Global Journal of Pure and Applied Mathematics.
ISSN 0973-1768 Volume 10, Number 5 (2014), pp. 685-695

http://www.ripublication.com

A New Crisp Neural Network for Solving Fuzzy Linear Programming Problems

G. Selvaraj and P. Pandian

Department of Mathematics, School of Advanced Sciences
VIT University, Vellore-14, INDIA
e-mail: gselvaraj89@gmail.com, pandian61@rediffmail.com

ABSTRACT

A new crisp neural network for solving fuzzy linear programming problems is proposed. Energy function of the proposed network is developed using the penalty function. The solution obtained by the proposed method is faster and more accurate. Numerical examples are shown to illustrate the efficiency of the proposed method.

Keywords: fuzzy linear programming problem, triangular fuzzy numbers, multi-objective linear programming problem, neural network

1. INTRODUCTION

Optimization plays an important role in decision making theory. In real life situation, the data obtained for decision making problems are not all precise. Zadeh[25] introduced the concepts of fuzzy set which is not precise. Fuzzy optimization has been developed on the basic concepts of fuzzy set theory and classical optimization. Now, in applied science and engineering applications, decision makers consider optimization problem in fuzzy environment where it has fuzzy objective function or/and fuzzy constraints to maximize/minimize the objective function. Based on the concepts of fuzzy sets by Bellman and Zadeh [2], many researchers (Zimmerman [26], Tanaka and Asai [21], Shaocheng [20], Ebrahimnejad et al. [5], Ganesan and Veeramani [8], Jayalakshmi and Pandian[11], Amit Kumar[1] and Pandian[18]) have studied linear optimization problems in fuzzy environment.

A neural network approach for solving linear programming (LP) problems was first introduced by Tank and Hopfield [22] in which a LP circuit on the net was used. Kennedy and Chua [12] proposed an improved Hopfield and Tank's neural network model which is always guaranteed the convergence, but it converges to only an approximation of the optimal solution. In the literature, many researchers (Maa and

Shanblatt [13], Xia [23], Nguyen [17], Malek and Yari [16], Malek and Alipour[15], Ghasabi-Oskoei et al. [10], Gao and Liao [9], Cichocki et al. [4], Effati and Nazemi [6]) have shown interest to solve crisp optimization problems by various neural network models. Yinzhen et al. [24] proposed a crisp neural network approach for solving fuzzy shortest path problem based on linear programming. Pehlivan and Apaydin [19] solved fuzzy LP problems by both simplex method and neural network approach and also, they compared the results. Effati et al. [7] proposed a fuzzy neural network model for solving two types of fuzzy LP problems which having either decision variable or co-efficient is fuzzy parameter but not both.

In this paper, we propose a new neural network model for solving fuzzy LP problems. In this proposed network, we develop a crisp neural network in which the energy function is employed with the use of penalty function. We decompose the given fuzzy LP problem into multi objective linear programming (MOLP) problem and then, solve the MOLP problem by a crisp neural network.Then, we obtain a solution to the fuzzy LP problem using the solution of the MOLP. The solution obtained by the proposed method is realistic, meaningful and applicable. For solving real life LP problems having more number of decision variables with fuzzy parameters, the proposed neural network model helps to provide realistic solution to the problem in fast manner and more accurate.

2. PRELIMINARIES

We need the following definitions of the basic arithmetic operators and partial ordering relations on fuzzy triangular numbers based on the function principle which can be found in Bellman and Zadeh [2].

Definition 2.1: A fuzzy number $\tilde{a}$ is a triangular fuzzy number denoted by (a_1, a_2, a_3) where a_1, a_2 and a_3 are real numbers and its member ship function $\mu_{\tilde{a}}(x)$ is given below:

$$\mu_{\tilde{a}}(x) = \begin{cases} (x-a_1)/(a_2-a_1) & \text{for } a_1 \le x \le a_2 \\ (a_3-x)/(a_3-a_2) & \text{for } a_2 \le x \le a_3 \\ 0 & \text{otherwise} \end{cases}$$

Let $F(R)$ be the set of all real triangular fuzzy numbers.

