Immune Multi-Target Design of System Compensators Placement Schedule

T.Y. Hwang[1] T.L. Huang[1] Y.T. Hsiao[1] C.H. Huang[2] P.H. Kuo[1] C.C. Huang[2] Y.N. Lien[2]

[1] Computer Science Department, National Taipei University of Education

[2] Computer Science Department, National Chengchi University

Email:tsongliang@tea.ntue.edu.tw

***Abstract*–This work proposes a two-stage immune algorithm that embeds the compromise programming to perform multi-objective optimal compensator placement. A new problem formulation model that involves fuzzy sets to reflect the imprecise nature of objectives and incorporates multiple planning requirements is presented. The proposed approach finds a set of non-inferior (Pareto) solutions rather than any single aggregated optimal solution. Additionally, this developed approach eliminates the need for any user-defined weight factor to aggregate all objectives. Comparative studies are conducted on an actual system with encouraging results, demonstrating the effectiveness of the proposed approach.**

***Keyword*—Compensator placement, immune algorithm, non-inferior set, compromise programming.**

1. Introduction

TYPICAL Distribution systems operate in a radial configuration; they are supplied from substations and feed to distribution transformers The spatial density of the load is high in urban areas, where underground cables and large transformers are used, but lower in mixed and rural areas, where overhead lines and smaller transformer units are used. Numerous shunt capacitors are installed along distribution feeders to compensate for reactive power to regulate the voltage; reduce energy; correct the power factor, and release system capacity, for both urban and rural areas. The general capacitor placement problem is to locate and determine the sizes of capacitors to be installed at the nodes of a radial distribution system under various loading conditions.

Various attempts form different perspectives have been made to solve the capacitor placement problem. For instance, the problem has been formulated as a mixed integer programming problem in which power flows and voltage constraints are applied [1].Heuristic approaches have also been presented to identify sensitive nodes from the strengths of the effects on system losses and, then, optimizing the net savings of system losses [2]. An equivalent circuit of a lateral branch has been used to simplify the distribution loss analysis. In so doing, capacitor operating strategies were elucidated according to the reactive load duration curve and the sensitivity index [3].Optimal capacitor planning has been implemented based on the fuzzy algorithm in practical distribution systems [4]. A solution technique based on simulated annealing (SA) has been developed; implemented in a software package, and tested on a real distribution system with 69 buses [5,6]. The Tabu Search technique has been applied to determine the optimal capacitor planning in the distribution system used in [6], and the results of the TS compared with those of the SA. Genetic algorithms (GA) have been used to determine the optimal selection of capacitors [8,9].In [9], Gas were implemented to optimize the selection of capacitors, but the objective function considered only the cost of the capacitors and the power losses, without imposing operation constraints.

Notably, most of these approaches treat the capacitor placement problem as a single objective problem. However, in recent years, customers have made strong demands of electrical utility companies [10]. Various problems have multiple and conflicting objectives (such as simultaneously minimizing the cost of fabrication and maximizing the reliability of the system), which make the optimization problem interesting to solve. No single solution is an optimal solution to a problem with multiple conflicting objectives, so a multi-objective optimization problem has a number of trade-off optimal solutions. Classical optimization methods can at best find one solution in one simulation run, so such methods inconvenient when used to solve multi-objective optimization problems.

In light of the above, this study formulates the capacitor placement problem as a multiple objective problem, including operational requirements The problem formulation presented herein considers four objectives-minimizing the cost of installing capacitors, real power loss and deviation of the bus voltage, and maximizing the capacity margin of the feeders and the transformer. The imprecise nature of each objective function is incorporated by modeling these objective functions using fuzzy sets. This work also presents a two-staged immune algorithm to solve the constrained multiple objective problem.

The rest of this article is organized as follows. Section 2 describes a novel formulation of the capacitor placement problem. Section 3 introduces the immune algorithm for solving optimal problems. Section 4 briefly reviews multi-objective optimization, and develops the two-stage immune algorithm for multi-objective programming. Section 5 describes how to apply the proposed method to the capacitor placement problem. Section 6 then demonstrates the effectiveness of the solution algorithm when applied to power distribution systems. Section 7 draws conclusions.

