Aircraft Optimization at the Early Stages of Design with a Hybrid Technique

Nesrin Cavus
German Aerospace Center (DLR), Air Transportation Systems
Scientific Assistant
Blohmstr. 20, 21079 Hamburg, Germany
Nesrin.Cavus@dlr.de

ABSTRACT

Aircraft design requires possessing huge amount of data analyses which are executed consecutively as well as parallel with different configurations. Throughout the design process many experts contribute to the design from different specializations not only with well-known handbook methods but also using engineering intuition like heuristic principles. These practical experiences which are aimed to be gained by a training algorithm are the main motivation of this research. In this study, a hybrid method is used to replace the random walk method of some optimization algorithms. The goal is to improve the steps for convergence and the results of a multidisciplinary optimization problem with changing number of design variables, which normally require over 1000 function evaluations to converge. An example aircraft design problem is used to determine if there are improvements in convergence steps while searching the whole design space. The results show that the applied technique increases the efficiency of the optimization for the early stages of aircraft design.

KEYWORDS: *artificial intelligence, probabilistic neural networks, aircraft design, multidisciplinary optimization*

NOMENCLATURE

m - Number of variables
pt - Number of patterns for one variable
n - Number of training points
k - Number of intervals
pt^m - Number of total patterns
f(x) - Probability density function

1 INTRODUCTION

Optimization for air vehicles is a complex process with depending on many design variables and the highly non-linear physics models. Besides that, the optimization process of an air vehicle has relatively lower local minimums with increasing variable numbers while considering other kind of optimization problems which have many local gradient changes. An artificial intelligence technique is applied here to reduce divergence relative to well-known algorithms like genetic algorithm and simulated annealing, with the required number of runs for convergence as the objective for changing number of design variables.

The motivation of this study has many bases. At first, an aircraft mission is a perfect closed loop process with the law of conservation of energy. Clearly, it has standard segments and each segment and its requirements – inputs and outputs – are well-known. In other words, this closed loop system is more conservative than many statistical problems and it is worth to use directed search methods for aircraft design. However, when the number of independent design variables increases, then the statistics based methods are helpful at intermediate steps of an optimization algorithm. In this study, probabilistic neural networks, which was introduced by [2], is taken as an accelerator of the optimization algorithm for its success on classification and pattern recognition. The method is given in detail in the following part of this study.

In addition to having a closed loop process feature, at the beginning of the design process researches may have huge number of data from wind tunnel tests as well as from flight tests at least for the conventional aircrafts. All of these data with engineering sense of experienced engineers, which is inseparable throughout a design process, have great value on all of the design phases. Learning from data, which is data mining, with the combination of engineering sense has very precious fine tuning effect on a design with knowledge based methods. Some of the complex interactions can also be estimated with relative magnitudes just from the data at hand at the beginning of the design process, which accelerates the calculations and force the results to converge to better values. With more philosophy, these complex but systematic machines, aircrafts, with dependent and independent variables with known interactions to each other in an optimization process can be improved with the algorithms which mimic pattern recognition property of human intelligence, i.e. Artificial Intelligence (AI), for the early as well as ensuing stages of design.

Further, optimization process of an aircraft can be structured with two main sub functions, optimization part and aircraft design part, which also includes many sub-optimization loops. The selection of the independent variable values and the constraints with their ranges has a direct effect on the performance of an optimization process. Any poor selection of a design point on a design space cost extra time, which may be totally useless. Therefore, not only the optimization part itself also the aircraft design part expends runtime, which can be thought as the multiplication of these times. Thus, the total process should be improved not to select unfavorable design points in order to save the run time and have better convergence for optimum design.

If the structure of the analysis is concerned widely used methods like regression and kriging do interpolations between set of points collected from the comparison of dependent and independent variables. The idea in this study is to use instantaneous associations between dependent and independent variables. Besides considering pointwise approximated linear/nonlinear relations, the behavior of the dependent variable can be stored while shifting one design point to another. With storing instantaneous behaviors of variables, redundant candidate design points can be avoided during the optimization process.

In this paper, with a given training data set the relations between independent variables and dependent variables are examined and then search areas are selected based on the successful changes on objective function values. Stored successful actions on the variables are considered as patterns and classified according to the function values. Resultant successful patterns are used to extend and deepen the design search space with the help of probabilistic neural network algorithm introduced by [2] and also dealt with here in detail in methodology part.

2 ARTIFICIAL INTELLIGENCE IN AIRCRAFT DESIGN

Knowledge based systems, computational intelligence and hybrids are specified as the tools of artificial intelligence in [3]. The introduced hybrid method in this study is composed of agents and probabilistic neural network algorithm. Agents work on gridded search space to find out and sort the shifting actions of the values, and probabilistic neural networks is used for classification of the successful and promising patterns.