Definition 2.2: Let (a_1, a_2, a_3) and (b_1, b_2, b_3) be in $F(R)$. Then,

(i) $(a_1, a_2, a_3) \oplus (b_1, b_2, b_3) = (a_1+b_1, a_2+b_2, a_3+b_3)$.

(ii) $(a_1, a_2, a_3) \,\Theta\, (b_1, b_2, b_3) = (a_1-b_3, a_2-b_2, a_3-b_1)$.

(iii) $k(a_1, a_2, a_3) = (ka_1, ka_2, ka_3)$, for $k \ge 0$.

(iv) $k(a_1, a_2, a_3) = (ka_3, ka_2, ka_1)$, for $k < 0$.

(v) $$(a_1, a_2, a_3) \otimes (b_1, b_2, b_3) = \begin{cases} (a_1b_1, a_2b_2, a_3b_3), & a_1 \ge 0, \\ (a_1b_3, a_2b_2, a_3b_3), & a_1 < 0, a_3 \ge 0, \\ (a_1b_3, a_2b_2, a_3b_1), & a_3 < 0. \end{cases}$$

Definition 2.3: Let $\tilde{A} = (a_1, a_2, a_3)$ and $\tilde{B} = (b_1, b_2, b_3)$ be in $F(R)$, then

(i) $\tilde{A} \approx \tilde{B}$ iff $a_i = b_i$, $i = 1,2,3$;

(ii) $\tilde{A} \preceq \tilde{B}$ iff $a_i \le b_i$, $i = 1,2,3$ and

(iii) $\tilde{A} \succeq \tilde{B}$ iff $a_i \ge b_i$, $i = 1,2,3$.

We need the following result and definition which are found in Madan M Gupta et al. [14].

Theorem 2.1: (Lyapunov function method) Let x^* be an equilibrium point for the system $\frac{dx(t)}{dt} = f(x(t))$. Let $V : R^n \to R$ be a continuously differentiable function such that

(i) $V(x^*) = 0$ and $V(x) > 0, \forall x \ne x^*$;

(ii) $V(x) \to \infty$ *when* $\|x\| \to \infty$ and

(iii) $\frac{dV(x)}{dt} < 0$ *for all* $x \ne x^*$.

Then, $x = x^*$ globally asymptotically stable.

Definition 2.4: Any scalar function $V(x)$ that satisfies all the requirements of the Theorem 2.1. is called a Lyapunov function for the equilibrium state $x = x^*$.

3. FULLY FUZZY LINEAR PROGRAMMING PROBLEM

Consider the following fully fuzzy LP problem with m fuzzy equality constraints and n fuzzy decision variables as follows:

(P) Minimize $\tilde{c}^T \tilde{x}$

subject to $\tilde{A} \otimes \tilde{x} \approx \tilde{b}$, $\tilde{x} \succeq \tilde{0}$, (1)

where, $\tilde{a}_{ij}, \tilde{c}_j, \tilde{x}_j, \tilde{b}_i \in F(R)$, for $1 \le i \le m$ all and $1 \le j \le n$, $\tilde{c}^T = (\tilde{c}_j)_{1\times n}$, $\tilde{A} = (\tilde{a}_{ij})_{m\times n}$, $\tilde{x} = (\tilde{x}_j)_{nx1}$ and $\tilde{b} = (\tilde{b}_i)_{mx1}$.

A fuzzy vector $\tilde{x} = (\tilde{x}_j)_{n\times 1}$ where $\tilde{x}_j$ are in *F(R)* for all j, satisfies the relation (1) is called a feasible solution to (P).

A feasible solution $\tilde{x} = (\tilde{x}_j)_{n\times 1}$ to the problem (P) where $\tilde{x}_j$ are in *F(R)* for all j, is said to be optimal if $\tilde{c}^T \tilde{x} \preceq \tilde{c}^T \tilde{y}$, for all feasible $\tilde{y}$ of the problem (P).