2. Problem Formulation

This study formulates the capacitor allocation problem to determine the locations and size of capacitors to be installed in the nodes of a radial distribution system under various loading conditions. The problem formulation considers four objective functions, to minimize the total cost of capacitors to be installed, the energy loss and the deviation of bus voltage, and to maximize the system security margin of transformer capacity. These objective functions are formulated as fuzzy sets to incorporate their imprecise nature. A fuzzy set is typically represented by a membership function μf_i(x)for the ith objective function f_i(x). A higher membership function implies greater satisfaction with the solution. The membership function usually consists of lower and upper boundary values and is strictly monotonically decreasing and continuous. Without loss of generality, a membership function of a minimizing problem can be defined by

$$u_{fi}(x)=\begin{cases}1 \quad or \rightarrow 1, & if,\ f_i(x)<f_i^{\min} \\ h_i(f_i(x)), & if,\ f_i^{\min}\le f_i(x)\le f_i^{\max} \\ 0 \quad or \rightarrow 0, & if,\ f_i^{\max}<f_i(x)\end{cases} \qquad (1)$$

The lower and upper bounds, f_i^{min} (x), f_i^{max} (x) on each objective function under given constraints are established to elicit a membership function μf_i (x) *for each objective* function, f_i (x).

Then, a strictly monotonically decreasing and continuous function h_i (f_i (x)), which can be linear or non-linear, is determined. In the following, objective function with fuzzy models are introduced to formulate the capacitor placement problem.

2.1 Minimizing capacitor construction expenditure

The cost of capacitors includes two terms The first term represents the purchase cost while the second represents the installment and maintenance cost.

$$\min f_c = \sum_{i\in\Psi} \frac{1}{y}[k_p(q)+k_m(q)]a_i \qquad (2)$$

Where α_i is a 0-1 decision variable: α_i= 1 if ith bus is selected for capacitor installation; otherwise α_i= 0; Ψ represents the set of candidate locations of buses to be considered for capacitor injection; y denotes the life time (year) of the capacitors; K_p represents the purchased cost of capacitors of capacitance q; K_m denotes the fixed installment and maintenance cost. Notably, the cost function f_c is a non-differentiable step like function since the capacitors are grouped by the specific size. Figure 1 plots the fuzzy membership function f_c of the cost where f_{cmax} represents the cost of the maximum allowable number of capacitors to be installed in the system of interest.

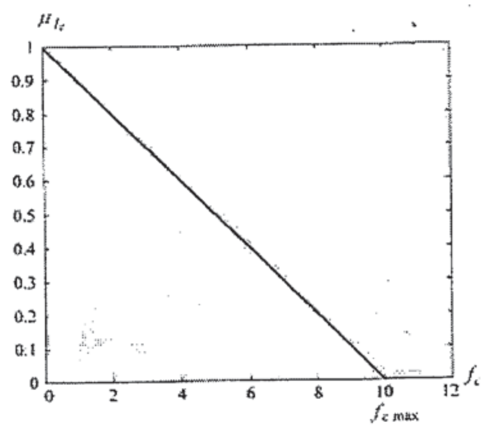

Fig. 1. Fuzzy membership function of the cost, f_c

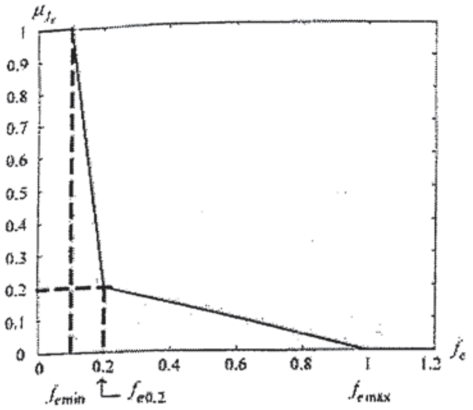

Fig. 2. Fuzzy membership function of the power loss

2.2 Minimizing real power loss

The total cost of the real power loss from line branches, is defined as,

$$\min f_e = \sum_{j=1}^{n_t} k_{ej} t_j p_{loss,j} \tag{3}$$

Where n_t represents the total number of load levels; k_{ej} represents the cost of power under load j; t_j represents the duration of the application of load j, and p_{lossj} is the total real power loss of the considered system under load j .Figure 2 displays the fuzzy membership function of power loss where f_{emax} represents the real power lost without capacitor compensation; $f_{e0.2}$, is 80% of the f_{emax} and f_{emin} is the expected real power loss in the considering system.

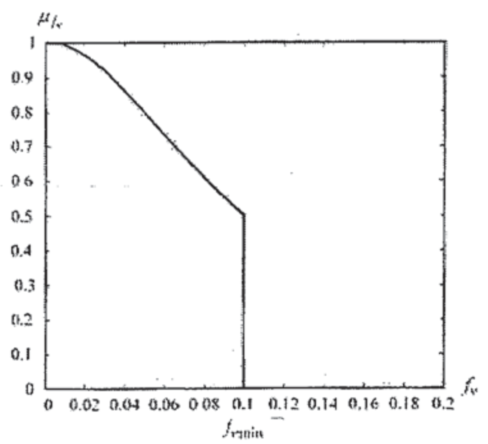

Fig. 3.Fuzzy membership function of the deviation of the bus voltage

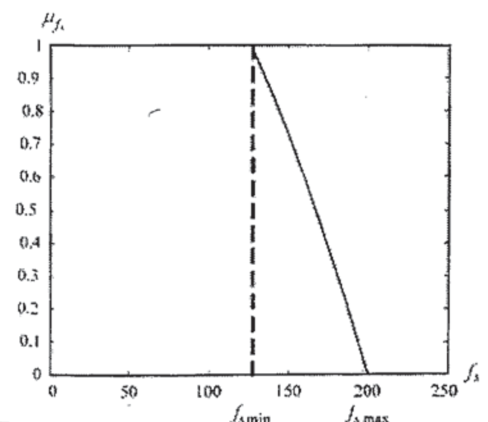

Fig. 4.Fuzzy membership function of the security margin of feeders and transformers