Recently, the use of artificial intelligence tools continues to increase in aircraft multidisciplinary design optimization due to their successful implementations. A Knowledge based engineering approach to support aircraft multidisciplinary optimization was used by [4] to develop both conventional and novel geometries. Finite element analysis models are generated and time reduction is gained with the automated method.

Another method, Concurrent Learning was used with adaptive neural network based approximate models by [5] for a nonlinear fixed-wing aircraft model. Non-iterative two models were used to estimate limit and control margins. Network weights were updated from both past and current information, with which resultant values were better calculated.

To decrease the aircraft design cycle time [6] used two different artificial intelligence algorithms, neural networks and fuzzy logic. Aircraft weight, engine thrust and wing area were determined with the applied methods for the early phase of the aircraft design process. A specific class of light business jets is selected as a design case to approximate the take-off wing loading and take-off thrust loading. The actual results are approximated about ten percent for the preliminary design phase.

A fuzzy logic based artificial intelligence algorithm was applied to an unmanned combat aerial vehicle control system by [7]. The success of the algorithm was proved in high fidelity simulation environment. The algorithm was found to be highly responsive to complex situations and uncertainties.

As introduced, artificial intelligence methods with knowledge based techniques are used by many designers to improve the design solutions and the required run time. In this paper a hybrid technique is used to increase the efficiency of the optimization process. Two different test cases, Rastrigin function[1] , Binh and Korn function[14]; and a design case, a supersonic air vehicle mission, were applied. Due to having many local minimums and wide search space Rastrigin function is one of the good test methods for single objective optimization with its flexibility to increase the independent design variables. Binh and Korn function is used for proving the effectiveness of the algorithm in two-objective optimization. Then, the optimization algorithm is implemented to a supersonic air vehicle mission with the objective to get minimum unit cost with initial sizing calculations [15].

The method studied here has advantage on other guided random search techniques with assigning the directions on the search space for the variables while searching the promising grids. With this method the training sets are used to decide on the next search space with the produced values of the cumulated runs. In the next section the methodology is explained more in detail.

3 METHODOLOGY

Optimization problems are defined as gathering many challenging variables and obtaining the best solution space. Although one may not have any time constraint the selected search method directly affects the quality of the results. Meanwhile one technique may be successful for one type of problem whereas it may fail in another type of problem. Because of that, researchers try to develop a common tool that can adapt to different situations. Especially for the complex projects executed between different scientist, or departments, or even different companies it becomes a must a unique tool controllable by everybody to catch the better solutions.

The optimization technique used in this study depends on probabilistic neural network algorithm, and this type of neural network is applied here for its success on pattern recognition and classification. In this hybrid method the advantages of both gradient based and evolutionary algorithms were used. In an optimization process the run time and the convergence characteristics are affected by increasing the number of variables drastically. This complicacy is overcome with a hybrid technique in this study.

The method works as an accelerator of the optimization part. This technique does not remove the main function of the optimization process but improves the random selection of the variables values. As illustrated in Fig. 1 the variable numbers and first set of training points are introduced to the program at first. Since the number of training points another concern for the success of the AI algorithms, two different cases are tested. For the design case selected in this study, training points are kept small enough for the startup and for each pattern 10 training points are randomly selected. Depending on the problem characteristics the number of training points can also be selected relative to the number of independent variables. As a second case increasing number of training points with increasing variable numbers like k^m where k is the number of intervals on a pattern and m is the number of independent variables.

