Proceedings of CECNet 2021
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA210408

Parametric Optimization of Integrated Circuit Assembly Process: An Evolutionary Computing-Based Approach

Tatjana SIBALIJA[1]
Belgrade Metropolitan University, Belgrade, Serbia

Abstract. Strict demands for very tight tolerances and increasing complexity in the semiconductors' assembly impose a need for an accurate parametric design that deals with multiple conflicting requirements. This paper presents application of the advanced optimization methodology, based on evolutionary algorithms (EAs), on two studies addressing parametric optimization of the wire bonding process in the semiconductors' assembly. The methodology involves statistical pre-processing of the experimental data, followed by an accurate process modeling by artificial neural networks (ANNs). Using the neural model, the process parameters are optimized by four metaheuristics: the two most commonly used algorithms - genetic algorithm (GA) and simulated annealing (SA), and the two newly designed algorithms that have been rarely utilized in semiconductor assembly optimizations - teaching-learning based optimization (TLBO) and Jaya algorithm. The four algorithm performances in two wire bonding studies are benchmarked, considering the accuracy of the obtained solutions and the convergence rate. In addition, influence of the algorithm hyper-parameters on the algorithms effectiveness is rigorously discussed, and the directions for the algorithm selection and settings are suggested. The results from two studies clearly indicate superiority of the TLBO and Jaya algorithms over GA and SA, especially in terms of the solution accuracy and the built-in algorithm robustness. Furthermore, the proposed evolutionary computing-based optimization methodology significantly outperforms the four frequently used methods from the literature, explicitly demonstrating effectiveness and accuracy in locating global optimum for delicate optimization problems.

Keywords. Integrated circuit assembly, parametric process optimization, evolutionary algorithms (EAs), teaching-learning based optimization (TLBO) algorithm, Jaya algorithm, artificial neural networks (ANNs)

1. Introduction

An accurate design of process parameters is of utmost importance in semiconductor industry, due to extremely tight tolerances and zero-defect demands for the process outputs (i.e. device characteristics). The objective of parametric optimization is to find an optimal process parameters set that meets requirements for the response mean and reduce its variability. There is a variety of optimization approaches from literature, but not all of them are fully efficient and objective for a multi-response case. The approaches based on statistical techniques are mainly unable to locate a global optimum, since they consider only discrete (local) process parameter values utilized in experimental trials or

[1] Corresponding author; E-mail: tsibalija@gmail.com, tatjana.sibalija@metropolitan.ac.rs

data sets. The exception is response surface methodology (RSM), the most common optimization method that applies a hill climbing or descending tools over a combined response plot [1]. However, it has a tendency to be trapped into local optima for a highly non-linear process with a large number of control factors, and it does not explicitly address correlations among responses and the response variation. The Taguchi's robust parameter design simultaneously assesses both the response mean and variability, taking into account the response specification by using signal-to-noise ratio (SNR) or quality loss (QL) function; the latter shows a loss encountered by the user if the product response deviates from the target [2]. But, the traditional Taguchi method has been designed for a single response. Various approaches have been developed to integrate multiple responses in Taguchi method. Some approaches used principal component analysis (PCA) over SNR or QL data to obtain uncorrelated variables, where only components with eigenvalue higher than 1 are considered [3], thus considering only a part of the original data variability. An improved approach that considered all components was applied to optimize wire bonding process [4]. In [5] PCA was applied directly on the response data, considering all independent components, but giving misleading results since the response specifications are not examined. A number of works applied grey relational analysis (GRA) over SNR data to integrate multiple responses by assuming that all responses are of the same significance [6]; however, the response weights are allocated in a subjective manner. For optimizing wire bonding parameters vs. two responses, conflicting response specifications were addressed by the fuzzy logic [7]. Although it resulted with improved process responses, this approach considered only local solutions.