2.3 Minimizing Deviation of Bus Voltage

The bus voltage, an important index, characterizes the security and power quality of a distribution system. Accordingly, an index is defined that quantifies the deficiency in the system caused by the bus voltage.

$$\min f_v = \max_i \left| v_i - v_i^{Rated} \right|, \; i=1, 2, 3, \ldots, n_b \tag{4}$$

Where n_b is the total number of buses; v_i and v_i^{Rated} denote the real and rated voltages of bus I, respectively, and f_v corresponds to a higher quality voltage profile and better system security. Figure 3 plots the fuzzy membership function of the deviation of the bus voltage where f_{vmax} is the maximum allowable deviation of bus voltage.

2.4 Maximizing the Security Margin of Feeders and Transformers

A simple index to assess the system security is the capacity margin of feeders and transformers The security index is defined as follows

$$\min f_s = 1 - \min_i \left| \frac{I^2_{iRate} - I^2_{iLoad}}{I^2_{iRate}} \right|, \; i = 1, 2, 3, \ldots, n_h \tag{5}$$

where I_{iLoad} and I_{iRetad} are the current flow and the rate flow of branch (transformer) i , respectively; n_h; represents the total number of branches (transformers), and f_s implies more secure system capacity. Figure 4 plots the fuzzy membership function of the of feeders (transformers) where f_{smax} denotes the rating of the considering feeders (transformers) and f_{smim} is the maximum expected security margin.

3. Immune Algorithm

The immune system is a natural, fast and effective defense mechanism for a host against infection. It includes a complex set of cells and molecules that protect our bodies against infection. Our bodies are under constant attack by antigens that can stimulate the adaptive immune system. Antigens might be foreign, such as surface molecules present on pathogens, or self-antigens, which are composed of cells or molecules of our own bodies [11,12].

The immune system has a fundamental ability to produced new types of antibody or find the best-fitting antibody to attack an invading antigen. The immune system produces very many antibodies against innumerable, unknown antigen, by trial and error. The diversity of the immune system can be mathematically formulated as a multi-objective function optimization problem, with multiple solutions rather than single solution, to elucidate the diversity of antibodies that is essential to adaptability against foreign viruses and bacteria in the environment. The presented algorithm uses parallel search vectors to find multiple solutions. The index of diversity is introduced and multiple solution vectors maintained as a memory cell mechanism in the immune system. The antigen can be regarded as a problem to be solved and the antibody a solution vector that best fits to solve the problem. The immune system in a higher mammal eliminates antigens by the genetic evolution of a lymphocyte population that can produce antibodies. Genes produce numerous types of antibody bf trial and error because the type of antigen is not known a priori. The best antibody among numerous candidates is selected to destroy the antigen by bio-chemical pattern matching between the antigen and the antibody. Accordingly, the immune system can be regarded as a combinatorial optimization process, which is to select the type of antibody (solution vector) from among a great many solution candidates, that best fits the antigen.

A measure of diversity of antibodies produced from a lymphocyte population is required and must be defined. Lymphocytes recognize an invading antigen and produce the antibodies to eliminate the antigen. Notably, the antigen and antibody in the immune algorithm are represented as objective and the feasible solution, respectively, in the optimization problem. Figure 5 depicts a model of a lymphocyte population consisting of antibodies, where j is the candidate solution. For the N antigens (antibodies) with L genes in the pool, according to information theory, the entropy $E_j(N)$ of the jth gene is defined as [11,12]

$$H_j(N) = -\sum_{i=1}^{N} p_{i,j} \log p_{i,j} \tag{6}$$

where p_{ij} represents the probability that locus j is allele i. If all alleles at the jth gene are the same, then the entropy of the jth gene equals zero. The mean of the informative entropy in a lymphocyte population is represented by

$$H(N) = \frac{1}{L}\sum_{j=1}^{L} H_j(N) \tag{7}$$

Where $H(N)$ denotes the mean of the informative entropy for all antibodies and L is the size of the genes in an antibody. This entropy specifies the diversity of the lymphocyte population. Two expressions for affinity are considered in the presented approach. One $(Ab)_{vw}$, is used to determine the diversity between two antibody v and w and can be represented as,

