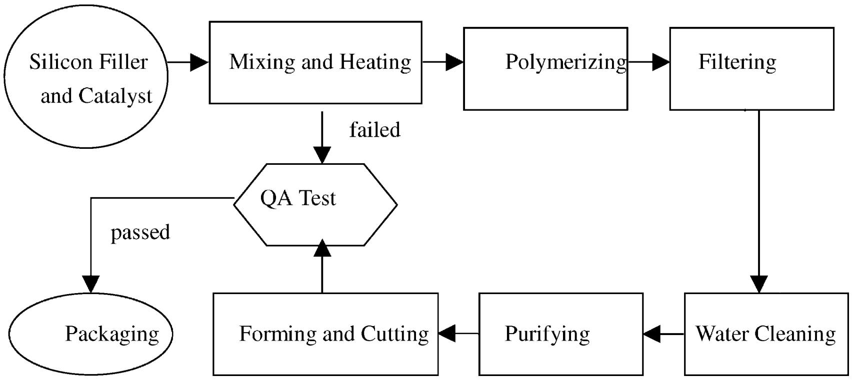

Fig. 2. Manufacturing process of silicon compound of RC50.

product's characteristics. Table 1 lists the operating ranges with respect to process parameters. To satisfy customer's requirements, QA takes a certain amount of finished goods to conduct a 2-day functional test after forming and cutting processes. The entire product is sent back to line for reworking if it fails the QA test. Consequently, it interferes with the regular production schedules, resulting in late deliveries.

The manufacturer had difficulty in achieving efficient process control due to insufficient knowledge of the relationship between process parameters and corresponding responses. Table 2 lists the process capability indices for the seven responses; the performance is obviously improved in terms of process capability. However, conducting experimental design to improve the quality would be infeasible since the factory employs a continuous process with an enormous amount of material and long production time. Hence, this study adopts the historical process data to obtain the required model, thereby optimizing the product's responses.

The problem considered herein is the multi-response optimization problem as the process parameter selection applied in a silicon manufacturing company. The problem dealing with the multi-input and multi-output is common in practice. The silicon factory must determine the levels of 14 parameters for seven quality characteristics to satisfy different customer's requirements.

4.2. Training of neural networks

The relationship model between parameters and responses is developed by using BP neural network, in which historical production data of four hundred lots are employed for training as well as 100 lots are used for testing. The convergence criterion employed in the network training is the RMSE. Table 3 lists

Table 1
Input parameters and their operation ranges

Parameter	Code	Range
Water content	P1	80–90
pH value	P2	10–20
Conductivity	P3	0.0017–0.002
Release agent	Z1	2.26–3.2
Specific area	W1	1.2–3.1
Oil content	W2	0.1–0.2
Whiteness	W3	1.0–1.1
Particle size	W4	3.3–3.7
Al_2O_3 content	W5	0.15–0.3
Fe_2O_3 content	F1	0.5–1.0
Residue sieve	F2	10–15
Volatile material	F3	7.5–17.5
Temperature	C1	17.1–25
Nitrogen flow	C2	50–60

Table 2
Specifications and process capability indices of seven responses

Responses (code, dimension)	Specification	C_p	C_{pk}
Density (Y_1, g/cc)	1.140–1.15	0.71	0.67
Plasticity (Y_2, point)	210–250	1.00	0.98
Hardness (Y_3, durometer)	51–55	0.52	0.49
Tensile strength (Y_4, kg/cm^2)	$\geq$6.2	0.80	0.80
Elongation (Y_5, %)	$\geq$250	0.79	0.79
Shrinkage (Y_6, %)	3.7–3.9	0.39	0.34
Rebound (Y_7, %)	$\geq$66	0.34	0.34

Table 3
Options for neural networks

Architecture	RMSE	
	Training	Testing
14-5-7	0.10121	0.11912
14-6-7	0.09963	0.09632
14-7-7	0.08521	0.09541
14-8-7	0.08754	0.09674
14-9-7	0.08737	0.09724

Note: Learning rate: 0.2, momentum: 0.9, and number of epochs: 10,000.

several options of the neural network architecture, in which the structure 14-7-7 is selected to obtain a better performance. The trained network 14-7-7 is used as the fitness function of the GA, as further explored in the next section.

