

Strong solution set by neutrosophic logic for weighted multi-objective evolutionary algorithm optimization

Cheung Liu and Rashid Khan

YB University of Science and Technology.

Corresponding Author / E-mail: clliu45@protonmail.com

KEYWORDS : Supply chain management, Optimization ,Analog circuits, Genetic algorithms, Circuit optimization

Abstract— Weight Based Genetic Algorithms (WBGA) are computationally efficient and simple to implement for multi-objective optimization. The solutions obtained in the converged region may not necessarily always yield maximum optimization for all the objective functions involved. But, the combination of the solutions from different iteration may yield optimized values for all the objective functions to a satisfactory level. The paper attempts to find a method which keeps the simplicity and computational efficiency of WBGA intact, but at the same time counters the problem of inferior pareto-optimal solutions. This is done by finding such a combinational set of solutions which yields strong values for all the objectives. The paper proposes neutrosophic logic (NL) as a postprocessor to the outcome of the WBGA. The NL assigns a percentage of truth, false and indeterminant value to the obtained solutions. The proposed postprocessor operation has been demonstrated with hand calculations on a test problem, and a complex practical example. The results obtained as compared to WBGA show the emergence of a superior solution-set and reaches in close agreement with NSGA-II, while maintaining the computational efficiency.

NOMENCLATURE

f_1, f_2	Objective function number 1 and 2
x_1, x_2	Parameter 1 and 2.
$x_{1,i}, x_{2,i}$	Near optimal solutions generated from GA, where, $i=1,2..N$ and i is the iteration.
$f_{1,i}, f_{2,i}$	Evaluated values of objective functions at each near optimal solution and i , is the iteration.
$f(x_1, x_2)_{strong}$	Strong, medium and weak values for the
$f(x_1, x_2)_{medium}$	obtained near optimal solutions
$f(x_1, x_2)_{weak}$	
l_1, l_2	Lower bounds of constraints for the parameters 1 and 2
U_1, U_2	Upper bounds of constraints for the parameters 1 and 2
Iteration 1, Iteration 2, Iteration 3	Are the three iterations picked randomly from the optimal region of the Genetic Algorithm (GA)
T, I, F	Truth, False and Indeterminant values
M	Is the ratio of width to length (W/L) of the transistors
Norm	Ratio of Evaluated value of the objective function from the solutions obtained from the optimal region iteration to the target value

1. Introduction

Consider an interesting practical issue with the WBGA's, which are usually applied to solve the multi-objective optimization problems.

- **Loss of pareto-optimal solutions**, when solution space is uniformly dispersed over a non-convex trade-off surface¹⁻².

Implying, WBGA's solution method does not focus on finding the best possible solution with respect to all the objective functions. More on this can be found in section 2 of this paper.

Almost all the well-known multi-objective optimization algorithms focus on addressing the sub-optimal pareto-solution problem arising due to WBGA or vector evaluated genetic algorithms (VEGA).

This leaves us with two important questions:

- Does there exist a way which actually does not disturb the computational efficiency and ease of implementation of WBGA?
- And yet, search for the optimal solution set without violating the random nature of the GA?

This can be achieved alternatively by combining the solutions obtained from different iterations of the WBGA in the convergence region. The solutions used for such combinational operation have to be strictly from the convergence region. This can be better explained as "fine-tuning" of the results. This can alternatively be understood as a local search.

This paper applies neutrosophic logic (NL) to "fine-tune" the solutions. The NL assumes that each generated solution set holds a certain degree of truth (T), indeterminant (I) and false (F) degree/percentage, with respect to each objective function¹⁰. Necessary NL theory to understand NLPP is explained in section 3 and 4.

Probably it is best to understand NLPP via a set of practical examples, the audience may refer to section 5. The fig.1 gives a macroscopic overview of the operation of NLPP and generation of a combination of a strong solution set from different optimal iterations generated by a typical WBGA.

For now, let's say in the case of an MO, one set of solution can exhibit strong candidature of optimum solution in one of the objective functions. But, at the same time it can be a relatively weak candidate in the case of the other objective function. A practical example of such a situation could be operational research or a CMOS circuit optimization or supply chain management. Where, in the latter case, the set of obtained solutions is the transistor width to length ratios, which decide

the performance characteristics of the designed circuit¹¹. After obtaining the width to length ratios, it is the job of the design engineer to manually choose the best possible solution. As the complexity of the circuit increases, the solution space expands manifold¹¹. So does the number of parameters. These parameters are directly proportional to the number of transistors. And the various trade-off parameters increase the number of objective functions.

Hence, the solution obtained can result into a strong maximized or minimized value in one of the objective functions, while it might vary for the rest of the objective functions and vice versa. Hence, it is safe to assume that there exists a degree of **truth**, **false** and **indeterminance** attached to each solution set. The essence is, one event might be true in one scenario. But, the same event might not be true in another scenario. Each solution set obtained is assigned a set of **truth**, **false** and **indeterminant** values¹⁰. The solution set are divided into STRONG, MEDIUM and WEAK sets.

The need is to find a combination set of solutions, obtained from more than one-iteration, in the near-optimal region to satisfy all the objective functions.

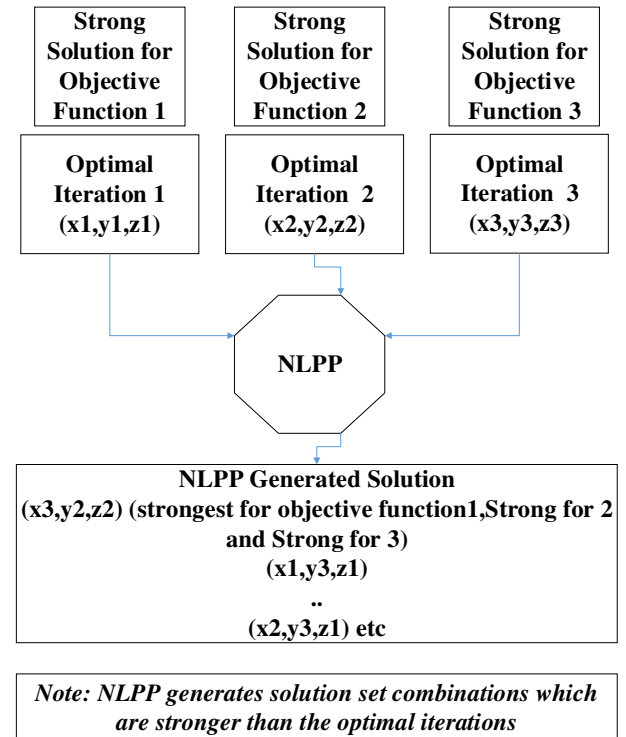


Fig.1. NLPP Operation principle

As shown in the Fig.2 the weighted multi-objective genetic algorithm generates the near optimal solutions in the convergence region. Objective function optimizations are composed of several parameters. However, in the case of

multi-objective optimizations utilizing the GA, the obtained values of parameters from iteration (convergence-region) might not produce the optimum value for all the objective functions. It is commonly observed that the combination of parameters obtained in n-th iteration and m-th iteration might produce an optimum value for **all the objective-functions**.