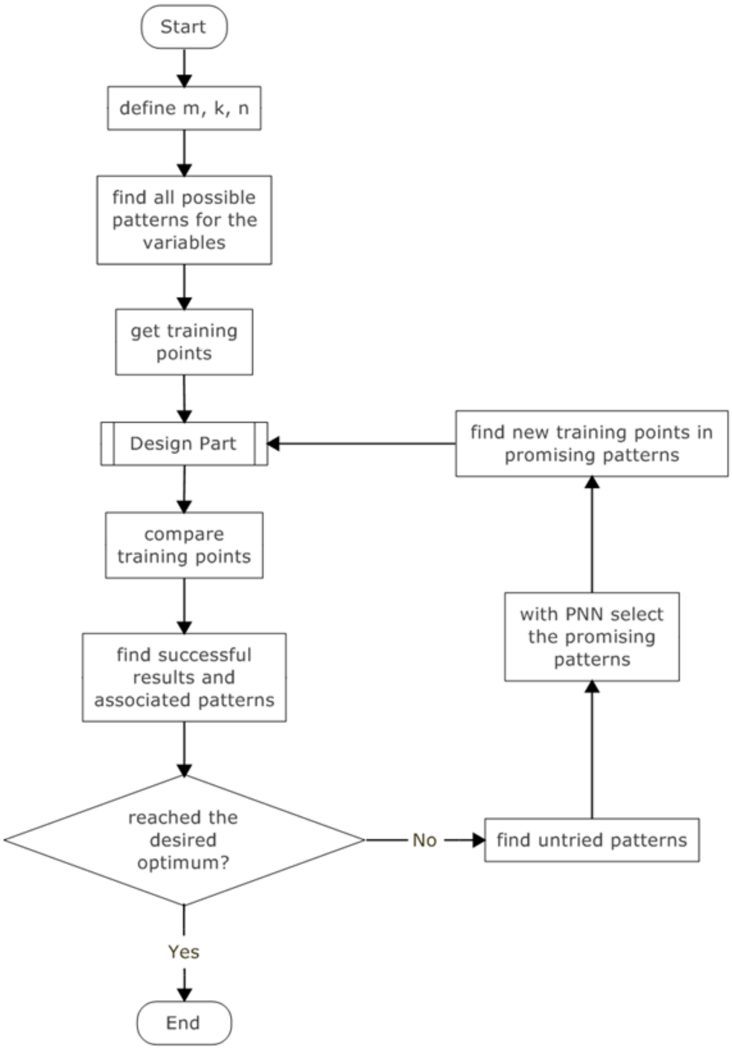

Figure 1: Flowchart of the algorithm

After selecting the number of training points, which are applied on each grid, the randomly selected variable values are sent the design part. Then the results are compared with each other to build up a correlation matrix which shows the changes in the independent design variable values and the objective values. This means, if the training point number is *n*, the number of resultant combinations between these training points will be *n.(n-1).* These combinations carry the knowledge of the decrement/increment effects between these points. At that stage, increment is symbolized as 1 and decrement as -1. If there is no change between the compared values it can be taken as 0. This numbering system will be used for next steps for handling the correlations as patterns. If the objective is to minimize the fitness function the correlation patterns with the objective correlation value -1 are taken as the successful patterns, others are left as unsuccessful patterns. Additionally, depending on the lower and upper bounds of the dependent and independent variables, the search space for each variable is divided in *k* intervals, and the interval values are stored. With bipolar (also 0) values and intervals, each combination of training points can be processed as patterns. Each variable has number of patterns, *pt*, calculated as in Eq. 1:

$$pt = 2 * (k + \sum_{i=1}^{k} (k - i)) \quad (1)$$

If the number of design variables is *m*, then the number of total patterns that can be tried is pt^m. As an example, for one variable and 5 intervals the possible patterns are illustrated sequentially in Fig. 2; for two variables and 5 intervals the patterns are illustrated in Fig. 3. Then, for the first training data set from the total patterns the unattempt patterns can be extracted.

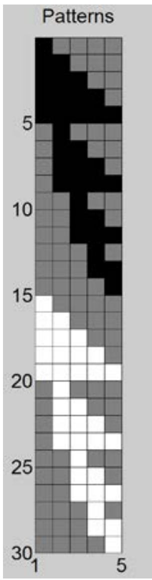

Figure 2: Patterns for one variable (k=5)

At that point, probabilistic neural networks come into action. Probabilistic neural networks use Bayes Strategy instead of using sigmoidal activation function that is widely used with an exponential function in back-propagation algorithm. This method can compute nonlinear decision boundaries, which can be updated immediately with a new data, and can also be operated in parallel [2]. Because of its structure, it is faster than back-propagation especially for pattern recognition and classification. Probabilistic Neural Networks were explained more in detail in [2] and also mentioned here a little through this reference.

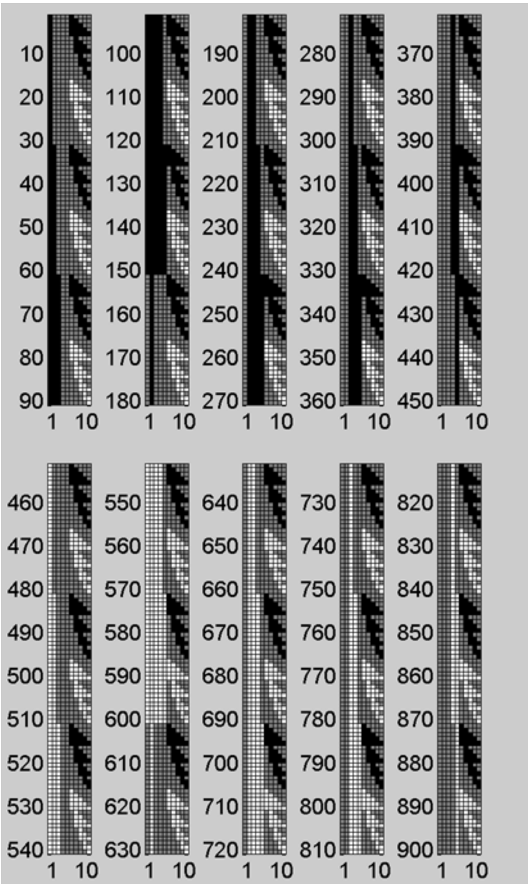

Figure 3: Patterns for two variables (k=5)