Approaches based on soft computing, including evolutionary computing techniques, present a viable alternative due to a search across a continuous multi-dimensional space of solutions. There are a few soft computing-based approaches that address parametric optimization problems from the semiconductor industry. For wire bonding process, multiple responses were integrated via GRA, and the process was mapped and optimized using combination of artificial neural networks (ANNs) and genetic algorithm (GA) [8]. The lithography process with two responses was modeled using ANNs and parameters were optimized by particle swarm optimization (PSO) [9]; however, the response specifications and correlations are not explicitly addressed. For the hi-power LED packaging, multiple responses were transformed into QL values and their sum presented the objective function; process was mapped by genetic programming and optimized by PSO [10]. But, the integration of QLs into a single objective was not performed in a totally objective manner. In overall, although effective in addressing various optimization problems, EAs have been frequently criticized due to a significant effect of their hyper-parameters on the accuracy of the obtained solutions.

This work briefly presents an advanced methodology for parametric process optimization, including four metaheuristic algorithms whose performances are benchmarked to obtain the most accurate solution. Its implementation in semiconductor industry, where accuracy is of paramount importance for the device quality and reliability, is illustrated in two use cases showing integrated circuit assembly process (wire bonding).

2. Evolutionary computing-based optimization methodology

The proposed optimization methodology involves three major stages: data pre-processing by statistical methods, process modeling and optimization. In the first stage, experimentally obtained response data are converted into quality loss (QL) values, according to the type of response. The QL directly shows a response financial significance for the user without imposing any assumptions, as defined by Taguchi [2]:

$$QL = K \cdot \left\{ \begin{array}{ll} \frac{1}{n}\sum_{i=1}^{n} {y_i}^2 & \ldots\ for\ smaller - the - better\ type \\ \frac{1}{n}\sum_{i=1}^{n}(y_i - t)^2 = \frac{n-1}{n}s^2 + (\bar{y} - t)^2 & \ldots\ for\ nominal - the - best\ type \\ \frac{1}{n}\sum_{i=1}^{n}\frac{1}{{y_i}^2} & \ldots\ for\ larger - the - better\ type \end{array} \right\} \quad (1)$$

Since responses from the same process are correlated, it is necessary to obtain independent variables by applying PCA on the normalized QL data (NQLs). In the proposed approach, the number of independent principal components – PCs ((j=1,..., p) corresponds to the number of responses (i=1,..., p). For each experimental trial (k=1,..., m), the PC sores ($Y_j(k)$) are obtained based on the elements of the corresponding eigenvector (V_{ij}) and the NQL data, as follows:

$$Y_j(k) = \sum_{i=1}^{p} NQL_i\,(k) \cdot V_{ij} \quad (2)$$

GRA is applied over $Y_j(k)$ data to integrate multiple Y_j, i.e. to obtain a grey relational grade ($\gamma_k \in [0,1]$) using a previously calculated grey relational coefficient ($\varepsilon_j(k)$) and PC weights (ω_j) obtained from PCA:

$$\gamma_k = \sum_{j=1}^{P} \omega_j \cdot \varepsilon_j(k) \quad (3)$$

The γ value is adopted as a process performance measure: the higher the γ, the better is the process. The details of PCA could be found in [11], and of GRA in [12].

In semiconductor industry the mathematical model of a process if typically unknown. Hence, back propagation (BP) ANNs are engaged to identify the relationship between process parameters (input) and γ (output), using the tangent sigmoid and linear transfer functions. The number of neurons in the hidden layer is varied to obtain the best topology with minimal mean square error (MSE) and maximal coefficient of correlation (R) between original and predicted data. The procedure is explained in detail in [13].

The optimization stage aims to obtain optimal process parameters that maximize process performance (γ). Since metaheuristic effectiveness highly depends on the hyper-parameters, four algorithms are tested: the most frequently used ones (GA and SA) that are highly affected by their settings, and two recent algorithms (TLBO and Jaya) that are free of the algorithm specific hyper-parameters. For the TLBO and Jaya, the only factors to be specified are common for all EAs: population size and total iterations count. In this work, the population size of 5n (n is number of design variables, i.e. process parameters) is adopted for all four algorithms, with the total iterations count of 2000.