4.3. Determination of the fitness function

Once a BP neural network was well trained, the weights connected between layers in the neural network structure illustrated the relationship between input parameters and output responses. The value of each response was calculated by weighted sum connected to output node and transferred by an activation function (e.g. sigmoid function). Hence, the weights obtained from a trained BP neural network and activation functions of each node formed the fitness function adopted in GA optimization procedure. In this case, responses Y_1, Y_2, Y_3, and Y_6 have the corresponding target values and Y_4, Y_5, and Y_7 have lower specifications. After training the BP neural network, the value of (Y_1–Y_7) will be the near-optimal solution in this case. Then, using the Eqs. (2) and (3) will transfer the value of Y_i into d_i. Herein, a geometric mean of seven responses is employed as a desirability function to solve the multi-response problem. We have

$$D = f(d_1, d_2, d_3, d_4, d_5, d_6, d_7) \tag{4}$$

where d_i is calculated from Eqs. (2) and (3). For computational convenience, p is equal to 1 in this case. While a d_i is approaching to 1, it means that d_i is approaching the target. While the D is approaching to 1, it is noted that each response in the process is simultaneously approaching to 1 (say the target). The value of D demonstrates the performance metric of the proposed method.

Table 4
Implementation results of GA

Item	Data
The largest D value in 20 runs	0.7212
The smallest D value in 20 runs	0.5724
Average D value	0.6602
S.E.	0.0413

4.4. Optimization using genetic algorithm

Each input parameter in a silicon factory is normalized to the value between 0 and 1 and they are combined into one string. For example, the input parameters listed in Table 1, are transformed into the chromosome representation (P_1, P_2, P_3, Z_1,..., C_2) in a string. Strings are randomly generated to form the initial population. When GA is applied to optimize

Table 5
A comparison of responses

Method	Parameter values			Predicted responses		
Initial state	$P_1 = 85.0$,	$P_2 = 15.00$,	$P_3 = 0.0019$,	$Y_1 = 1.142$,	$Y_2 = 224.00$,	$Y_3 = 53.4$,
	$Z_1 = 2.96$,	$W_1 = 1.70$,	$W_2 = 0.10$,	$Y_4 = 7.00$,	$Y_5 = 270.00$,	$Y_6 = 3.85$,
	$W_3 = 1.00$,	$W_4 = 3.30$,	$W_5 = 0.15$,	$Y_7 = 67.00$,		
	$F_1 = 1.00$,	$F_2 = 10.00$,	$F_3 = 15.00$,			
	$C_1 = 18.1$,	$C_2 = 60.00$,				
Proposed approach (optimal condition)	$P_1 = 84.3$,	$P_2 = 10.54$,	$P_3 = 0.0019$,	$Y_1 = 1.145$,	$Y_2 = 229.94$,	$Y_3 = 52.37$,
	$Z_1 = 2.76$,	$W_1 = 1.75$,	$W_2 = 0.16$,	$Y_4 = 8.4178$,	$Y_5 = 298.22$,	$Y_6 = 3.81$,
	$W_3 = 1.06$,	$W_4 = 3.53$,	$W_5 = 0.24$,	$Y_7 = 74.88$,		
	$F_1 = 0.64$,	$F_2 = 13.06$,	$F_3 = 14.43$,			
	$C_1 = 28.76$,	$C_2 = 50.18$,				

Table 6
A comparison of process capability

Response	C_p		C_{pk}	
	Initial	Proposed	Initial	Proposed
Y_1	0.71	1.18	0.67	1.15
Y_2	1.00	1.54	0.98	1.32
Y_3	0.52	1.31	0.49	1.23
Y_4	0.80	1.17	0.80	1.17
Y_5	0.79	1.20	0.79	1.20
Y_6	0.39	1.26	0.34	1.19
Y_7	0.34	1.13	0.34	1.13

the silicon parameter selection, the essential operators, including reproduction, crossover and mutation, should be determined in advance. Herein, a roulette wheel approach is adopted as the selection procedure. The crossover rate and mutation rate are set as 0.5 and 0.01, respectively. Fifty strings are randomly generated to establish the initial population. Notably, 5000 generations were processed. In this case, the optimal target, a geometric mean of seven responses will be set to 1. The fitness function is formed by the BP learning algorithm and desirability function. The specification of each response will be the constraints in the GA optimization procedure.