$$(Ab)_{vw} = \frac{1}{1+H(2)} \tag{8}$$

where $H(2)$ quantifies the diversity between two antibodies, according to E_q .(7)for N =2.For $H(2)$=0, the genes of the two antibodies are identical. The other affinity $(Ag)i$ is that between antigen A_g and antibody A_b and is defined by

$$(A_g)_i = \mu_{f_i}(A_{bi}) - \sum_{j=1}^{N_c} \mu_{g_j}(A_{bi}) \quad i = 1,2,\cdots,N_o \tag{9}$$

where $\mu f_i\ (A_{bi})$ is the value of the membership function for antibody A_{bi} on objective l; $\sum_{j=1}^{N_c} \mu_{g_j}(A_{bi})$re the values of the membership function with all applied constraints for antibody A_{bi} and N_c and N_0 are the numbers of constraints and objectives, respectively. The antibody is perfectly matched with the antigen when the affinity $(Ag)_i$ equals one. Antibodies that have high affinities toward an antigen are selected to proliferate, while antibodies with low concentrations are suppressed. The concentration C_v of each antibody can be defined as

$$c_v = \frac{1}{N_o}\sum_{w=1}^{N_o} ac_{v,w} \tag{10}$$

with

$$ac_{v,w} = \begin{cases} 1 & (Ag)_i \geq \varepsilon \\ 0 & \text{otherwise} \end{cases}$$

(11)

where ε is a preset threshold. If C_v (v=1,2,…,N_0) is greater than a given threshold δ_c, then this antibody becomes a memory antibody; else, it is suppressed. The goal of this step is to eliminate surplus solution candidates.

From the schema of the natural immune system, the mathematical optimization framework can be modeled as an algorithm, realized by the following steps.

Step 1 : Identify the optimization problem;

Step 2 : Generate random antibodies (candidate solutions);

Step 3 : Calculate the affinity $(Ag)_i$ between the antibody and antigen according to Eq (9);

Step 4 : Determine the concentration C_v of each antibody in the repertoire according to Eq (10):

Step 5 : If C_v exceeds a given threshold δ_c, then proceed to the next step; else, proceed to step 8.

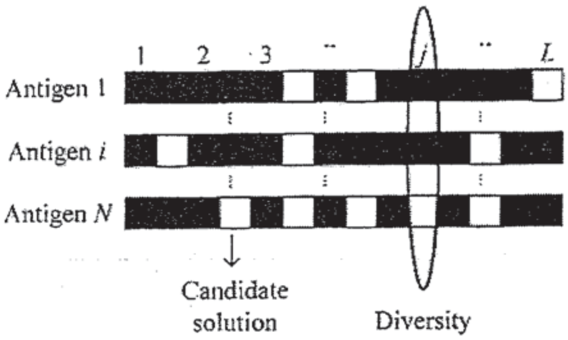

Fig. 5. Informative entropy of antigens.

Step 6 : Calculate the affinity $(Ab)_{vw}$ using Eq.(8) for each antibody v =1,2,…,N_0 to the antibody w , which has the highest concentration;

Step 7 : If all affinities $(Ab)_{vw}$ exceed a threshold δ_a, then this antibody becomes a memory antibody; proceed to step 10; else, proceed to step 8.

Step 8 : Suppress (eliminate) antibodies with low concentration (affinity).

Step 9 : Generate new antibodies using genetic variation operators, such as crossover and mutation, to replace the antibodies eliminated in the previous steps.

Step 10 : Repeat steps 3 to 9 until a certain stopping criterion is fulfilled.

Notably, in the above immune algorithm, the number of generated antibodies and the number of iterations can be experimentally determined. The rate of the crossover and mutation are also determined on a trial basis.

4. Multi-Objective Optimization

A multiple objective problem can be considered to have the following form.

$$\text{Min } f_i(x) \quad i=1,2,\ldots,N_0 \tag{12}$$

Subject to

$$g_j(x)=0, j=1,2,\ldots,N_{cg} \tag{13}$$

$$h_k(x)\leq 0, k=1,2,\ldots,N_{ch} \tag{14}$$

where $f(x)$ are N_0 distinct objective functions of the decision vector x, and $g(x)$=0and h(x)≤0 are constraints. In most cases, the objective functions of the multi-objective optimization problem are in conflict with one another, so no objective function can be improved upon without worsening at least one of the other objective function. This concept is known as Pareto optimality (or non-inferior solutions, or non-dominated solutions, alternative solutions) [13,14].