While classifying patterns the decision rules are set to minimize the expected risks. These rules or strategies are called Bayes Strategies and can be applied to any number of categories [2 and 8]. Considering the categories A and B; the state θ with the category A is θ_A, and with the category B is θ_B, and also the probability density functions are $f_A(x)$ and $f_B(x)$ respectively. Also, l_A and l_B are the loss functions related with the decisions $d(x) = \theta_A$ when $\theta = \theta_B$ and $d(x) = \theta_B$ when $\theta = \theta_A$ (the losses are taken to be equal to zero when the decisions are correct). Further, h_A and h_B are the priori probability of occurrence of patterns from category A and B, and $h_B = 1- h_A$.

Then, for a state θ based on a set of measurements represented by a p-dimensional vector $x^t = [x_1 \ldots x_j \ldots x_p]$ the Bayes decision rule is written as in Eq. 2:

$$d(x) = \theta_A \; if \; h_A l_A f_A(x) > \; h_B l_B f_B(x)$$
$$d(x) = \theta_B \; if \; h_A l_A f_A(x) < h_B l_B f_B(x) \tag{2}$$

Also, the boundary between the region in which Bayes decision $d(x) = \theta_A$ and the region in which Bayes decision $d(x) = \theta_B$ is given as in Eq. 3:

$$f_A(x) = K f_B(x) \tag{3}$$

Where

$$\mathrm{K} = \frac{h_B l_B}{h_A l_A} \tag{4}$$

The ratio of the loss functions, h_B / h_A , can be set to -1 if there is no reason for biasing the decision. According to Parzen [9] a family of estimates of $f(x)$, at all points x the probability density function is continuous, is given with Eq. 5:

$$f_n(x) = \frac{1}{n\lambda} \sum_{i=1}^{n} W \left[\frac{(x - x_{Ai})}{\lambda}\right] \tag{5}$$

Eq. 6 is for weighting function $W(y)$ and states that weights are not bounded and cannot reach infinity:

$$sup_{-\infty<y<\infty} |W(y)| < \infty \tag{6}$$

where sup indicates the supremum.

$$\int_{-\infty}^{\infty} |W(y)| dy < \infty \tag{7}$$

$$\lim_{y\to\infty} |yW(y)| = 0 \tag{8}$$

and

$$\int_{-\infty}^{\infty} W(y) dy = 1 \tag{9}$$

In Eq. 5, let λ is chosen as a function of n then $\lambda = \lambda$ (n), and

$$\lim_{n\to\infty} n\lambda(n) = \infty \tag{10}$$

Parzen [9] proved that the expected error goes to zero with the number of training samples going to infinity:
$E|f_n(x) - f(x)|^2 \; \to 0 \; as \; n \to \; \infty$

Murthy [10 and 11]) relaxed the assumptions of the absolute continuity of the distribution *F(x)* and Cacoullos [12] extended Parzen's results for multivariate case. Then the multivariate estimates are found by Eq. 11 as:

$$f_A(x) = \frac{1}{(2\pi)^{p/2}\sigma^p}\frac{1}{m} \times \sum_{i=1}^{m} exp\left[\frac{(x-x_{Ai})^T(x-x_{Ai})}{2\sigma^2}\right] \qquad (11)$$

where σ is the smoothing parameter and has a very important influence on the approximations

The probabilistic neural networks like feed forward networks have parallel structure. This type of neural networks is very flexible to accepting new data and easy with one-step only learning technique. It does not learn from trials, instead learns from experience that others made for the neural network. Therefore it depends on the functions used inside the neuron. Because of these characteristics, the probabilistic neural networks are faster than back-propagation and they perform well with few training points. In this study, probabilistic neural networks are preferred to use for their advantages and success on the pattern recognition and classification, and also tolerance the usage of binary-bipolar numbering combination.

As an example, in Fig. 4 the usage of probabilistic neural networks in the hybrid algorithm is illustrated for two variables, X_1 and X_2, and the objective, Y. At first, the successful and unsuccessful patterns are distinguished according to their influence on the objective function. If the objective function is decreasing (for finding minimum) with a pattern then the pattern is defined as the successful pattern and shown here with arrow headed blue lines. Head of the arrow shows the direction of the action. If it is away from the lower bound of the variable it means the variable value is increased at the successive point and has the interval value of 1 for that variable. If it is in opposite direction then the value for that interval becomes -1. If the pattern has an effect on the objective function to increase then the pattern is defined as unsuccessful pattern and shown with red headed lines. The directions of the arrows and the numbering for the successful patterns are also valid for these unsuccessful patterns.