Now, the triangular fuzzy numbers $\tilde{c}_j$, $\tilde{x}_j$, $\tilde{a}_{ij}$ *and* $\tilde{b}_i$ are represented as (p_j, q_j, r_j), (x_j, y_j, z_j), (a_{ij}, b_{ij}, c_{ij}) and (b_i, g_i, h_i) respectively. Then, the problem (P) can be written as follows:

(P) Minimize $\sum_{j=1}^{n} (p_j, q_j, r_j) \otimes (x_j, y_j, z_j)$

subject to

$$\sum_{j=1}^{n}(a_{ij},b_{ij},c_{ij})\otimes(x_j,y_j,z_j)\approx(b_i,g_i,h_i)\text{, for all } i=1,2,...,m$$

$(x_j,y_j,z_j)\succeq\tilde{0}$, j =1, 2, …, m.

Now, using the arithmetic operations and partial ordering relations, we construct the following MOLP problem from the problem (P):

$$\text{Minimize } M_1=\sum_{j=1}^{n}\text{lower value of }\left((p_j,q_j,r_j)\otimes(x_j,y_j,z_j)\right)$$

$$\text{Minimize } M_2=\sum_{j=1}^{n}\text{middle value of }\left((p_j,q_j,r_j)\otimes(x_j,y_j,z_j)\right)$$

$$\text{Minimize } M_3=\sum_{j=1}^{n}\text{upper value of }\left((p_j,q_j,r_j)\otimes(x_j,y_j,z_j)\right)$$

subject to,

$$\sum_{j=1}^{n}\text{middle value of }\left((a_{ij},b_{ij},c_{ij})\otimes(x_j,y_j,z_j)\right)=g_i\text{, for all } i=1,2,...,m;$$

$$\sum_{j=1}^{n}\text{ lower value of }\left((a_{ij},b_{ij},c_{ij})\otimes(x_j,y_j,z_j)\right)=b_i\text{, for all } i=1,2,...,m\ ;$$

$$\sum_{j=1}^{n}\text{upper value of }\left((a_{ij},b_{ij},c_{ij})\otimes(x_j,y_j,z_j)\right)=h_i\text{, for all } i=1,2,...,m;$$

$M_2\geq M_1$; $x_j\leq y_j$, $M_3\geq M_2$; $y_j\leq z_j$, $x_j\geq 0$, $\forall$ j $=1,2,\ldots,n$.

The above MOLP problem can be rewritten in the matrix form as follows:

(M) Minimize $M_1=C_1^TX_1$

Minimize $M_2=C_2^TX_2$

Minimize $M_3=C_3^TX_3$

subject to $A_1X_1=B_1$, $C_2^TX_2\geq C_1^TX_1$, $X_1\leq X_2$, $A_2X_2=B_2$,

$A_3X_3=B_3$, $C_3^TX_3\geq C_2^TX_2$, $X_2\leq X_3$, $X_i\geq 0, i=1,2,3$.

where $X_1=(x_1,....,x_n)$, $X_2=(y_1,....,y_n)$, $X_3=(z_1,....,z_n)$ and C_1,C_2,C_3 are the objective coefficients, A_1,A_2,A_3 are the co-efficient of constraints for the variables X_1,X_2,X_3 respectively and also corresponding RHS constant column vectors are stated as B_1,B_2,B_3 .

Now, we obtain a relation between the solution of the problem (M) and the solution of the fuzzy LP problem (P). This result is used in the proposed neural network for obtaining the solution of the fuzzy LP problem.

Theorem 3.1: If $\{X_1^\circ,X_2^\circ,X_3^\circ\}$ is an optimal solution to the problem (M), then $\tilde{X}^\circ=(X_1^\circ,X_2^\circ,X_3^\circ)$ is an optimal solution to the fuzzy LP problem (P).

Proof: Now, from the construction of the MOLP problem and since $\{X_1^\circ, X_2^\circ, X_3^\circ\}$ is an optimal solution to the problem (M), $\tilde{X}^\circ = (X_1^\circ, X_2^\circ, X_3^\circ)$ is a feasible solution to the fuzzy LP problem (P).
Let $\tilde{Y} = (Y_1, Y_2, Y_3)$ be a feasible solution to the problem (P).
Clearly, $\{Y_1, Y_2, Y_3\}$ is a feasible solution to the problem (M).
Now, since $\{X_1^\circ, X_2^\circ, X_3^\circ\}$ is an optimal solution to the problem (M), we have
$C_1^T X_1^\circ \le C_1^T Y_1$; $C_2^T X_2^\circ \le C_2^T Y_2$ and $C_3^T X_3^\circ \le C_3^T Y_3$.
This implies that $(C_1^T X_1^\circ, C_2^T X_2^\circ, C_3^T X_3^o) \preceq (C_1^T Y_1, C_2^T Y_2, C_3^T Y_3)$.
That is, $\tilde{C}^T \otimes \tilde{X}^\circ \preceq \tilde{C}^T \otimes \tilde{Y}$.
Thus, $\tilde{X}^\circ$ is an optimal solution to the problem (P).
Hence the theorem is proved.