Definition:

The *feasible region* ,Ω, in the decision vector space X is the set of all decision vectors x that satisfy the constraints, such that

$$\Omega = \{\, x \mid g(x)=0, h(x)\leq 0 \,\} \tag{15}$$

The *feasible region,* Λ, in the objective function space F is the mage of f in the feasible region Ω in the decision vector space:

$$\Lambda = \{\, f \mid f=f(x), x\in\Omega \,\} \tag{16}$$

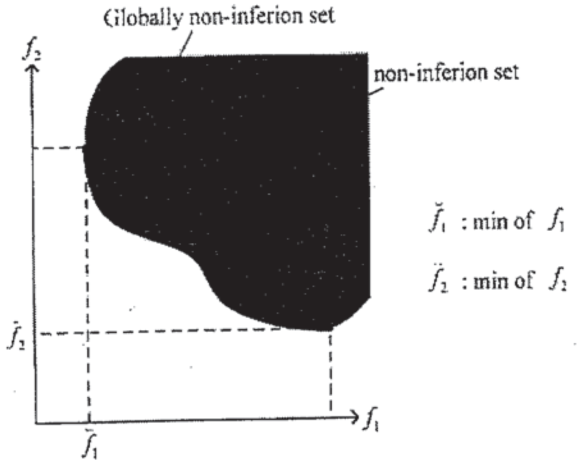

Fig. 6.Global non-inferior solutions to two-objective optimization problem.

A point $\hat{x} \in \Omega$ is a *local non-inferior point* if and only if for some neighborhood of $\hat{x}$, there does not exist Δx such that $(\hat{x}+\Delta x) \in \Omega$ and,

$$f_i(x+\Delta x) \le f_i(\hat{x}),\ i = 1, 2, \ldots, N_o \tag{17}$$

$$f_j(x+\Delta x) < f_j(\hat{x}),\ \text{for some } j \in \{1, 2, \ldots, N_o\} \tag{18}$$

A point $\hat{x} \in \Omega$ is a *local non-inferior point* if there no other point $x \in \Omega$ exists such that

$$f_i(x) \le f_i(\hat{x}),\ i = 1, 2, \ldots, N_o \tag{19}$$

$$f_j(x) < f_j(\hat{x}),\ \text{for some } j \in \{1, 2, \ldots, N_o\} \tag{20}$$

Restated, $\hat{x}$ is a local non-inferior point in a neighborhood $N(\hat{x}, \varepsilon)$, such th at for any other point $\hat{x} \in N(\hat{x}, \varepsilon)$,at least one component to f exceeds its value at $\hat{x}$ or $f_i(x) > f_i(\hat{x})$) , i=1,2,...N_0. A global non-inferior solution of the multi-objective problem is one for which any improvement of one for which any improvement of one objective function can be achieved only at the expense of at least one of the other objectives. In multi-objective optimization, as opposed to single-objective optimization, an unambiguous optimal solution may not exist. Characteristic of multi-objective optimization problem is a very large set of acceptable solutions that are superior to the tested solutions in search space when all objectives are considered. They are simultaneously not optimal with respect to any single objective. These solutions are known as the non-inferior solutions. The rest of the solutions are referred to as inferior solutions. Figure 6 plots the global non-inferior solutions for a two-objective optimization problem. None of the solutions in a non-inferior set is absolutely better than any other, so any one of them is acceptable. The choice of one particular solution depends on the features of the problem and a number of related factors.

The notion of non-inferiority is only the first step toward solving a multi-objective problem. Compromise programming is also necessary to find non-inferior alternatives. Compromise programming has been described elsewhere [11,15]. This study presents a two-stage immune algorithm embedded the compromise program to solve multi-objective problems.

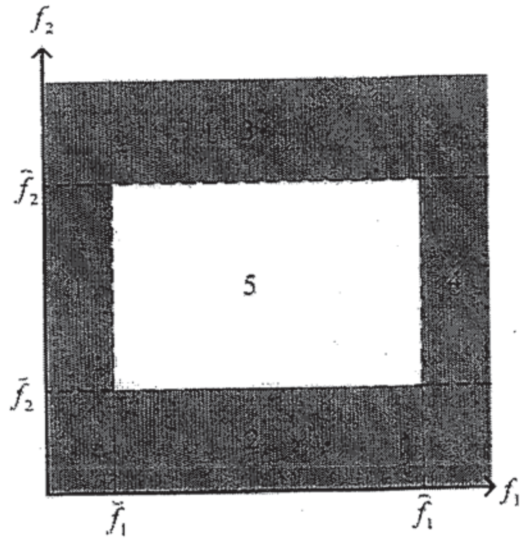

Fig. 7. Decision region on a two-objective space.