The GA is the most frequently used metaheuristic in process optimizations. It searches throughout the continual space of solutions using a population, so it belongs to EAs [14]. To improve the current members, it employs several genetic operations: selection, crossover, mutation, migration. Members with the highest objective function are adopted for the next iteration, and iterative procedure is repeated until a total iterations count is met. In this work, three values of the two major operators are tested: (i) selection: *stochastic uniform, roulette wheel* and *tournament*, (ii) crossover: *single point, two points* and *arithmetic*. Hence, in total 9 GAs are developed for each optimization problem.

The SA algorithm does not belong to EAs, since it is based on a point-to-point search. It mimics the metal annealing process: heating a material to the melting point and slowly decreasing the temperature to maintain a thermal equilibrium. Starting with a sufficiently high initial temperature, new points are generated using an annealing function. The probability of a new point acceptance is defined by the expression [15]:

$$P(E,T) = \exp(-\frac{\Delta E}{K\cdot T}) \tag{4}$$

where ΔE is the difference of energy, i.e. difference of the objective function between a new and the old point, T is the current temperature and K is the Boltzmann constant. The temperature decrease is controlled by a temperature function. When a temperature is very low, a reannealing is performed (controlled by reannealing interval) to raise up the temperature. The above procedure continues until a specified number of iterations is reached. Different values of the four major parameters are tested: (i) initial temperature: 10, 100 and 500, (ii) temperature function: *exponential, fast* and *Boltzmann* function, (iii) annealing function: *fast* and *Boltzmann* algorithm; (iv) reannealing interval: 10 and 100. Therefore, 36 SA algorithms with heterogeneous settings are run for each problem.

The TLBO algorithm is EA that mimics the teaching-learning behavior in a classroom: the population is composed from students; the design variables are teaching subjects; the student's knowledge (grade) refers to the objective function. First, students are learning from the teacher to improve their knowledge. For the jth subject (j = 1, … n) in the kth teaching-learning cycle (k = 0, 1, … I_{max} ; I_{max} is the total iterations count), the updated solution for the ith student is [16]:

$$X^k_{new,i,j} = X^k_{old,i,j} + r\left(TG^k_j - T_F LG^k_j\right) \tag{5}$$

where TG_j^k is the teacher grade, LG_j^k is the average students' grade, r is a random number between 0 and 1, and T_F is the teaching factor between 1 and 2. If a new solution is better than the old one, it is adopted for the next stage where student knowledge is further improved by interacting with other fellows. Assuming that uth student learns from the vth student (u ≠ v), the updated solution in this stage is computed based on the objective functions $f(X_u^k)$ and $f(X_v^k)$ of uth and vth students, respectively [16]:

$$if\ \ f(X^k_u) < f(X^k_v)\ \ then\ \ X^k_{new\_L,u,j} = X^k_{u,j} + r\left(X^k_{v,j} - X^k_{u,j}\right)$$

$$\text{else}\ \ X^k_{new\_L,u,j} = X^k_{u,j} + r\left(X^k_{u,j} - X^k_{v,j}\right) \tag{6}$$

Based on the objective function assessment for an updated solution, the students with the best knowledge are involved in the next iteration. The process is repeated until a termination condition (i.e. the total number of iterations) is met.

The Jaya algorithm is a very simple EA, based on a straightforward principle: the optimal solution is obtained by approaching the best and moving away from the worst solutions. For the ith design variable (i = 1, … n) of the kth population member (k = 1, … m) in the lth iteration, the candidate solutions are calculated according to the design variables of the best ($X_{i,best,l}$) and the worst ($X_{i,worst,l}$) members in the population [17]:

$$X'_{i,k,l} = X_{i,k,l} + r_{1,i,l}\left(X_{i,best,l} - \left|X_{i,k,l}\right|\right) - r_{2,i,l}\left(X_{i,worst,l} - \left|X_{i,k,l}\right|\right) \tag{7}$$

where $r_{1,k,l}$ and $r_{2,k,l}$ are random numbers between 0 and 1. The objective functions of updated solutions are evaluated; solutions that enhance the objective are adopted for the next iteration. The process is reiterated until the specified iterations count is reached.