4.5. Results

The above information is used and the GA is executed 20 runs. Table 4 summarizes the implementation results. The higher the D value implies a much better compromised solution. The largest D value is 0.7212 and its optimum chromosome is (84.3, 10.54, 0.0019, 2.76, 1.75, 0.16, 1.06, 3.53, 0.24, 0.64, 13.06, 14.43, 28.76, 50.18). These settings are the optimal condition for our 14 process parameters. The predicted responses under the optimal condition are $Y_1 = 1.145$, $Y_2 = 229.94$, $Y_3 = 52.37$, $Y_4 = 84.178$, $Y_5 = 298.22$, $Y_6 = 3.81$, $Y_7 = 74.88$. Table 5 compares the responses between the initial condition and the proposed one (optimal condition).

Table 6 also compares the initial process capability and the process capability based on the proposed approach. According to this table, the proposed approach outperforms the original state. Correspondingly, the feasibility of the proposed approach is established.

5. Conclusion

This study proposes an integrated method using neural network, genetic algorithm, and desirability function to optimize the manufacturing process with multiple responses. The neural network is used to explore the nonlinear multivariate relationship between the parameters and responses and then GA is performed to obtain the optimal parameter settings. During the implementation of GA, the fitness function is defined in terms of a desirability function, which is utilized to transform multiple responses into a single response. The proposed approach can easily and efficiently achieve the optimization of the complex process with multiple responses. These settings facilitate the process engineers in achieving acceptable process control during the production. In addition, all of the experiments are conducted under computerized simulations with historical production data without any manufacturing interruption. The improvement in process capability allows the factory to more easily fabricate products with superior quality.

References

[1] P. Das, Concurrent optimization of multi-response product performance, Quality Engineering 11 (3) (1999) 365–368.

[2] J.A. Cornell, A.I. Khuri, Response surfaces: Designs and Analysis, Marcel Dekker, New York, 1987.

[3] C.Y. Tai, T.S. Chen, M.C. Wu, An enhanced Taguchi method for optimizing SMT processes, Journal of Electronics Manufacturing 2 (1992) 91–100.

[4] J.J. Pignatiello, Strategy for robust multi-response quality engineering, IIE Transactions 25 (3) (1993) 5–15.

[5] K.L. Layne, Method to determine optimum factor levels for multiple responses in the designed experimentation, Quality Engineering 7 (4) (1995) 649–656.

[6] D.M. Byrne, S. Taguchi, The Taguchi approach to parameter design, Quality Progress 20 (1987) 19–26.

[7] N. Logothetis, A. Haigh, Characterizing and optimizing multi-response processes by the Taguchi method, Quality and Reliability Engineering International 4 (2) (1988) 159–169.

[8] G. Derringer, R. Suich, Simultaneous optimization of several response variables, Journal of Quality Technology 12 (4) (1980) 214–218.

[9] D.W. Coit, B.T. Jackson, A.E. Smith, Static neural network process model: considerations and cases studies, International Journal Production and Research 36 (11) (1998) 2953–2967.