Stage 1:Build decision region

Firstly, the multi-objective optimization problem is transformed to a single objective optimization problem by selecting the kth objective as the primary objective function in turns k=1,2,...N_0 and converting the other objectives to constraints with individual maximum allowable values $\bar{f}_i$ where i = 1,2,...,N_0 and $i \neq k$. Then, the resulting single-objective optimization problem is solved as follows.

$$\min f_i(x) \tag{21}$$

such that

$$f_i(x) \leqq \breve{f}_i,\ i=1,2,\ldots,N_o \text{ and } i \neq k \tag{22}$$

$$x \in \Omega \tag{23}$$

$$g(x)=0 \tag{24}$$

$$h(x) \leqq 0 \tag{25}$$

In solving the above single objective optimization problem by turns $k=1,2,\ldots,N_0$,

$$\breve{f}_k = f_k(\bar{x}),\ k=1,2,\ldots,N_o \tag{26}$$

where $\breve{f}_k$ represents the ideal value of the single objective k and denotes $\hat{f}_i$ worst value of the objective i .For illustration, Fig. 7 explains the decision region in a two-objective space. The decision region is bounded by the ideal and worst values of each objective. Figure 7 demonstrates that no optimal solution exists in areas 1 and 2. Areas 3 and 4 have worse solutions. Area 5 is the only decision region in which non-inferior solutions can be found in the second stage. In general, for multi-objective problems, a solution $\hat{x}$ such that $\breve{f} = f_k(\hat{x})$ does not exist for all $k \in \{1,2,\ldots,N_0\}$. Restated, the ideal values (unattainable best solutions) are used to determine the search direction for solving a multi-objective problem, and the hypothetical worst values are treated as the bottom boundary of the solution space. Notably, the decision region is not bound by constraints but has reasonable limits.

Stage 2: Search for the set of the non-dominant solutions.

In this stage, the non-inferior set for all objectives is obtained by compromise programming. Compromise programming finds the best compromise with respect to all the objectives by computing a normalized Euclidean distance measure. (The best compromise is a solution that is "closest" to the ideal solution and lies on the non-inferior frontier.)

$$D=\sum_{i=1}^{N_o}\frac{f_i(x)-\breve{f}_i}{\hat{f}_i-\breve{f}_i} \tag{27}$$

This normalized Euclideam distance is used to evaluate how close the computed non-inferior solution is to the Pareto front. A smaller D indicates the current computed non-inferior solution is closer to the Pareto front. For a multi-objective problem, the ideal value of each objective (from stage 1) and the maximum allowable value of each individual objective where I and $k = 1,2,\ldots,N_0$, can be used to express the overall multi-objective minimizing objective minimizing objective function, as follows.

$$\text{Min } D=\sum_{i=1}^{N_o}\frac{f_i(x)-\breve{f}_i}{\hat{f}_i-\breve{f}_i} \tag{28}$$

V. Solution Algorithm for Optimal Placement of Capacitor

This section presents an efficient two-staged algorithm to achieve the best compromise among these conflicting objectives and thus solve the multi-objective capacitor placement problem. The first stage of the solution algorithm to find the decision region that is bounded by the ideal and worst solutions of the individual objective function. The second stage utilizes the compromise programming embedded in the immune algorithm to search for the trade-off solutions (non-inferior solutions). The pseudo code of the two-staged immune algorithm is described below.

/ * Stage 1 * /

1.Input system data and control parameters.

2.Set the number of antigens to the number of objectives (such that each antigen corresponds to an individual objective).

For objective = 1,2,...,N_0, do step 3-12, otherwise, proceed to step 13.

3.Randomly generate the initial antibodies (solutions, representing the location and size of capacitors to be installed)

4.Call late the affinity $(Ag)_I$ between the antigen and the antibody using Eq. (9).

/ * Herein, only the affinity between the antigen and its corresponding antibodies is calculated. * /

5.Determine the concentration c_v of each antibody in the repertoire, according to Eq. (10)

TABLE.I

ENERGY COST UNDER VARIOUS LOADS.

Load levels	Time interval (Hours)	Cost (NT$/kvar)
Peak-load(1.0)	1000	0.68
Medium-load(0.8)	6760	1.80
Light-load(0.5)	1000	2.85

TABLE II

CAPACITORS (kvar) TO BE INSTALLED

	Methods						
NO.OFBUS	The proposed method					Hung	Chiang
19	300	1200	300	600	300	600	300
50	300	900	600	900	300	300	1200
53	1200	600	600	600	600	300	0
Total_kvar	1800	2700	1500	2100	1200	1200	1500

6.If c_v exceeds a threshold δ_c, then this antibody becomes a memory antibody; proceed to the next step; else, proceed to step 10.

7.Select the best antibody with the maximum affinity for each antigen.

8.Calculate the affinity $(Ab)_{vw}$ between antibody v and the best antibody w using Eq. (8).

9. If these affinities $(Ab)_{vw}$ are greater than a preset value δ_a, then record the optimal $\hat{f}_i$ of the current generation and then proceed to step 12; otherwise proceed to the next step.