Effectiveness of the four algorithms are compared in respect to the accuracy of the obtained solution (the highest objective function) and the convergence rate (minimal number of iterations needed to locate the global optimum).

3. Implementation of optimization methodology in semiconductor industry

3.1. Use case 1: Thermosonic copper wire bonding

Aiming to ensure a reliable performance of a microelectronic device assembled using a copper wire, the objective was to establish a solid bond between copper wire (diameter 50 µm) and aluminum pads at the die that contains integral circuitry, in a thermosonic wire bonding of the power amplifier device. The following control parameters were varied at two levels: contact power (CP_{Cu}), contact force (CF_{Cu}), base force (BF_{Cu}), base power (BP_{Cu}). Two responses were observed at the output: (i) the average ball shear test (BS) showing the average strength of the intermetallic connections between Cu wires and Al die pads (in a device with 41 ball bonds), and (ii) the number of oxide damages (NoOD) found as a failure mode after shear tests in a device. The objective is to find an optimal parameters set to achieve the nominal BS value of 160 10^{-2} N and to minimize NoOD occurrence that indicates corrosion micro cracks inside the die pad.

The experimental plan was based on the orthogonal array (OA) L_8 including five added replicates [18] (Table 1). The responses were converted into QLs as per formula (1). Application of PCA over NQLs resulted with two independent PCs, integrated in the process performance (γ) using weights obtained from PCA: [0.684; 0.316]. The best BP ANN with topology 4-7-1 established an accurate process model (process parameters vs. γ), with minimal error (MSE=4.4 10^{-4}) and maximal R (99.8).

Four algorithms were employed in the optimization stage. As seen from Table 2, TLBO found a marginally better solution than Jaya; their solutions are significantly better than the solution obtained by SA; the maximal objective found by GA is lower that the best SA algorithm result. Figure 1 shows convergence of the best algorithms (ones that found maximal objectives, as presented in Table 2), i.e. change in the objective function along iterations. It could be seen that GA converged in 260th iteration, SA in 20th iteration, TLBO in 1876th iteration and Jaya in 276th iteration. Therefore, SA was the fastest algorithm followed by GA, but both failed to locate the global solution. TLBO was the slowest; Jaya convergence rate is comparable to GA. Although TLBO was the slowest algorithm, it is important to note that TLBO and Jaya obtained the global or near-to-global solution contrary to GA and SA. The most disperse results are obtained by GA; TLBO and Jaya showed a narrow range of results obtained in multiple runs since they do not have algorithm specific hyper-parameters to be tuned, contrary to GA and SA.

Table 1. Use case 1: experimental results, NQLs and process performance measure

trial	**Control parameters**				**Responses**		**NQL$_i$(k)** i=1,2; k= 1,...,13		γ_k
no.	CP_{Cu}	CF_{Cu}	BF_{Cu}	BP_{Cu}	BS	NoOD	NQL_{BS}	NQL_{NoOD}	*k=1,.,13*
1	0	250	200	40	110.15	1	0.846	0.01	0.3830
2	0	250	200	80	175.81	10	0.129	1.00	0.3536
...	…	…	…	…	…	…	…	…	…
13	30	250	350	40	146.57	0	0.112	0.00	0.8235

Table 2. Use case 1: results of four metaheuristic algorithms

Metaheuristic algorithm	**GA**	**SA**	**TLBO**	**Jaya**
Objective function range	0.9255÷0.9687	0.9702÷0.97054	0.97351÷0.97362	0.97351÷0.97361
Optimal parameters range	[15÷30.1; 400; 299.2÷306.4; 40÷45.5]	[39.3÷40; 395.1÷400; 380.2÷395.6; 62÷69]	[23.3÷23.4; 400; 261.1÷264; 40]	[23.3÷23.8; 400; 261.1÷264.5; 40]
Maximal objective	0.96870	0.97054	0.97362	0.97361
Optimal parameters	[28; 400; 299; 40]	[40; 400; 385; 66]	[23.4; 400; 264; 40]	[23.3; 400; 264.4; 40]
Iterations to max. objective	260	20	1876	276