[10] C.A. Chang, C.T. Su, A comparison of statistical regression and neural-network methods in modeling measurement errors for computer vision inspection systems, Computers and Industrial Engineering 28 (3) (1995) 593–603.
[11] R.P. Lipmann, Introduction to computing with neural nets, IEEE ASSP Magazine 4 (2) (1987) 4–22.
[12] H.C. Zhang, S.H. Huang, Applications of neural networks in manufacturing: a state-of-the-art survey, International Journal of Production Research 33 (3) (1995) 705–728.
[13] K.M. Tay, C. Butler, Modeling and optimizing of a MIG welding process-a case study using experimental designs and neural networks, Quality and Reliability Engineering International 13 (2) (1997) 61–70.
[14] C.C. Chiu, C.J.S. Huang, S.C. Chen, T.N. Cheng, C.T Su. G.S. Yang, International Journal of Quality Science 2 (2) (1997) 106–120.
[15] Y.F. Zhang, J.Y. Fuh, A neural network approach for early cost estimation of packaging products, Computers Industrial Engineering 34 (2) (1998) 433–450.
[16] J. Chen, P.T. Chu, D. Shan, H. Wong, S.S. Jang, Optimal design using neural network and information analysis in plasma etching, Journal of Vacuum Science Technology 17 (1) (1999) 145.
[17] M.D. Baker, C.D. Himmel, G.S. May, Time series modeling of reactive ion etching using neural networks, IEEE Transactions on Semiconductor Manufacturing 8 (1) (1995) 62–71.
[18] J.J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, National Academy of Science 79 (1982) 2554–2558.
[19] J.J. Hopfield, D.W. Tank, Neural computation of decisions in optimization problems, Biological Cybernetics 52 (1985) 141–152.
[20] K. Funahashi, One approach realization of continuous mapping by neural network, Neural Network 2 (3) (1989) 183–192.
[21] D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addision-Wesley, MA, USA, 1989.
[22] Z. Khan, B. Prasad, T. Singh, Machining condition optimization by genetic algorithms and simulated annealing, Computers and Operations Research 24 (7) (1997) 647–657.
[23] S.L. Hung, H. Adeli, A parallel genetic/neural network learning algorithm for MIMD shared memory machines, IEEE Transactions on Neural Networks 5 (6) (1994) 900–909.
[24] H.M. Hsu, S.P. Tsai, M.C. Wu, C.K. Tzuang, A Genetic algorithm for the optimal design of microwave filters, International Journal of Industrial Engineering 6 (4) (1999) 282–288.
[25] M. Qiu, Prioritising and scheduling road projects, genetic algorithm, Mathematics and computers in simulation 43 (3) (1997) 569–574.
[26] M. Mori, C.C. Tseng, Genetic algorithm for multi-mode resource constrained project scheduling problem, European Journal of Operational Research 100 (1) (1997) 131–141.
[27] A.K. Jain, H.A. Elmaraghy, Single process plan scheduling with genetic algorithms, Production Planning and Control 8 (4) (1997) 363–376.
[28] R. Cheng, M. Gena, Y. Tsujimura, A tutorial survey of job-shop scheduling problems using genetic algorithms: part II hybrid genetic search strategies, Computers and Industrial Engineering 37 (1) (1999) 51–55.
[29] Y. Gupta, M. Gupta, A. Kumar, C. Sundaram, A genetic algorithm-based approach to cell composition and layout design problems, International Journal of Production Research 34 (2) (1996) 447–482.
[30] L. P Khoo, K.M. Loh, A genetic algorithms enhanced planning system for surface mount PCB assembly, International Journal of Advanced Manufacturing Technology 16 (5) (2000) 289–296.
[31] S. Sette, L. Boullart, L. Van Langenhove, Using genetic algorithms to design a control strategy of an industrial process, Control Engineering Practice 6 (4) (1998) 523–527.
[32] R.E. Ilhan, G. Sathyanarayanan, R.H. Storer, T.W. Liao, Off-line multiresponse optimization of electrochemical surface grinding by a multi-objective programming method, International Journal Machine Tools and Manufacture 32 (3) (1992) 435–451.
[33] L. Zadeh, Optimality and non-scalar-value performance criteria, IEEE Transactions on Automatic Control 8 (59) (1963) 59–60.
[34] T. Murata, H. Ishibuchi, H. Tanaka, Multi-objective genetic algorithm and its applications to flowshop scheduling, Computers and Industrial Engineering 30 (4) (1996) 957–968.
[35] E. Harrington, The desirability function, Industrial Quality Control 21 (10) (1965) 494–498.

Te-Sheng Li received the BS degree in industrial management from National Chen-Kung University, Tainan, Taiwan, ROC and the MS degree in engineering management from University of Missouri-Rolla, USA in 1985 and 1990, respectively. He holds the PhD degree in industrial engineering and management from the National Chiao-Tung University, Hsinchu, Taiwan, ROC in 2002. Dr. Li is currently an associate professor in the Department of Industrial Engineering and Management at Ming Shin University of Science and Technology, Taiwan. His research interests include quality engineering, neural networks application and data mining in semiconductor manufacturing industry.

Chao-Ton Su received the PhD degree in Industrial Engineering from University of Missouri-Columbia, USA, in 1993. He is a Professor in the Department of Industrial Engineering and Management at National Chiao Tung University, Taiwan. His current research activities include quality engineering and management, operation management and data mining in industrial applications. He has published over 50 referred international journal papers. Dr. Su obtained the 2000–2001 Outstanding Research Award of the National Science Council of the Republic of China. In 2001, Dr. Su also obtained the Individual Award of the National Quality Awards of the Republic of China.

Tai-Lin Chiang is currently an associate professor in the Department of Business Administration at Ming Shin University of Science and Technology, Taiwan. He holds a PhD in Industrial Engineering and Management from National Chiao-Tung University, Taiwan. His research interests include Six Sigma, quality engineering, neural networks applications, and semiconductor manufacturing engineering.