10. Suppress the antibodies with low concentrations (affinity).

11. Reproduce the antibodies by applying

$$A_{bi,new} = (x_{k\max} - x_{k\min}) \times d + x_{k\min} \qquad (29)$$

where $x_{k\max}$ and $x_{k\min}$ are the maximum and minimum value of the antibody respectively, and d is a random value between 0 and 1 .

12. If a given number of generations is reached, then go to the next step; otherwise, proceed to step4.

13. output the optimal solution $\hat{f}_i$ of the individual objective for $i = 1,2,\ldots N_0$.

/ * The outputs from the first stage include the unattainable best solutions of the individual objective $\hat{f}_i$ and the hypothetical worst solution $\breve{f}_k$ of the individual objective k

(and $k \neq i$), where theses outputs serve as the boundaries of the decision region, which is searched to find the global set of non-inferior solutions in the next stage.*/

/ * Stage 2 (Compromise Programming) * /

If the stop criterion is not met, perform steps 14 and 15; otherwise, proceed to step 16

14. Apply immune algorithm (as in stage 1, so a detailed description is not presented again here) to minimum the Euclidean distance, as described in Eq.

15. Check stop criterion: If over five consecutive *generations*, the sampled mean cost function does not change noticeably, or the number of generations reaches a preset value, and then stop the compromise programming.

16.Output the optimal non-inferior solutions.

5. Simulation Results

The presented solution algorithm was implemented and tested using Matlab [17]. The testing system includes seven branches and 69 buses, as presented in [6]. Table 1 lists the parameters of the objective functions, used to calculate the cost of the capacitors and the power loss. The unit of one capacitor bank is 300 Kvar at a cost of NT$61,900/bamk. The presented method outputs five non-inferior solutions (options)with different features, one of which is to be selected by the decision-makers. Tables 2,3 and 4 compare the results with those in [6] and[18], in terms of the capacitor to be installed,

TABLEIII

THE RESULT OF THE REAL POWER LOSS (kw) WITH AND WITHOUT INSTALLING CAPACITORS

Load level	Without Compen-sation	With compensation						
		The proposed method					Huang	Chiang
Light	538	393	457	347	401	337	347	345
Medium	1,715	1,019	995	1,042	994	1,116	1,186	1,040
Peak	3,190	1,865	1,752	1,965	1,806	2,134	2,276	1,964
Total_loss	5,443	3,204	3,204	3,354	3,201	3,587	3,809	3,349

6. Conclusions

Multi-objective optimization is of increasing importance in various field, and has a diverse range of applications. Highly effective and efficient multi-objective algorithms can promote the real power loss with and without compensation, and the cost of construction and power loss. The total costs of options 1,2 and4 are lower than those in [6] and [18], and the costs of options 3and5 are similar to those of [6] and [18]. Table 5 displays the maximum and minimum bus voltage before and after the capacitors are installed. Table 6 compares the results with those in [6] and [18], in terms of loading margin under various loads. Tables 5 and 6

demonstrate that the deviations of bus voltage and loading margin are similar.

In summary, the non-inferior solutions obtained using the presented method, in terms of voltage deviation, power loss, cost and loading margin, are better than (or similar to) those obtained using the methods of [6] and [18]. The simulation results reveal that the capacitor placement algorithm presented herein has the following merits.

(1) Allows the decision maker to obtain a set of optimal non-inferior solutions (multiple options) rather than single solution.

(2) Identifies plans for multi-object problems.

(3) Can be applied to large-scale distribution systems.

(4) Considers a more realistic problem formulation.

TABLE IV

COST OF REAL POWER LOSS AND CAPACITORS

Voltage (pu)	With compensation	With compensation						
		The proposed method					Huang	Chiang
Power loss	24,577,362	14,782,935	14,598,169	15,000,331	14,459,379	15997,953	16,972,622	14,977,768
Capacitor	0	371,400	557,100	309,500	433,300	247,600	247,600	309,500
Total_cost	24,577,362	15,154,335	15,309,831	15,309,831	14,892,679	16,245,553	17,220,222	15,287,268

TABLE V

MAXIMUM AND MINIMUM VOLTAGES OF THE TESTING SYSTEM BEFORE AND AFTER ACAPCITORS ARE INSTALLED

Voltage (pu)	With compensation	With compensation						
		The proposed method					Huang	Chiang
Maximum	1	1	1	1	1	1	1	1
Minimum	0.9092	0.937	0.9388	0.9317	0.9371	0.9271	0.9224	0.9298

TABLE VI

COMPARISON OF LOAD MARGINS WITH AND WITHOUT INSTALLED CAPACITORS

Load margin (pu)	With compensation	With compensation						
		The proposed method					Huang	Chiang
light	0.0992	0.2044	0.224	0.0673	0.0964	0.0673	0.0677	0.1568
medium	0.2689	0.201	0.2295	0.173	0.1752	0.2018	0.2034	0.2123
Peak	0.4378	0.3364	0.2732	0.2923	0.2742	0.3427	0.3455	0.2948

both scientific research and engineering applications in various areas. This work proposes the two-stage immune algorithm, embedding compromise programming, for solving the multi-objective capacitor placement problem. The concept of the non-inferior set is applied herein to obtain the set of optimal compromise solutions from which the decision maker can choose one. The simulation results indicate that the advantage of using the proposed technique is that it can find the best compromised solutions in a single run.