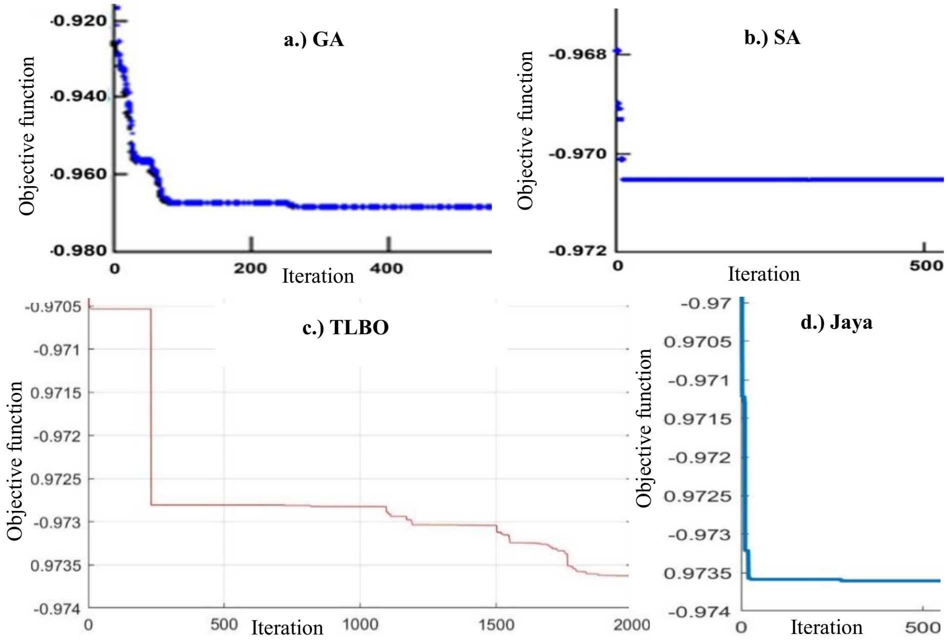

Figure 1. Use case 1 - algorithms convergence: a.) GA, b.) SA, c.) TLBO, d.) Jaya algorithm.

3.2. Use case 2: Thermosonic gold wire bonding

From the control charts it has been detected that behavior of a smaller group of machines in gold wire (75µm) bonding significantly deviates from the common machine outputs, i.e. the gold ball bond characteristics. Therefore, an experiment was performed to establish an optimal process parameters for the whole group of machines, considering a gold bond formation at the aluminum die pads. Machine M1 was selected as a representative of a major group with common performance, while machine M2 represented a smaller group of machines with deviant outputs. Two major control parameters were considered at three levels: base power (BP_{Au}: 55, 65 and 75 mW), and base force (BF_{Au}: 85, 100 and 115 10^{-2} N). Three responses were observed at the output, with the objective to reach a nominal (target) value: the average ball shear test value (BS; target=270 10^{-2} N), the average ball diameter (D; target=193 µm), and the average ball height (H; target=50 µm) in a device with 42 bonds. Responses were measured for each individual bond to obtain mean values and assess variability needed to compute QLs.

The experiment was based on OA L_9 with five replicates [19] (Table 3). PCA was applied over NQLs, for M1 and M2 separately. The obtained PCs were synthesized into the process performance (γ) using weights from PCA: [0.667; 0.246; 0.087] for M1, and [0.915; 0.066; 0.019] for M2. In the modeling phase, BP ANN models with topology 2-9-1 showed the best performance for both machines (M1: MSE=1.9 10^{-4}, R=0.97; M2: MSE=1.5 10^{-4}, R=0.97). In optimization, all algorithms found the global optimum for M1; GA failed to locate the global solution for M2. GA showed a high dispersion of results obtained by different tunings, while SA results were more homogenous. TLBO and Jaya obtained identical results in multiple runs, showing a high repeatability and robustness. TLBO and Jaya displayed a rapid convergence, slightly faster, in average, than SA; SA converged faster than GA (Table 4). In overall, TLBO and Jaya showed remarkable performance due to full repeatability of results and fast convergence, while GA and SA results were dispersed (caused by different algorithm specific hyper-parameters), and GA get trapped into a local optimum for M2.