After collecting the successful and unsuccessful patterns, probabilistic neural network is trained. Then untried patterns are picked out from the combined set of total possible patterns which are formed with the variable and interval numbers. Untried patterns are applied on the trained neural network and the possible successful and unsuccessful patterns are distinguished. Fig. 4 shows how probabilistic neural network selects promising patterns from the successful and unsuccessful patterns. It actually passes the patterns which have one or two digit differences from the successful patterns and their combinations. As a result, as in the Fig. 4b the promising pattern would be similar to one of the gray lines and headed according to the other successful patterns which may be neighboring, parallel or both. From the performance values of the tried patterns, neighboring patterns which have greater probability to minimize the fitness function are selected with the help of probabilistic neural networks. These promising patterns send to the design part, which includes the fitness function, and calculated results are turned with their patterns to further deliberation. These successful patterns are eliminated based on three criteria.

The pattern:

- that minimizes the results;
- that has the equal resultant values (due to probability of convergence);
- that has the minimum result (due to probability of convergence).

At the end of this process throughout the design space there would not be any untraced space just from the first set of training points. For the next step, as in Fig. 4c, a pattern is selected and next set of training points are applied to this area. Further, the following step would like Fig 4d. The process will continue with the successful and promising successful grids on the search area in the loop as in the flowchart of the algorithm in Fig. 1.