The proposed NL- based post-processor applied to WBGA searches for the existence of such a combinational solution set, which produces the maximum optimal point for each objective function. By inspection/min-max operation the weak solutions are eliminated. In the second run the solution sets, which exist in the intersection of MEDIUM and WEAK space, are eliminated. Finally only STRONG candidates are left.

To address the problem with WBGA's others have proposed several methods. Let us briefly review their work. NSGA-II was proposed². Where, the focus was to find the pareto-

optimal sets by non-dominated sorting. SPEA-II was proposed, to improve the strength pareto evolutionary algorithm (SPEA)³.

According to the work presented, NSGA-II- crowding distance works in objective space only¹. SPEA-II involves computationally expensive fitness and density calculation⁴⁻⁶. Pareto archived evolution strategy (PAES), is not a population based approach and performance is dependent on the cell sizing⁷. Although, rank-density-based multi-objective genetic algorithm (RGDA) is robust when dealing with numerous objectives, but it is very difficult to implement⁸. Niched Pareto genetic algorithm (NPGA) involves very simple tournament selection⁹. But, it requires an additional parameter to perform tournament selection⁹. Rest of this paper focuses on the NLPP implementation.

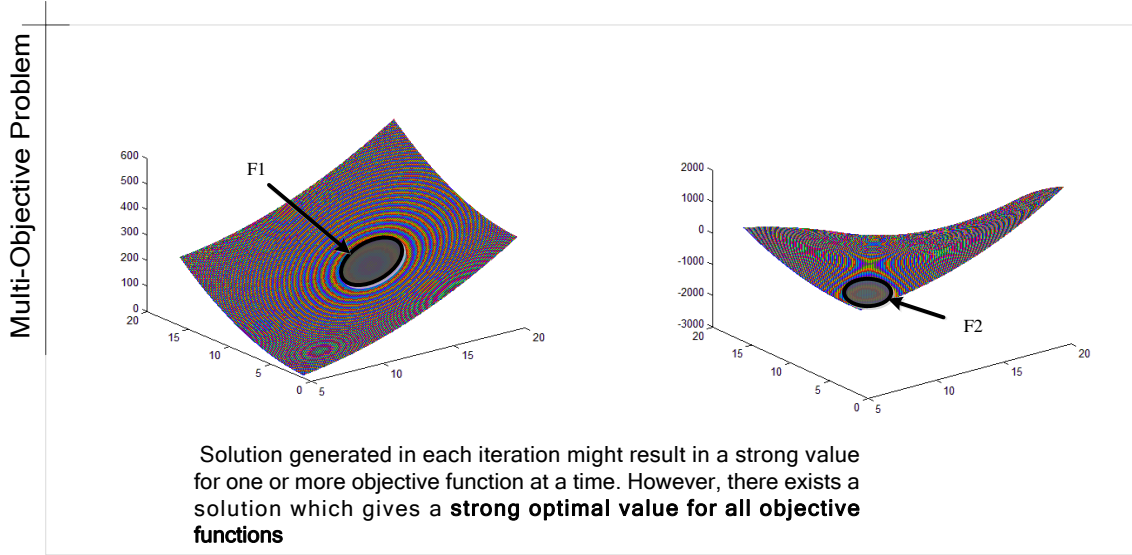


Fig.2. The Multi-objective optimization problem: **the search for common strong solution**.

2. Problem Description

When multi-objective optimization problem is solved using the genetic algorithm, in the convergence region a set of near optimal solutions is generated. The solution set and multi-objective functions can mathematically be expressed as:

$$\begin{aligned} f_1 &= f_1(x_1, x_2) \\ f_2 &= f_2(x_1, x_2) \\ s.t. \\ l_1 &< x_1 < U_1 \\ l_2 &< x_2 < U_2 \end{aligned} \quad (1)$$

Where, L and U are upper and lower bounds of the constraints for parameters x_1 and x_2 ; f_1 and f_2 are objective functions. The solutions generated by GA:

$$\begin{aligned} x_1 &= \{x_{1,1}, x_{1,2}, x_{1,3}, \dots, x_{1,i}\} \\ x_2 &= \{x_{2,1}, x_{2,2}, x_{2,3}, \dots, x_{2,i}\} \end{aligned} \quad (2)$$

Where, 'i' is the iteration number or generation number. While the solution set is (x_1, x_2) and, is represented as:

$$(x_1, x_2)_i = \{(x_{1,1}, x_{2,1}), (x_{1,2}, x_{2,2}), \dots, (x_{1,i}, x_{2,i})\} \quad (3)$$

From the solution sets generated in (equation 3), the objective functions are evaluated as:

$$\{f_1(x_1, x_2)\} = \{f_{1,1}(x_{1,1}, x_{2,1}), f_{1,2}(x_{1,2}, x_{2,2}), \dots, f_{1,i}(x_{1,i}, x_{2,i})\} \quad (4)$$

$$\{f_2(x_1, x_2)\} = \{f_{2,1}(x_{1,1}, x_{2,1}), f_{2,2}(x_{1,2}, x_{2,2}), \dots, f_{2,i}(x_{1,i}, x_{2,i})\} \quad (5)$$

where,

$$x_{1,1}, x_{2,1}, \dots, x_{1,i}, x_{2,i} \in \{x_1, x_2\}$$

The evaluation of objective functions due to strong and weak solution sets can be expressed as:

$$\begin{aligned} \{f_1(x_1, x_2)\}_{\text{strong}} &= f_{1,1}(x_{1,1}, x_{2,1}) & \{f_2(x_1, x_2)\}_{\text{strong}} &= f_{2,1}(x_{1,1}, x_{2,1}) \\ \{f_2(x_1, x_2)\}_{\text{weak}} &= f_{2,1}(x_{1,1}, x_{2,1}) & \{f_1(x_1, x_2)\}_{\text{weak}} &= f_{1,1}(x_{1,1}, x_{2,1}) \end{aligned} \quad (6)$$

While, the solution set $(x_{1,1}, x_{2,1})$ might result in a strong value of f_1 , it need not be so in the case of f_2 . Likewise, other solution sets may result in weak or medium values as compared to the desired optimal values of f_1 and f_2 . Hence, when weight functions are used in multi-objective GA for finding the optimized solution, commonly encountered problem is loss of strong solutions. This has been pointed out in¹. Many authors have suggested improvements in GA e.g. fuzzy goal programming¹², ant colony optimizations¹³⁻¹⁴, TABU search¹⁵, genetic local search for MO combinatorial optimization¹⁶⁻³¹. There, might still be an alternative way to find a common strong solution within the WBGA. One sensible method of addressing the problem is to manually pick the best values from the generated solutions. But, picking the best solution from the huge set of near optimal solutions can be a taxing effort. Let solution $x_{1,n}$ and solution $x_{2,m}$ be the solutions generated out of two different iterations i.e. n -th iteration and m -th iteration. There exists a solution set such that, which gives a strong value for both f_1 and f_2 . This particular strong solution set can exist as a combination of generated values from n -th and m -th iteration.