Table 3. Use case 2: experimental results, NQLs and process performance measure

	trial	Control parameters		Responses			NQL$_i$ (k) i=1,2,3; k= 1,...,14			γ_k
	no.	BF$_{Au}$	BP$_{Au}$	BS	D	H	NQL$_{BS}$	NQL$_D$	NQL$_H$	*k=1,..,14*
M 1	1	55	85	229.6	182	58.9	0.96	0.89	0.95	0.5263
	2	65	85	260	187	54.8	0.17	0.51	0.38	0.5871
	…	…	…	…	…	…	…	…	…	…
	14	65	115	277.4	201	43.3	0.14	0.52	0.56	0.6362
M 2	1	55	85	232.2	180	58.7	0.99	0.90	0.99	1.0000
	2	0	85	262.4	188	55.8	0.17	0.27	0.56	0.3733
	…	…	…	…	…	…	…	…	…	…
	14	65	115	281.7	199	43.85	0.18	0.21	0.49	0.5769

Table 4. Use case 2: results of four metaheuristic algorithms

	Metaheuristic algorithm	GA	SA	TLBO	Jaya
M1	Objective function range	0.8807÷0.88120	0.8810÷0.88120	0.88120	0.88120
	Optimal parameters range	[85; 99÷100]	[85; 99÷99.5]	[85; 99]	[85; 99]
	Maximal objective	0.88120	0.88120	0.88120	0.88120
	Optimal parameters	[85; 99]	[85; 99]	[85; 99]	[85; 99]
	Iterations to max. objective	8	21	8	15
M2	Objective function range	0.7081÷0.71280	0.7578÷0.75801	0.75801	0.75801
	Optimal parameters range	[85; 95÷100]	[85; 98.5÷99]	[85; 99]	[85; 99]
	Maximal objective	0.71280	0.75801	0.75801	0.75801
	Optimal parameters	[85; 95]	[85, 85]	[85, 85]	[85, 85]
	Iterations to max. objective	710	3	3	3

3.3. Comparison with optimization methods from the literature

Benefits of the suggested evolutionary-based optimization methodology are demonstrated in comparison with the four frequently used methods from the literature: RSM [1], methods proposed by Fung&Kang [3], Liao [5] and Yang *et al.* [6]. The proposed methodology (with any of the four metaheuristics) surpasses the benchmarked methods in both studeis (Table 5), due to: (i) search over a continous space to detect a global solution; (ii) weaknesses of the methods from literature in terms of adressing correlations among responses, developing a process performance in a completely objective manner or a tendency towards local solutions, as discussed in the introduction.

Table 5. Use cases 1 and 2: comparison of results obtained by different optimization methods

Use case	Optimization methodology	RSM [1]	Fung & Kang (2005) [3]	Liao (2006) [5]	Yang et al. (2014) [6]	Proposed method with GA	Proposed method with SA	Proposed method with TLBO	Proposed method with Jaya
1	Maximal objective (process performance)	0.8514	0.9613	0.9286	0.9613	0.96870	0.97054	0.97362	0.97361
	Optimal parameters	[20; 350; 300;45]	[30; 400; 350; 40]	[30; 400; 350; 80]	[30; 400; 350; 40]	[28; 400; 299; 40]	[40; 400; 385; 66]	[23.4; 400; 264; 40]	[23.3; 400; 264.4; 40]
2 – M1	Maximal objective (process performance)	0.6303	0.5637	0.5768	0.5637	0.88120	0.88120	0.88120	0.88120
	Optimal parameters	[65;100]	[65; 115]	[75;115]	[65; 115]	[75; 100]	[85; 99]	[85; 99]	[85; 99]
2 – M2	Maximal objective (process performance)	0.6379	0.6379	0.5876	0.5876	0.71280	0.75801	0.75801	0.75801
	Optimal parameters	[65;100]	[65;100]	[75;115]	[75;115]	[85; 95]	[85, 85]	[85, 85]	[85, 85]