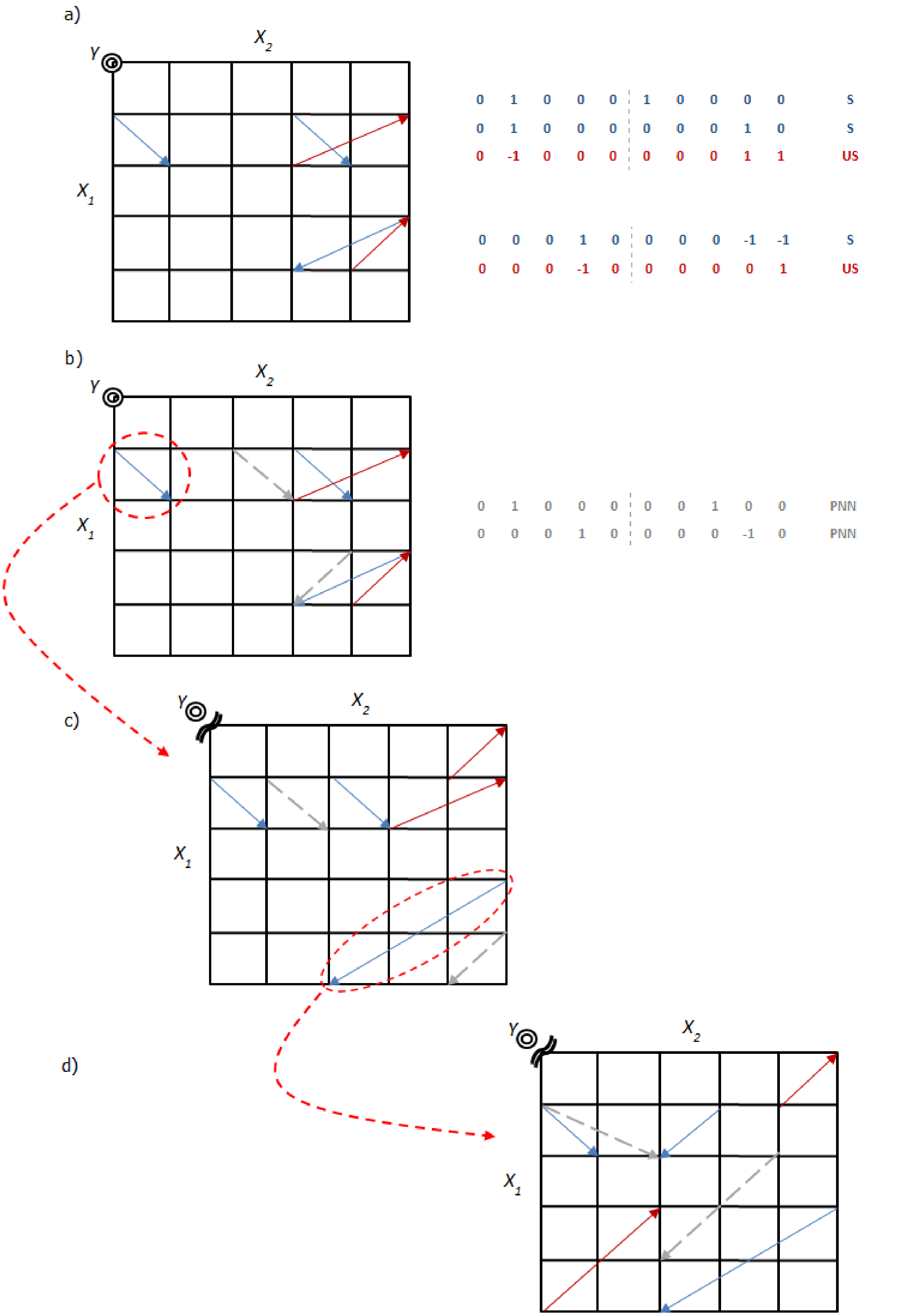

Figure 4: Selecting promising patterns with PNN

When the required criterion for terminating the program is reached then the process is ended with the minimum value (or maximum value for the maximization). If the fitness is not at the desired level then the selected successful patterns is sent to the design part for generating further training points for the related intervals as described on the flowchart in Fig. 1. The loop is repeated until the desired optimum reached.

3.1 Test Case I

As an initial case, Rastrigin function[1] due to it is large search space and large number of local minimums is used to test the algorithm. For n-dimensional domain, it is given as:

$$f(x) = 10n + \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i)] \tag{12}$$

and $x_i \in [-5.12, 5.12]$, Rastrigin function has a global minimum at $x=0$ where $f(x) = 0$. It is shown at Fig. 5 for two independent variables:

For this test case, the number of independent variables, the interval numbers for the patterns and the number of training points are taken as 2, 4 and 10, respectively.

Results show that, the required number of evaluations for hybrid AI, 20, is almost one third of the evaluations of simulated annealing algorithm, 60, which converges to local minimums with changing starting point.

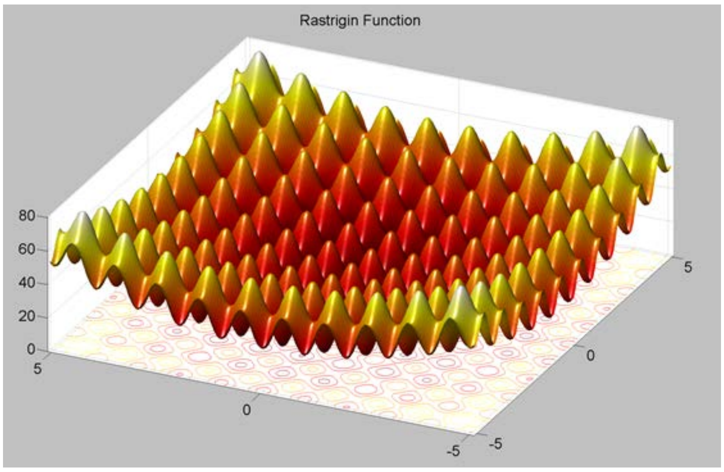

Figure 5: Rastrigin function for two variables

3.2 Test Case II

The hybrid technique is used for Binh and Korn function[14] function for two objective optimization which is given by Eq. 13:

$$f_1(x,y) = 4x^2 + 4y^2$$
$$f_2(x,y) = (x-5)^2 + (y-5)^2 \tag{13}$$

Constraints:

$$g_1(x,y) = (x-5)^2 + y^2 \leq 25$$
$$g_2(x,y) = (x-8)^2 + (y+3)^2 \geq 7.7 \tag{14}$$

Search domain:

$$0 \leq x \leq 5$$
$$0 \leq y \leq 3 \tag{15}$$

Two objective optimization problems are as simple as single objective optimization problems. For both minimizing functions it is simply integrating the functions into one fitness function by adding them. Fig. 6 shows the results of the hybrid method and genetic algorithm for Binh and Korn function. The effectiveness of the hybrid method is seen for the non-dominated points. On the other hand, the required time and evaluation number for the convergence for the hybrid method is almost same as genetic algorithm. However, further improvements can be done by eliminating the points which comes from the similar or partially related intervals. In another way, the intervals which are covered by another pattern should be extracted from successful pattern cluster, and the intervals that share the same boundaries could use a common matrix.