In this paper the problem is addressed by applying a NL post processor to automate the process of identifying the existence of near optimal strong valued solution sets $x_{1,n}$ and $x_{2,m}$. The neutrosophic logic attaches a truth, indeterminacy and false value to each generated solution. This allows the user to handle the solutions with imprecision in a more superior way. Zadeh, had proposed fuzzy logic to handle imprecision²². However, fuzzy logic cannot answer paradoxes. On the other hand with the help of neutrosophic logic it is possible to successfully answer the paradox¹⁰.

3. Neutrosophic Logic

In 2002, Neutrosophic logic (NL) was proposed by Florentin Samarandache. Later in 2003 it was explained in details and its application to intuitionistic fuzzy logic. A collection of the theories related to NL has been accumulated in¹⁰. A brief overview of the NL is provided in this section, relevant to the

proposed NL post-processor. Definition of Neutrosophic components: T (True), I (indeterminant), F (false) be the standard or non-standard real subsets of a non-standard interval $]-0,1+[$. For more details on non-standard intervals refer to¹⁰. With supremum (sup) of T, I, F be the T_{sup} , I_{sup} , F_{sup} . With: $n_{\text{sup}} = T_{\text{sup}} + I_{\text{sup}} + F_{\text{sup}}$ and $n_{\text{inf}} = T_{\text{inf}} + I_{\text{inf}} + F_{\text{inf}}$. And inf is inferior set. It is noteworthy that T, I and F can exist as subsets. Also, for every case of an event, the T, I, F values are expressed as percentages.

Thus a statement, or an event or a set of events or phenomena or equations which might be true in this world or present situation might be true in this case. But they might have a percentage of truth, indeterminacy, falsehood attached to them in another world, or events or a set of events or phenomena or equations in another situation¹⁰.

4. Proposed NL Post-Processor

The neutrosophic logic post-processor (NLPP) identifies the near optimal solution candidates which offer the “**common - strong**” values in all the objective functions under question. The existence of such a solution can also be concluded from the **Lemma 1** and **2**.

Lemma.1. There exists atleast one set of solutions generated at two different iterations of an evolutionary algorithm in the near optimal zone namely, n -th and m -th iteration such that, the objective functions f_1 and f_2 attain a strong value.

$$\begin{aligned} \{f_1(x_1, x_2)\}_{\text{strong}} &= f_{1,n}(x_{1,n}, x_{2,n}) \\ \{f_2(x_1, x_2)\}_{\text{strong}} &= f_{2,m}(x_{1,m}, x_{2,m}) \end{aligned} \quad (7)$$

Proof: The fig.3 shows the plot of the two objective functions f_1, f_2 whose design variables are (x_1, x_2) . Have their strong optimal levels at iterations n and m respectively.

Lemma.2. There exists, atleast one combination set of solutions

$$\begin{aligned} (x_{1,n}, x_{2,n}) &| \left(\{f_1\}_{\text{Strong}} \quad \{f_2\}_{\text{near_Strong}} \right) \\ & \left(\{f_1\}_{\text{strong}} \quad \{f_2\}_{\text{medium}} \right) \\ (x_{1,m}, x_{2,m}) &| \left(\{f_1\}_{\text{near_Strong}} \quad \{f_2\}_{\text{strong}} \right) \\ & \left(\{f_1\}_{\text{medium}} \quad \{f_2\}_{\text{strong}} \right) \end{aligned} \quad (8)$$

Proof: From fig.3, consider the m -th iteration. The value of objective function f_2 is “strong”, the corresponding value for f_1 is not “weak”, instead it can be classified as near “strong”, or “medium”. Likewise, a similar analogy can be applied to the values at n -th iteration. Where, the f_1 attains a “strong” value while f_2 can be classified as near “strong” or medium, but not weak.

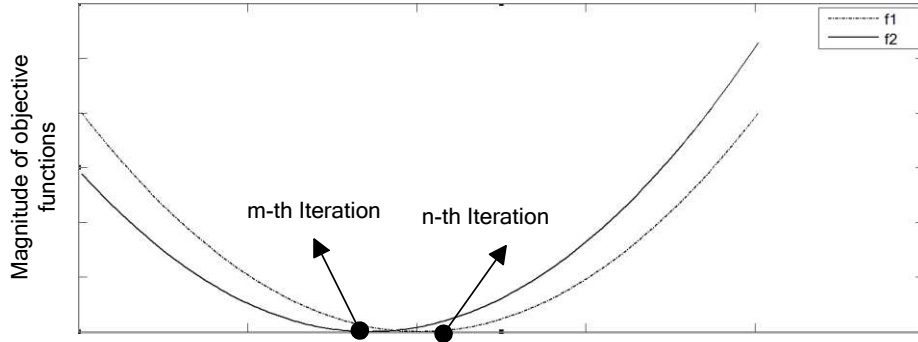


Fig.3. Plot of objective functions, f_1 and f_2 .

This is achieved by applying T, I, F values to the evaluation of objective functions obtained from each solution set as represented in equation (4) and (5). From the NL theory the percentage of T, I, F for objective function f_1 , is obtained from the solution set generated by the first iteration of the converged region and is, expressed as:

$$\left. \begin{aligned} T_{f_1-x_{1,1}-x_{2,1}} &= \{t \pm \delta\}_{f_1-x_{1,1}-x_{2,1}} = \{t - \delta, t, t + \delta\}_{f_1-x_{1,1}-x_{2,1}} \\ F_{f_1-x_{1,1}-x_{2,1}} &= \{f \pm \delta\}_{f_1-x_{1,1}-x_{2,1}} = \{f - \delta, f, f + \delta\}_{f_1-x_{1,1}-x_{2,1}} \\ I_{f_1-x_{1,1}-x_{2,1}} &= \{i \pm \delta\}_{f_1-x_{1,1}-x_{2,1}} = \{i - \delta, i, i + \delta\}_{f_1-x_{1,1}-x_{2,1}} \end{aligned} \right\} \quad (9)$$

Similarly, for objective-function, f_2 :

$$\left. \begin{aligned} T_{f_2-x_{1,1}-x_{2,1}} &= \{t \pm \delta\}_{f_2-x_{1,1}-x_{2,1}} = \{t - \delta, t, t + \delta\}_{f_2-x_{1,1}-x_{2,1}} \\ F_{f_2-x_{1,1}-x_{2,1}} &= \{f \pm \delta\}_{f_2-x_{1,1}-x_{2,1}} = \{f - \delta, f, f + \delta\}_{f_2-x_{1,1}-x_{2,1}} \\ I_{f_2-x_{1,1}-x_{2,1}} &= \{i \pm \delta\}_{f_2-x_{1,1}-x_{2,1}} = \{i - \delta, i, i + \delta\}_{f_2-x_{1,1}-x_{2,1}} \end{aligned} \right\} \quad (10)$$

Let the set of T, I, F values be represented as $\{t\}, \{i\}, \{f\}$. The matrix NL, represents the solution space and it's T, I, F values corresponding to its objective functions.

$$\{x_{1i}, x_{2i}\} = [NL_{f_1}, NL_{f_2}]_i \quad (11)$$

The details of the matrix NL are shown in tabular format in the table 1.