7. References

[1] M.E. Braran, and F.F. Wu " Optimum Capacitor Placment on Radial Distribution Systems", *IEEE Trans Power Delivery*, Vol. 4, No. 1,pp.725-734,Jan.1989.

[2] M. Chis, M.M.A. Salama, and S. Jayarm, "Capacitor Placement in Distribution Systems using Geuristic Search Strategies," *IEEE Proceedings-Generation, Transmission and Distribution*, Vol. 144, No.3,pp.225-230,May 1997.

[3] M.Y Cho, and Y.W. Chen, "Fixed/Switched Type Shunt Capacitor Planning of Distribution Systems by Considering Customer Load Pat-tems and Simplified Feeder Model," *IEEE Proceedings-Generation, Transmission and Distribution*, Vol. 144, No. 6,pp.533-540,Nov.1997

[4] C.T. Su, and C.C. Tsai, "A New Fuzzy Reasoning Approach to Optimum Capacitor Allocation for Primary Distribution system," in *Proceedings of 1996 IEEE on Industrial Technology Conference*, 1996,pp237-241.

[5] H.D Chiang, J.C Wang, O. Cockings, and H.D. Shin, Part I: A New Formulation and Overall Problem ",*IEEE Trans. Power Delivery*, Vol.5,No.2,pp.634-642,April 1990.

[6] H.D Chiang, J.C Wang, O. Cockings, and H.D. Shin, Part II: A New Formulation and Overall Problem ", *IEEE Trans. Power Delivery*, Vol.5, No.2, pp.634-642, April 1990.

[7] Y.C Huang, H.T. Yang, and C.L. Huang, "Solving the Capacitor Placement Problem in a radial Distribution System Using Tabu Search Approach," *IEEE Trans. Power Systems*,Vol.11,No. 4,pp.1868-1873,Nov.1996.

[8] S. Sundhararajan, and A. Pahwa, "Optimal Selection of Capacitors for Radial Distribution Systems Using a Genetic Algorithm," *IEEE Trans. Power Systems*, Vol. 9, No,pp.1499-1507,Aug.1994.

[9] V. Ajjarapu, and Z. Albanna, " Application of Genetic Based algorithms to Optimal Capacitor Placement, " in *Proceedings of the First International Form on Applications of Neural Networks to Power Systems*, July 1991,pp.251-255.

[10] F.C. Schweppe, H.M. Nerrill, and W.J. Burke, "Least-cost Planning: Issues and Methods," *Proceedings of the IEEE*, Vol. 77,No.6,pp.899-907, July 1989.

[11] D. Dasgupta, *Artifical Immune Systems and Their Applications*, Springer-Verlag Berlin Heidelberg, Saladruck, Berlin, 1999.

[12] Y. Tsujimura and M. Gen, "Entropy-based Genetic Algorithm for Solving TSP," in *1998 Second International Conference on Knowledge-Based*

Intelligent Electronic Systems, No. 2,21-23 April , 1998, Adelaide, Australia,pp.285-290.

[13] L.V. Wassendhove and L. Gelder, "Solving a Bircriterion Schedule Problem," European Journal of Operational Research, Vol. 4, pp. 42-48,1978.

[14] J. Horn, N. Nafpliotis, and D.E. Goldberg, "A Niched Pareto Genetic Algorithm for Multiobjective Optimization," in *Proceedings of the First IEEE Conference on Evolutionary Computation, 1994.IEEE World Congress on Computational Intelligence.*, Vol. 1,27-29 June 1994, pp.82-87.

[15] A. Goicoechea, D.R. Ganse, and L. Duckstein, *Multiobjective Decision Analysis with Engineering and Business Applications*, John Wiley and Sons, New York, 1982.

[16] M. Zeleny, *Multiple Criteria Decision Making*, McGraw-Hill Book Company, New York, 1982

[17] The Math Works,Inc. http://www.Mathworks.com/index.htm

[18] S.J Huang, "An Immune-Based Optimization Method to Capacitor Placement in A Radial Distribution System," *IEEE Trans. Power Delivery*, Vol.15, No.2,pp.744-749, April 2000.