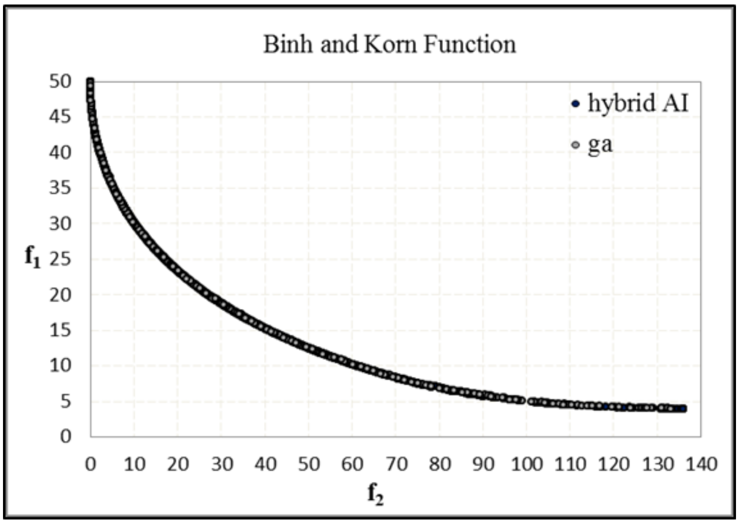

Figure 6: Binh and Korn function

3.3 Design Case

An unmanned supersonic aircraft, whose conceptual design model is given in detail Ref. 13, is taken as the design case for a single objective optimization. That covers a general mission of a combat aircraft which consists of take-off, climb, cruise, loiter, descent, combat, climb, cruise, descent and landing. The objective is to reduce the unit cost of the aircraft. As a single objective optimization problem, the variables are compared with the resultant cost and patterns are selected to decrease this objective function. Wing span, wing sweep angle, vertical tail sweep angle, horizontal and vertical tail volume coefficients, loiter time and payload are selected as the design variables.

The upper and lower bounds of the variables are listed on Table 1. Optimization is made for 3, 4, 5, 6 and 7 variables. They are selected according to order of the table. The minimum point is reached at the point where the unit cost is $ 17.49 million for each changing number of variables' runs. The dependent and the independent variables are listed on the Table 2.

Because, always it is worth to mention the number of training points for a neural network, in this study two different runs for each variable set are included as fixed (n=10) and changing number of training points ($n=k^m$).

Table 1: Design variables and their boundaries

	Wing span (m)	Wing sweep angle (deg)	Vertical tail sweep angle (deg)	Horizontal tail volume coefficient	Vertical tail volume coefficient	Loiter time (hour)	Payload (kg)
Upper bound	15	50	55	0.45	0.45	0.75	2500
Lower bound	8	30	35	0.40	0.40	0.10	1500

Table 2: Variables and optimization results

Airfoil wing	NACA64A210	W_0 (kg)	5763
Airfoil horizontal tail	NACA0012	W_e (kg)	3112
Airfoil vertical tail	NACA0012	W_f (kg)	1151
F110-GE-100 with ab (kN)	129.4	Static margin %	2.5
Maximum Ceiling (ft)	55000	$W/S_{takeoff}$	360
Cruise altitude (ft)	40000	Range (km)	2168
M_{cruise}	1.37	$Endurance_{max}$ (hour)	2.4
C_{D0}	0.0234	V_{corner} (km/h)	569
$C_{Lacmax\ \ takeoff}$	1.3	Available sustained load $factor_{max}$	9.0
Wing area (m^2)	16	Number of turns	3
Aspect ratio	4	Wing span (m)	8
Taper ratio	0.216	Wing sweep angle (deg)	30
Fuselage length (m)	7.93	Vertical tail sweep angle (deg)	35
Horizontal tail sweep (deg)	35	I_{HTco}	0.4
T/W	1.08	I_{VTco}	0.4
Engine bypass ratio	0.87	$W_{payload}$ (kg)	1500
Quantity	500	Loiter (hour)	0.1
Cost ($ million)		17.49	

For the first case, 10 training points are given for the first run and for the following each successful iterations. The interval number is kept small and selected as 2 for the increasing number of variables. Optimization results for a fixed number of training points at 10 are given in Table 3, and for the changing number of training points, proportional to the interval number and the variable numbers are given at Table 4. For the first case, when the number of training points are fixed the network lose its success with the increasing number of variables. Also, as Table 3 and Table 4 are considered at the same time; when the training points are more than the required value (as for n=3), the network is highly trained and it does not let enough patterns to be included in the promising pattern set. Thus the success of the method decreases. And also, for the increasing number of variable numbers (m>3) the fixed number of training points are less than the required level, then the success of the promising patterns of the optimization diminishes. This effect is seen in both of the tables, Table 3 and Table 4. For a relative number of intervals and variable numbers, the results are gathered in Table 4 after calculating each pattern values, and it is seen that with increasing number of variable numbers, the success of the algorithm increases because of eliminating less promising patterns, which come from more digit changes.