Table 1. NL Matrix

	x_{11}	x_{12}	-	x_{1i}
x_{21}	$[NL_{f_1}, NL_{f_2}]_{11}^{21}$	$[NL_{f_1}, NL_{f_2}]_{12}^{21}$		$[NL_{f_1}, NL_{f_2}]_{1i}^{21}$
x_{22}	$[NL_{f_1}, NL_{f_2}]_{11}^{22}$	$[NL_{f_1}, NL_{f_2}]_{12}^{22}$		$[NL_{f_1}, NL_{f_2}]_{1i}^{22}$
-	-	-	-	-
x_{2i}	$[NL_{f_1}, NL_{f_2}]_{11}^{2i}$	$[NL_{f_1}, NL_{f_2}]_{12}^{2i}$		$[NL_{f_1}, NL_{f_2}]_{1i}^{2i}$

$$NL_{f_1} = \{(t - \delta, t, t + \delta), (i - \delta, i, i + \delta), (f - \delta, f, f + \delta)\}_{f_1}$$

$$NL_{f_2} = \{(t - \delta, t, t + \delta), (i - \delta, i, i + \delta), (f - \delta, f, f + \delta)\}_{f_2}$$

4.1. Determination of Truth, Indeterminant and False values

The first step towards the NLPP is determination of T, I, F values. There are a number of ways of determining the T, I, F values. One way is to determine T, I, F through the objective function itself with respect to the target value. Another method is the use of emphatic values as followed in the case of inconsistent fuzzy rules in²³. The emphatic values in this case were determined with respect to the user experience data. Similar to the followed method, one can determine the percentage of T, I, F values. The method has its own advantages and this area needs further exploration when applying NL to decision making and optimization problems²³. In the examples only the former method of determining the T, I, F values has been used, i.e. using the objective function itself.

5. Practical Examples

In the following section two practical examples have been considered for demonstrating the algorithm. It is worth noting that these examples are for giving the audience an insight only. In the real-world the optimal solution space generated by the GA is huge and the computer code shall do the job of NLPP application.

5.1. Example I: Test Problem

The generalized MO optimization problem has two objective functions and two design parameters to be optimized. The T, I, F values are determined in this example using the objective function equations itself.

The following is obtained for the values obtained from the first iteration:

Table 2 shows chosen sample near-optimal solutions obtained from the GA for demonstration sake for this example. It is given that the target values of optimization for the objective functions is, $f_1=22$ and $f_2=13$.

Table 2 Sample optimal solutions considered for the hand calculation example as a demonstration.

X₁	19	17	16.9
X₂	6	8	9

$$\begin{aligned}
 f_1(X_1, X_2) &= A \times X_1 + B \times \sqrt{X_2} \\
 f_2(X_1, X_2) &= C \times X_2 + 1 / \sqrt{X_1} \times D \times X_2 \\
 A = B = C = D &= 1 \\
 S.T. \rightarrow &20 < X_1 < 5; \\
 &10 < X_2 < 1;
 \end{aligned} \tag{12}$$

5.1.1 Determination of T, I, F values:

Consider the first set of solutions obtained, $\{X_1, X_2\} = (19, 6)$, the obtained values of f_1 and f_2 are 21.44 and 7.3765. The norm value represented by *Norm* is given by the ratio

$$\text{Norm} = \frac{\text{Evaluated value from iteration}}{\text{Target value}}$$

The table 3 enlists the solution methodology by hand calculations.

Table 3. Hand calculations of the Example I

Solution Set (19,6)	$Norm_f1 = \frac{21.44}{7.3765} = 0.9727 \rightarrow Average_Norm = \frac{[Norm_f1 + Norm_f2]}{2}$ $Norm_f2 = \frac{7.37}{13} = 0.5669 \quad Average_Norm = \frac{0.9727 + 0.5669}{2} = 0.7688$ $T_f1 = Norm_f1 = 0.9727$ $I_f1 = Norm_f1 - Average_Norm = 0.2029$ $F_f1 = 1 - T_f1 - I_f1 = 0.1756$ Similarly for f_2 $T_f2 = Norm_f2 = 0.5669$ $I_f2 = Norm_f2 - Average_Norm = 0.2029$ $F_f2 = 1 - T_f2 - I_f2 = 0.2302$
------------------------	---

Similarly the values obtained for all the cases are listed in table 4.

Table 4: T, I, F Values for Example I

Set (x1,x2)	f ₁			f ₂		
	T	I	F	T	I	F
19,6	0.9727	0.2029	0.1756	0.5669	0.2029	0.2302
17,8	0.9	0.067	0.033	0.7646	0.0677	0.167
16,99	0.9045	0.0219	0.0736	0.8607	0.0219	0.1174
19,8	0.9918	0.1176	0.1094	0.7566	0.1176	0.1258
19,9	1	0.1258	0.1258	0.8508	0.1258	0.0234
17,6	0.8836	0.1151	0.0387	0.5735	0.1551	0.3114
17,9	0.9091	0.0245	0.0664	0.8602	0.0245	0.1153
16,96	0.8795	0.1528	0.0323	0.5738	0.1528	0.2734
16,98	0.8967	0.0658	0.0375	0.7651	0.0658	0.1691

On inspection of table 4 one finds that, although solution sets (19,6),(17,8) show strong percentages of truth values, for objective function f₁, however they have somewhat lower percentages of truth values when compared with other candidates in the case of objective function f₂. Solution sets: (19,9), (17,9) lie in the STRONG region. Solution sets (16,9,8),(17,8),(16,9,9),(19,8) are MEDIUM candidates. While, solution sets (19,6),(17,6),(16,9,6) are in the WEAK region in the given problem at hand. It is noteworthy that the T+I+F values for f₂, f₃ and f₄ > 1. This stems from the non-standard analysis concept from where the neutrosophic logic has been derived¹⁰. Hence, it is safe to say that in NL there is no restriction on T, I, F other than the condition that they are subsets of]0,1+[:

$$Sup(T) + Sup(I) + Sup(F) \leq 3^+ \quad (13)$$

5.2. Example II: 3-Stage NMC Operational amplifier optimization problem

Consider fig.4 which shows the 3-stage NMC Op-Amp. The optimization problem considered here has six objective functions, and twenty-four design parameters, represented by W and L. Other details of the assumptions of the problems are immaterial as the main purpose is to visualize the NLPP operation on any complex MO optimization problem. The objective functions are as follows listed in equation (14)-(19).