4. Conclusion

Application of the four algorithms, within the proposed methodology, in two wire bonding studies demonstrated superior results of TLBO and Jaya over GA and SA, especially for the solution accuracy. The GA showed the worst performance, due to inability to find a global optimum, high dispersion of the results obtained with different hyper-parameters and slow convergence. The *stochastic uniform* selection with *single* or *two point* crossover gave better results than the other combinations. The SA performed better than GA; the most beneficial SA settings are: initial temperature 100 or 500 with reannealing interval 10, with *Boltzmann* annealing and *Boltzmann* or *fast* temperature function. The TLBO and Jaya algorithms showed remarkable results, demonstrated also in a recent optimization study [20]. Therefore, they could be recommended for future applications. In this work they were applied with 5*n* population size (*n* is the number of design variables) and 2000 iterations, so these setting could be recommended for future applications for medium-size problems. Since TLBO needed a large number of iterations to reach the global solution in the first study, a higher iterations count might be demanded for more complex problems with a bigger number of design variables.

References

[1] Myers RH, Montgomery DC. Response surface methodology: process and product optimization using designed experiments. Wiley, New York, 2002.

[2] Taguchi G. Introduction to quality engineering. Asian Productivity Organization, New York, 1986.

[3] Fung CP, Kang PC. Multi-response optimization in friction properties of PBT composites using Taguchi method and principle component analysis. J Mater Process Technol 2005;17:602–610.

[4] Lin CT, Lin Cj. Multi-response optimization of wire bonding process for evaluating alternative wire material. Microelectronics J. 2020;106:104925.

[5] Liao HC. Multi-response optimization using weighted principal component. Int J Adv Manuf Technol 2006;27:720–725.

[6] Yang YS, Shih CY, Fung RF. Multi-objective optimization of the light guide rod by using the combined Taguchi method and Grey relational approach. J. Intell. Manuf. 2014;25:99-107.

[7] Tsai JT, Chang CC; Chen WP *et al.* Optimal Parameter Design for IC Wire Bonding Process by Using Fuzzy Logic and Taguchi Method. IEEE Access 2016;4:3034-3045.

[8] Tsai TN. A hybrid intelligent approach for optimizing the fine-pitch copper wire bonding process with multiple quality characteristics in IC assembly. J Intell Manuf 2014;25:177–192.

[9] Li TS, Hsu CM. Parameter optimization of sub-35 nm contact-hole fabrication using particle swarm optimization approach. *Expert* Syst Appl 2010;37:878-885.

[10] Hsu CM. Improving the lighting performance of a 3535 packaged hi-power LED using genetic programming, quality loss functions and particle swarm optimization. Appl *Soft* Comput 2012;12:2933.

[11] Ramsay JO, Silverman BW. Functional Data Analysis. Springer, New York, 2005.

[12] Lu M, Wevers K. Grey System Theory and Applications: A Way Forward. J. Grey Syst. 2007;10:47-54.

[13] Sibalija T, Majstorovic V. The advanced multiresponse process optimization – an intelligent and integrated approach. Springer, Switzerland, 2016.

[14] Sivanandam SN, Deepa SN. Introduction to Genetic Algorithms. Springer, Heidelberg, 2008.

[15] Spall J. Introduction to stochastic search and optimisation. John Wiley & Sons, Inc., New Jersey, 2003.

[16] Rao RV, Savsani VJ, Vakharia DP. Teaching–learning-based optimization: a novel method for constrained mechanical design optimization problems. Computer-Aided Design 2011;43(3):303-315.

[17] Rao RV. Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int. J. Ind. Eng. Comput. 2016; 7(1):19-34.

[18] Sibalija T, Majstorovic V, Miljkovic Z. An intelligent approach to robust multi-response process design. Int J Prod Res 2011;49:5079-5097.

[19] Sibalija T, Majstorovic V. Novel Approach to Multi-Response Optimisation for Correlated Responses. FME Transactions 2010;38: 39-48.

[20] Sibalija T, Kumar S, Patel GCM *et al.* A soft computing-based study on WEDM optimization in processing Inconel 625. Neural Comput & Applic 2021; DOI:10.1007/s00521-021-05844-8.