Table 3: Optimization results for fixed number of training points (n=10)

Number of variables	7	6	5	4	3
Number of training points	10	10	10	10	10
Number of intervals	2	2	2	2	2
Total pattern number	279936	46656	7776	1296	216
Used patterns	90	90	76	84	58
Unused patterns	279846	46566	7700	1212	158
Successful patterns	45	45	38	42	29
Unsuccessful patterns	45	45	38	42	29
Promising patterns	147638	24109	4011	614	83
Total promising patterns	147683	24154	4049	656	112
Successful promising patterns	125287	21861	3737	611	106
Success of the method	0.848	0.905	0.923	0.931	0.946

Table 4: Optimization results for changing number of training points ($n=k^m$)

Unit cost ($ million)	17.49	17.49	17.49	17.49	17.49
Time (sec)	410	218	122	74	50
Number of variables	7	6	5	4	3
Number of training points	128	64	32	16	8
Number of intervals	2	2	2	2	2
Total pattern number	279936	46656	7776	1296	216
Used patterns	12586	3076	718	138	32
Unused patterns	267350	43580	7058	1158	184
Successful patterns	6323	1543	363	70	16
Unsuccessful patterns	6323	1543	363	70	16
Promising patterns	134851	22052	3587	583	92
Total promising patterns	141174	23595	3950	653	108
Successful promising patterns	138166	22980	3836	629	103
Success of the method	0.979	0.974	0.971	0.963	0.954

4 CONCLUSION

The new hybrid technique studied here is used for Rastrigin function, Binh and Korn function and for a supersonic aircraft discrete mission to minimize the unit cost. It is proved that the used hybrid artificial intelligence method increases the efficiency of the optimization and improves the design task; and it is seen to be competitive to the other optimization techniques.

In addition, the number of training points and variables are found out as the determining parameters on the efficiency of the hybrid algorithm. While increasing the values of these parameters the convergence is improved but the cycle time and the memory usage are also increased.

In this study, it is shown that the probabilistic neural networks with the combination of rule based agent systems the optimization of a design is possible and has advantages on different type of problems. Although, searching the whole design area with promising patterns expends the run time it also helps to reduce the calculation time for poor design points which are experienced by the previous trials.

REFERENCES

1. L.A. Rastrigin; 1974; "Systems of External Control"; Nauka; Moscow
2. D.F. Specht; 1998; "Probabilistic Neural Networks for Classification, Mapping, or Associative Memory"; *IEEE International Conference on Neural Networks*; **1**; July; pp. 525–532
3. A.A. Hopgood; 2001; *Intelligent Systems for Engineers and Scientists*; Second edition; CRC Press; Boca Raton; FL
4. G. La Rocca, M. J. L. van Tooren; 2009; "Knowledge-Based Engineering Approach to Support Aircraft Multidisciplinary Design and Optimization"; *Journal of Aircraft*; **46**(6); pp. 1875–1885; doi:10.2514/1.39028
5. G. Gursoy, I. Yavrucuk; 2016; "Direct Adaptive Limit and Control Margin Estimation with Concurrent Learning"; *Journal of Guidance, Control, and Dynamics*; **39**(6); pp. 1356–1373; doi:10.2514/1.G001515
6. M.A.A. Oroumieh, S.M.B. Malaek, M. Ashrafizaadeh, S.M. Taheri; 2013; "Aircraft Design Cycle Time Reduction using Artificial Intelligence"; *Aerospace Science and Technology*; **26**; pp. 244–258
7. N. Ernest, D. Carroll, C. Schumacher, M. Clark, K. Cohen, G. Lee; 2016; "Genetic Fuzzy based Artificial Intelligence for Unmanned Combat Aerial Vehicle Control in Simulated Air Combat Missions"; *Journal of Defense Management*; **6**(1); doi: 10.4172/2167-0374.1000144
8. A.M. Mood, F.A. Graybill; 1962; *Introduction to Theory of Statistics;* Macmillan; New York
9. E. Parzen; 1962; "On Estimation of a Probability Density Function and Mode"; *Annals of Mathematical Statistics*; **33**; pp. 1065-1076
10. V.K. Murthy; 1965; "Estimation of a Probability Density"; *Annals of Mathematical Statistics*; **36**; pp. 1027-1031
11. V.K. Murthy; 1966; "Nonparametric Estimation of Multivariate Densities with Applications", in P.R. Krishnaiah Ed.; *Multivariate Analysis*, Academic Press; New York; pp. 43-58
12. T. Cacoullos; 1966; "Estimation of Multivariate Density"; *Annals of the Institute of Statistical Mathematics*; **18**(2); Tokyo; pp. 179-189
13. N. Cavus; 2009; "Multidisciplinary and Multiobjective Design Optimization of an Unmanned Combat Aerial Vehicle"; *M.S. Thesis*; Middle East Technical University; Ankara
14. T.T. Binh, U. Korn; 1997; "Multiobjective Evolution Strategy with Linear and Nonlinear Constraints"; *In Proceedings of the 15th IMACS World Congress on Scientific Computation, Modeling and Applied Mathematics*; pp. 357-362
15. D.P. Raymer; 1999; *Aircraft Design: A Conceptual Approach*; Third edition; AIAA Education Series; American Institute of Aeronautics and Astronautics; Virginia