$$f_1 = \left[k \times \sqrt{\frac{W_1}{L_1}} \times \left[\frac{1}{\frac{W_2}{L_2} + \frac{W_4}{L_4}} \right] \times \frac{1}{W_1} \times k \times \sqrt{\frac{W_6}{L_6}} \times \left[\frac{1}{\frac{W_6}{L_6} + \frac{W_7}{L_7}} \right] \times \frac{1}{W_6} \times k \times \sqrt{\frac{W_{10}}{L_{10}}} \times \left[\frac{1}{\frac{W_{10}}{L_{10}} + \frac{W_{11}}{L_{11}}} \right] \times \frac{1}{W_9} \right] \quad (14)$$

$$f_2 = \left[k \times \sqrt{\frac{W_1}{L_1}} \times \frac{1}{\sqrt{C_L}} \right] \quad (15)$$

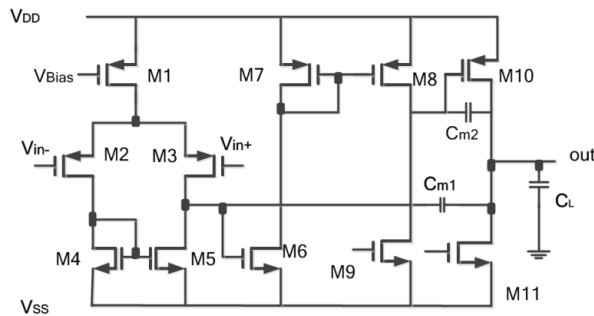


Fig.4. 3-Stage NMC Operational amplifier optimization problem

$$f_3 = \left[k \times \sqrt{\frac{C_c}{2} \cdot \left(\frac{W_1}{L_1} \right) \cdot \left(\frac{1}{\frac{W_2}{L_2}} + \frac{1}{\frac{W_4}{L_4}} \right)} \right] \quad (16)$$

$$f_4 = \left[\frac{k}{2 \times C_L} \times \sqrt{\left(\frac{W_1}{L_1} \right)} \right] \quad (17)$$

$$f_5 = \left[k \times r' \times \frac{\frac{W_1}{L_1}}{\frac{W_3}{L_3}} \times C_L \right] \quad (18)$$

$$f_6 = \left[k \times \frac{r'^2}{1 + 2r'} \times \frac{\frac{W_2}{L_2}}{\frac{W_3}{L_3}} \times C_L \right] \quad (19)$$

Where k, C_c, C_L, r and r' is a constant. The target values are listed in table V and the set of obtained parameters from the GA are listed in Table VI.

Table 5. Target parameters

Objective functions	Range of Target Values
f1	100-170
f2	1.1-1.9
f3	2.9-4.3
f4	1.5-1.9
f5	74-102
f6	46-80

Table 6 sample optimal solutions from GA

Parameters	Converged Iteration1	Converged Iteration2	Converged Iteration3
(W/L)1	89.23/3.10	87.70/2.78	88.20/7.18
(W/L)2,3	32.57/1.78	30.32/1.39	31.42/1.09
(W/L)4,5	51.21/4.23	46.31/3.71	48.31/3.51
(W/L)6	27.80/0.87	20.21/0.38	23.31/0.28
(W/L)7,8	39.8/0.47	36.65/0.66	38.65/0.56
(W/L)9	1.79/1.15	1.66/0.64	1.06/0.45
(W/L)10	80.15/0.882	83.79/1.2	81.79/1.9
(W/L)11	10.2/1.07	11.1/1.25	10.1/1.31
Cm1	70.27	82.049	24.17
Cm2	35.21	46.83	46.83

5.2.1 Solution Approach:

The fig. 5, depicts the solution approach. In order to simplify the problem, the W/L is taken as a ratio in the Table 7. Each W/L ratio is represented by 'M' with a corresponding index number from 1 to 11. It is worth noting that each of the objective function listed as equation (14)-(19) is dependent on the W/L ratios. The solution approach depicted in fig.4 has five important steps in sequence.

First step: The near-optimal solutions obtained from the GA are listed. In this case three independent iterations in the converged region of the GA were considered. These iterations have been named in this section as iteration 1, iteration 2 and iteration 3, for the sake of simplicity of representation. Iteration 1,2 and 3 are not the first iterations of the GA. They are the three iterations, which were chosen from the optimal region of the GA. This has also been explained in the **note of fig 5.

Second step: The ratios M are determined as all the possible combinations of different iterations. To clarify this statement further, M_1 is a ratio of W_1 and L_1 . Hence, if the W_1 value belongs to iteration 1 and L_1 value belongs to iteration 3, then, they are represented as $(M_1)_{1,3}$. Thus, the subscript-suffix (1,3), represents the index of iteration for (W, L). Same follows for other ratios and for $(M_{11})_{3,1}$. As a result of second step, a new set of solution space emerges, combining the best and worst possible solutions of all the converged region solutions.

Third step: involves the evaluation of the six objective functions under consideration equation (14) to (19).

Fourth step: This involves the determination of T, I, F values for the combined solutions of different iteration as explained in the second step. The fourth step has been explained in details in the later part of this paper under the subsection “obtaining T, I, F values”, section 5.2.3. Alternatively the T, I, F values can also be determined by the approach mentioned in the *illustrative example 1*.

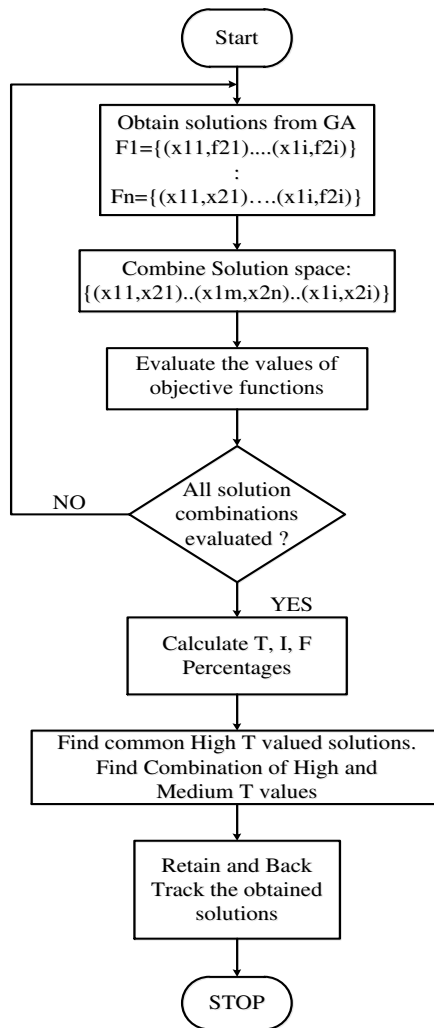
Fifth step: This step is to discard the solutions with strong-false, weak-truth, strong or medium-indeterminant values. The strong-truth, medium-truth and weak-indeterminant values are to be retained.

5.2.2 Additional Details:

The first column in the table 7 represents the ratio of solution obtained from each of the three iterations, e.g. consider the table 6, where, $(W/L)_1 = 89.23/3.10 = 28.7839$, which is listed in the second column as $[(1,1),M1]$. Likewise, $[(1,2),M1] = 89.23/2.78 = 32.0971$, in table 7. One has to remember that the value 89.23 is the value, which has been obtained from the first iteration solution, and the value 2.78 is obtained from the second iteration solution from table 6. In simpler terms, it is nothing but permutations and combinations of W and L variables obtained from different iterations under consideration.

Note: The present example considers only three converged iteration solutions obtained from the GA. From the obtained values of different W/L ratios, the next step is to evaluate the six objective functions under question. The value of $V_{ss} = -V_{DD} = 0.75$ V, since a 0.18- μ m process is considered.

Flow chart



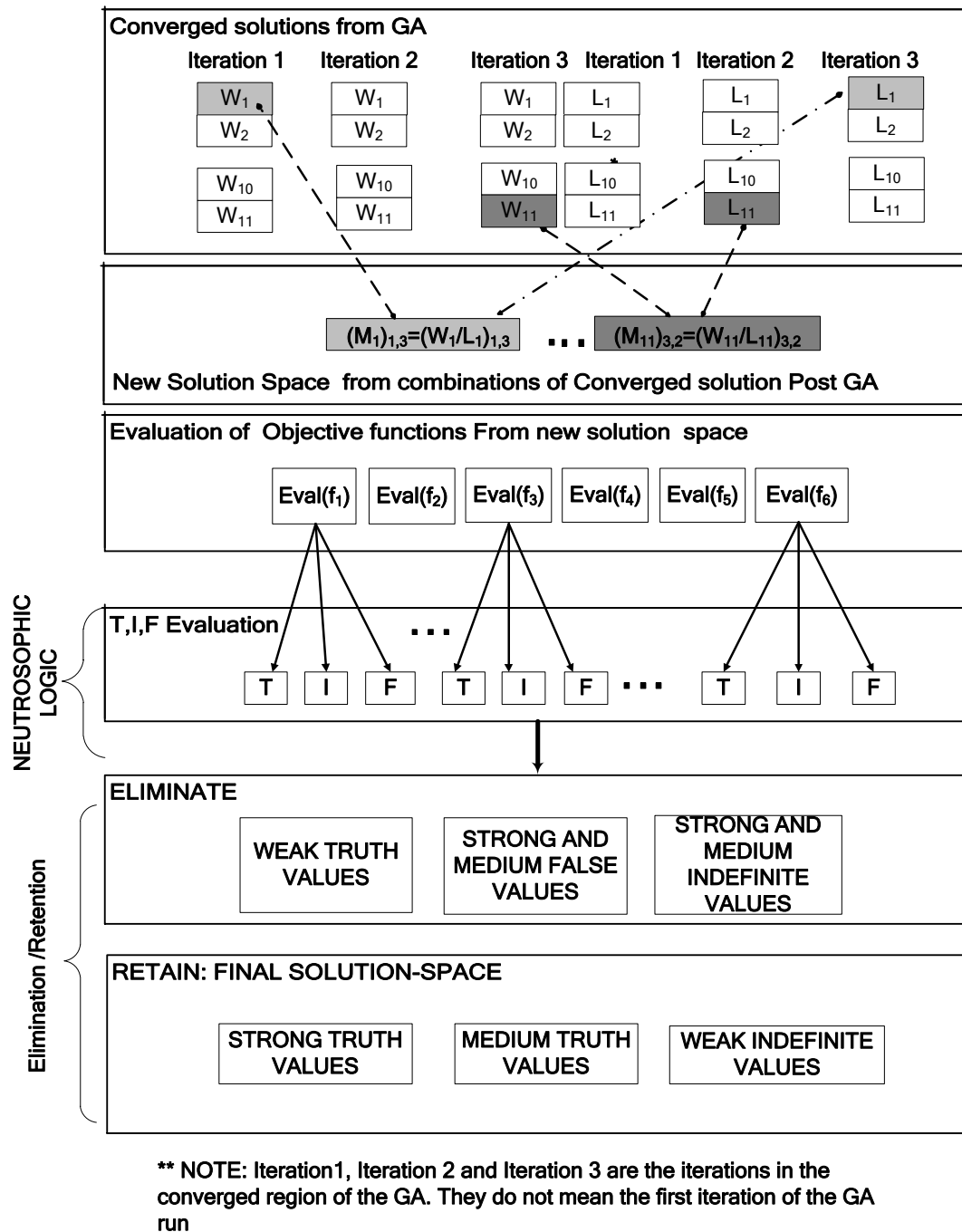


Fig.5. Solution approach for Illustrative Example II.

Table 7. Ratio of solutions with respect to target values (refer to section 5.2.2)

	M1	M2	M4	M6	M7	M9	M10	M11
1,1	28.7839	18.2978	12.106	31.954	84.6809	1.5565	90.8730	9.5327
1,2	32.0971	23.4317	13.8032	73.1579	60.3030	2.7969	89.0556	8.1600
1,3	12.4276	29.8807	14.5897	99.2857	71.0714	2.7538	42.1842	7.7863
2,1	28.2903	17.0337	10.9480	31.9540	84.6809	1.5565	90.8730	9.5327
2,2	31.5468	21.8129	12.4825	53.1842	55.5303	2.5938	69.8250	8.88
2,3	12.2145	27.8165	13.1937	72.1786	65.4464	3.6889	44.1000	8.4733
3,1	28.45516	17.6517	11.4208	26.7931	82.2340	0.9217	92.7324	9.4393
3,2	31.7266	22.6043	13.0216	61.3421	58.5606	1.6563	68.1583	8.0800
3,3	12.2841	28.8257	13.7635	83.2500	69.0179	2.3556	43.0474	7.7099

Table 8. Determination of norm values

	f_1	f_2	f_3	f_4	f_5	f_6
Norm	0.588	0.926	0.9744	0.92	0.875	0.58

As already demonstrated in table 3 of illustrative example the average norm for all the 6 objective functions are calculated as:

$$Average_Norm = \frac{(Norm_f_1 + Norm_f_2 + Norm_f_3 + Norm_f_4 + Norm_f_5 + Norm_f_6)}{6} = 0.8115$$

The T,I,F values are determined keeping the constraint $T+I+F=1$ as follows,

$$\begin{aligned} T_f_1 &= 0.588 \\ I_f_1 &= |Norm_f_1 - Average_Norm| = |0.588 - 0.8115| = 0.2235 \\ F_f_1 &= |1 - T_f_1 - I_f_1| = |1 - 0.588 - 0.2235| = 0.1885 \end{aligned}$$

Table 9. Determination of T,I,F values in Example II

f_1			f_2			f_3			f_4			f_5			f_6		
T	I	F	T	I	F	T	I	F	T	I	F	T	I	F	T	I	F
.58	.223	.118	.926	.114	.040	.974	.162	.13	.92	.108	.028	.875	.063	.061	.58	.231	.1885

Note: The table 9 shows the T,I,F values for one case of combination of iterations only, the rest of the cases can be evaluated by the audience. .

5.2.3 Obtaining T, I, F values:

Consider the target range defined in table 5, the method for obtaining the T, I, F values for objective function have been demonstrated for one case only (listed in table 7 to 8), and the values for rest of the cases are obtained in the same way. By inspection of the target range values from f_1 is 100-170. The values are obtained in table 9 by dividing the resultant by the upper bound of the target value (table 5), in this case, 170. After obtaining the normalized values as demonstrated in table 3, hand calculations of illustrative example I. The T,I,F values have to be determined.

After obtaining T, I, F values for all the cases, the strongest solutions identified have been listed in Table 10. There are still other possible ways of determination of T, I, F values. Perhaps, expert data and machine learning techniques can also be utilized to determine T, I, F values as proposed in

6. Results and Discussions:

In order to evaluate the performance measure of the proposed method, the investigation of the proposed method was done on Matlab environment. The CPU speed was 2.0GHz, Intel i3 processor. The method was

compared with regular NSGA-II algorithm for the given example II. And, example II demonstrates the validity of the NLPP model for complex problems ranging from decision making to VLSI circuits. The table 11 shows the comparison of WBGA along with its refinement after NLPP is applied. Table 12 compares the proposed method with the outcome of NSGA-II algorithm, when applied to example II. The NLPP when applied as a post-processor to WBGA, retains the simplicity of and ease of implementation along with its computational efficiency. It has been observed that the NLPP application to WBGA,

eliminates the need to compromise on the pareto-optimal fronts. A generalized theory on the NLPP for the GA has been proposed and presented. The existence of relative percentage of truth, false and indeterminant state in the neutrosophic logic has been exploited to find the best possible solution candidates. The model has demonstrated the evaluation of the common-strong, common-weak and common-medium solution sets for a multi-objective optimization problem. The validity of the model has been demonstrated by the practical examples. They show the results to be in agreement with the proposed NLPP

Table 10: Solution sets obtained from WBGA

Parameters	Solution1	Solution 2	Solution 3
(W/L)1	28.7839	28.7839	28.7839
(W/L)2,3	18.2978	18.2978	18.2978
(W/L)4,5	12.106	12.106	12.106
(W/L)6	31.954	31.954	31.954
(W/L)7,8	84.6809	84.6809	84.6809
(W/L)9	0.9217	1.5565	1.5565
(W/L)10	90.8730	42.1842	90.8730
(W/L)11	9.5327	9.5327	8.16
Cm1	89.78	74.32	89.245
Cm2	15.62	26.47	46.83

Table 11. Comparison of results of WBGA and NLPP applied to WBGA.

	Near Optimal Iteration Results			NL-PP Post-processing results	
	Iteration1	Iteration2	Iteration3	Strong	Parameters as shown in Table 9
WBGA	28.78	30.02	12.28	30.02	
	18.29	21.81	28.825	18.29	
	12.106	12.48	13.76	12.106	
	31.95	53.18	83.25	31.95	
	84.68	55.53	69.01	84.68	
	1.55	2.59	0.92	0.92	
	90.87	69.82	43.04	90.87	
	9.53	8.88	7.7	8.88	
	87.98	74.32	89.245	89.245	
	15.62	26.47	46.83	46.83	

Table 12. Proposed method compared to NSGA-II

Parameters	NSGA-II		WBGA-NLPP	
Solutions	(W/L)1	30.89	(W/L)1	30.02
	(W/L)2,3	18.01	(W/L)2,3	18.29
	(W/L)4,5	12.07	(W/L)4,5	12.106
	(W/L)6	31.79	(W/L)6	31.95
	(W/L)7,8	84.25	(W/L)7,8	84.68
	(W/L)9	0.98	(W/L)9	0.92
	(W/L)10	91	(W/L)10	90.87
	(W/L)11	9.3	(W/L)11	8.88
	Cm1	89.10	Cm1	89.245
	Cm2	46.27	Cm2	46.83
Computational Time	1008 seconds		703 seconds	
Number of Iterations	25		25	
Population	600		600	

7. Conclusions

In this paper we have presented an NL post processor for WBGA

- **Retains Non-Dominant Solutions:** NLPP can be an alternative to NSGA-II, it retains strong solution sets.
- **Computational Efficiency:** The performance of WBGA-NLPP is very similar to NSGA-II, however it maintains the simplicity and computational efficiency of WBGA when compared to NSGA-II.
- **Flexibility:** NLPP can be extended to a large number of multi-objective optimization problems. Namely- decision making, artificial intelligence, supply chain management, analog IC design etc.
- **Random nature of GA intact:** The research also shows that the random-operation of principle of evolutionary algorithm has not been hampered by the application of NLPP.

ACKNOWLEDGMENTS

This research was financially supported by the Ministry of Science, ICT & Future Planning and National Research Foundation of Korea (NRF) through the Human Resource Training Project for Regional Innovation

References

- [1] Konak, A., Coit, D.W., Smith, A.E., "Multi-objective optimization using genetic algorithms: A tutorial" *Reliability Engineering & System Safety*, vol. 91, no.9, pp. 992-1107, 2006.
- [2] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T., "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II", *IEEE Transactions on Evolutionary Computation*, vol.6, no.2, pp.182-197, 2002.
- [3] Zitzler, E., Laumanns, M., Thiele, L., "SPEA2: Improving the Performance of the Strength Pareto Evolutionary Algorithm", *Technical Report 103, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH) Zurich*, 2001.
- [4] Eskandari, H., Geiger, C.D., Lamont, G.B., "FastPGA: A Dynamic Population Sizing Approach for Solving Expensive Multiobjective Optimization Problems" *Evolutionary Multi-Criterion Optimization. Lecture Notes in Computer Science*, 4403, pp.141-155, 2007.
- [5] Eskandari, H., Geiger, C.D., "A fast Pareto genetic algorithm approach for solving expensive multiobjective optimization problems" *Journal of Heuristics*, vol.14, no.3, pp.203-241, 2008.
- [6] Raza, Z., Vidyarthi, D.P., "A Scheduling Model with Multi-Objective Optimization for Computational Grids using NSGA-II", *International Journal of Applied Evolutionary Computation*, vol.1, no.1, 2010.
- [7] Knowles, J., Corne, D., "The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimisation", *IEEE Proceedings of the congress on evolutionary computation-CEC99*, 1999.
- [8] Lu, H., Yen, G.G., "Rank-density-based multiobjective genetic algorithm and benchmark test function study" *IEEE Trans Evol Comput*, vol.7, no.4, pp.325-343, 2003.
- [9] Horn, J., Nafpliotis, N., Goldberg, D.E., "A niched Pareto genetic algorithm for multiobjective optimization," *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on*, vol., no., pp.82,87 vol.1, 27-29 Jun 1994.
- [10] Smarandache, F. "Multispace and Multistructure. Neutrosophic Transdisciplinarity (100 Collected Papers of Sciences)" vol. IV. *North-European Scientific publishers, Helsinki*, (Chapter: Neutrosophic Logic and Set), 2010.
- [11] Jafaria, A., Bijamib, E., Banac, H.R., & Sadrib, S., "A design automation system for CMOS analog integrated circuits using New Hybrid Shuffled Frog Leaping Algorithm", *Microelectronics Journal*, vol.43, no.11, pp.908-915, 2012.
- [12] Rai, R., Kameshwaran, S., & Tiwari, M.K., "Machine-tool selection and operation allocation in FMS: solving a fuzzy goal-programming model using a genetic algorithm" *International Journal of Production Research*, vol.40, no.3, pp. 641-665, 2002.
- [13] Lee, Z.J., Su, S.F., Chuang, C.C., Liu, K.H., "Genetic algorithm with ant colony optimization (GA-ACO) for multiple sequence alignment", *Applied Soft Computing*, vol.8, no.1, pp.55-78, 2008.
- [14] Garai, G., Chaudhuri, B.B., "A novel hybrid genetic algorithm with Tabu search for optimizing multi-dimensional functions and point pattern recognition" *Information Sciences*, 221, pp.28-48, 2013.
- [15] Gallego, R.A., Romero, R., Monticelli, A.J., "Tabu search algorithm for network synthesis", *IEEE trans Power Systems*, vol.15, no.2, pp.490-495, 2000.
- [16] Jaskiewicz, A., "Genetic local search for multi-objective combinatorial optimization" *European Journal of Operational Research* vol.137, no.1, pp.50-71, 2002.
- [17] Jaskiewicz, A., Tuytens, D., (2006). Metaheuristics in multiple objective optimization. *European Journal of Operational Research* 169(3), 873-874.

- [18] Jaszkiewicz, A., (2004). On the computational efficiency of multiple objective metaheuristics. The knapsack problem case study. *European Journal of Operational Research* 158(2), 418-433.
- [19] Calvete, H.I., Gale, C., Iranzo J.A., (2013). An efficient evolutionary algorithm for the ring star problem. *European Journal of Operational Research*, 231(1) 22-33.
- [20] Ke, L., Zhang, Q., Battiti, R. "MOEA/D-ACO: A Multiobjective Evolutionary Algorithm Using Decomposition and Ant Colony", *IEEE Trans Cybernetics*, 99, pp.1-15, 2013.
- [21] Bandhopadhyay, S., "Multiobjective Simulated Annealing for Fuzzy Clustering With Stability and Validity", *IEEE Trans Systems, Man, and Cybernetics: Part C*, vol.41, no.5, pp.682-691, 2011.
- [22] Zadeh, L.A., "Fuzzy sets", *Information and control*, vol.8, no.3, pp.338-353, 1965.
- [23] Shim, H., Lee, J., Lee, F.Y. and Jun, B.H., "Optimal design of frequency selective surface by genetic algorithm" *International Journal of Precision Engineering and Manufacturing*, vol. 11, no. 5, pp. 725-732, October 2010.
- [24] Gu, D.S., Kim, Y.C., Lee, J.M., Choi, B.K. "Optimum design of simple rotor system supported by journal bearing using enhanced genetic algorithm" *International Journal of Precision Engineering and Manufacturing*, vol. 14, no.9, pp. 1583-1589, September 2013.
- [25] Tzeng, C.J., Chen, R.Y. "Optimization of electric discharge machining process using the response surface methodology and genetic algorithm approach" *International Journal of Precision Engineering and Manufacturing*, vol. 14, no. 5, pp. 709-717, May 2013.
- [26] Lin, C.W. "Simultaneous optimal design of parameters and tolerance of bearing locations for high-speed machine tools using a genetic algorithm and Monte Carlo simulation method" *International Journal of Precision Engineering and Manufacturing*, vol. 13, no. 11, pp. 1983-1988, Nov. 2012.
- [27] Park, J.Y., Han, S.Y. "Swarm intelligence topology optimization based on artificial bee colony algorithm" *International Journal of Precision Engineering and Manufacturing*, vol. 14, no. 1, pp 115-121, January 2013.
- [28] Su, J.C.P., Chu, C.H., Wang, Y.T. "A decision support system to estimate the carbon emission and cost of product designs" *International Journal of Precision Engineering and Manufacturing*, vol. 13, no.7, pp. 1037-1045, July 2012.
- [29] Park, H.S., Nguyen, T.T., "Optimization of roll forming process with evolutionary algorithm for green product", *International Journal of Precision Engineering and Manufacturing*, vol. 14, no. 12, pp. 2127-2135, Dec. 2013.
- [30] Cho, H.K., "Design optimization of laminated composite plates with static and dynamic considerations in hygrothermal environments" *International Journal of Precision Engineering and Manufacturing*, vol. 14, no. 8, pp. 1387-1394, Aug. 2013.
- [31] Hossain M.A.M., Hasan M.T., Hong, S.T., Miles M., Cho H.H., Han, H.N., "Failure behaviours of friction stir spot welded joints of dissimilar ferrous alloys under quasi-static shear loads", *International Journal of Materials and Product Technology*, vol.48, no.1, pp.179-193, Feb 2014.
- [32] Chauhan, N.C., Kartikeyan, M.-V., Mittal, A., "CAD of RF Windows Using Multiobjective Particle Swarm Optimization," *IEEE Transactions on Plasma Science*, , vol.37, no.6, pp.1104,1109, June 2009.
- [33] Kumar A., and Kartikeyan, M.V., "A Comparative study of Microstrip Bandstop Filters Loaded with Various Dumbbell-Shaped Defected Ground Structures", *International Journal of Microwave and Optical Technology*, vol.8, no.2, pp.53-60, March-2013.