Remark 3.1: The converse part of the Theorem 3.1. is also true.

4. NEURAL NETWORK

Now, we construct an energy function which can make a neural network related to MOLP problem. The behaviour and the direction to search out the solution in a network are claimed by the energy function. For constructing the energy function, we have employed penalty function method such that the energy function has minimum at the solution of the problem.
Now, we define the required energy function as,

$$E(U) = \frac{1}{2}\sum_{i=1}^{3}\left\|C_i^T X_i\right\|^2 + \frac{k}{2}\sum_{i=1}^{3}\left\|A_i X_i - B_i\right\|^2 + \sum_{i=1}^{2}(X_i - X_{i+1})^T((X_i - X_{i+1}) - \left|X_i - X_{i+1}\right|) \quad (2)$$

where, $U = (X_i)_{i=1,2,3}$, $X_1 = (x_j)$, $X_2 = (y_j)$, $X_3 = (z_j)$, $j = 1,2,...,n$ $, k > 0$.
Note that, from the properties of convex functions [3], we can conclude that $E(U)$ is convex. We now, develop a neural network model for solving the fuzzy LP problem using the newly defined energy function $E(U)$. Now, since the energy function $E(U)$ attains its minimum at an optimal solution of MOLP problem and $E(U)$ is convex [12], we have to find the local minimum of the energy function. For finding local minimum of the energy function by the gradient search approach, we get the neural network model as given below.

$$\frac{dU}{dt} = -\nabla E(U) \quad (3)$$

Now, since $x - |x| = 2(x)^-$, where $(x)^- = \min(x,0)$. The network related to (3) becomes,

$$\left.\begin{aligned}\frac{dX_1}{dt} &= -[C_1(C_1^T X_1) + kA_1^T(A_1X_1 - B_1) + 2(X_1 - X_2)^-]\\ \frac{dX_2}{dt} &= -[C_2(C_2^T X_2) + kA_2^T(A_2X_2 - B_2) + 2(X_1 - X_2)^- + 2(X_2 - X_3)^-]\\ \frac{dX_3}{dt} &= -[C_3(C_3^T X_3) + kA_3^T(A_3X_3 - B_3) + 2(X_2 - X_3)^-]\end{aligned}\right\} \quad (4)$$

The above dynamical system (4) can be represented as a network as follows:

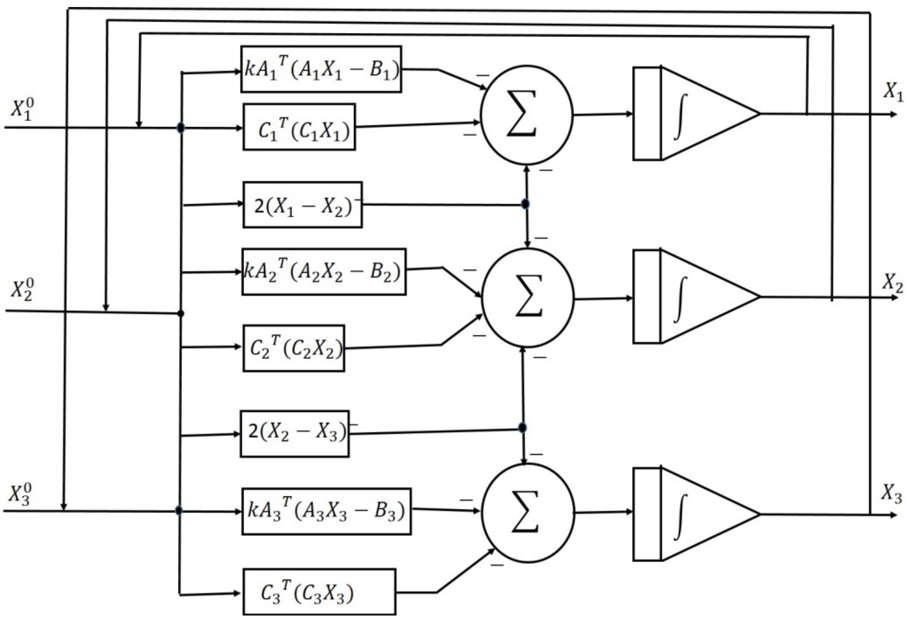

Fig.1 Representation of the network

Now, we prove the stability conditions for the newly defined neural network using (2) which are ensured the convergence of the solutions in the network:

Theorem 4. 1: $E(U)$ is a Lyapunov function for the proposed network (4) and the network is globally asymptotically stable in the sense of Lyapunov.

Proof: Now, since $k > 0$, is an arbitrary positive number, $E(U)$ is a positive for every U.

Now, $\frac{dE(U)}{dt} = \sum_{i=1}^{3} \frac{\partial E(U)}{\partial X_i} \cdot \frac{dX_i}{dt}$

$$=[C_1(C_1^T X_1)+kA_1^T(A_1X_1-B_1)+2(X_1-X_2)^-].\frac{dX_1}{dt}$$

$$+[C_2(C_2^T X_2)+kA_2^T(A_2X_2-B_2)+2(X_1-X_2)^-+2(X_2-X_3)^-].\frac{dX_2}{dt}$$

$$+[C_3(C_3^T X_3)+kA_3^T(A_3X_3-B_3)+2(X_2-X_3)^-].\frac{dX_3}{dt}$$

$$<0$$

Therefore, $E(U)$ is a Lyapunov function for the crisp network (4). Thus, the neural network (4) is globally asymptotic stability at the equilibrium $U=U^*$ in the sense of Lyapunov.
Hence the theorem is proved.

Remark 4.1: From the construction of the energy function (2), we can conclude that the proposed network can also be used for solving Multi-objective linear programming problems.

5. SAMPLE PROBLEMS

We, now present three different numerical examples to show the computational efficiency of the newly defined neural network for solving fuzzy LP problems. For solving MOLP problems in MATLAB with the proposed neural network, we use ode45.

Example 5.1: Consider the following fully fuzzy LP problem:
Max $(-1,2,3)\otimes\tilde{x}_1\oplus(2,3,4)\otimes\tilde{x}_2$
subject to
$(0,1,2)\otimes\tilde{x}_1\oplus(1,2,3)\otimes\tilde{x}_2\approx(2,10,24)$
$(1,2,3)\otimes\tilde{x}_1\oplus(0,1,2)\otimes\tilde{x}_2\approx(1,8,21)$
$\tilde{x}_1,\tilde{x}_2\succeq 0$.
Now, using zero as the initial state for all decision variables, we obtain the following solution by the proposed neural network model:
$\tilde{x}_1=(1.0000, 2.0035, 3.0129)$ and $\tilde{x}_2=(1.9920,\ 3.9876\ ,\ 5.9779)$.
The convergence of the solution of the problem is shown below (Fig-2.):

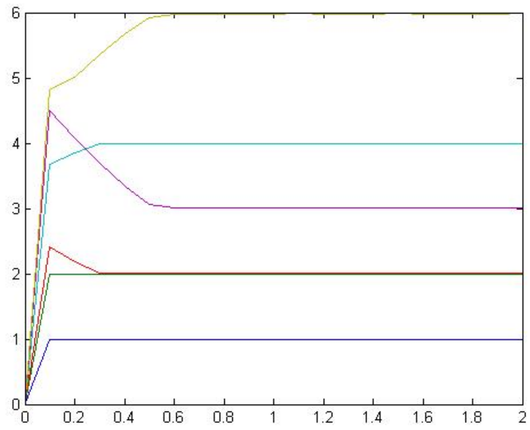

Fig-2 Convergence of the solution for the Example 5.1

Remark 5.1: The solution of the Example 5.1. by the proposed neural network is very close to the optimal solution, $\tilde{x}_1 = (1, 2, 3)$ and $\tilde{x}_2 = (2, 4, 6)$ obtained by bound and decomposition method [11].

Example 5.2: Consider the following fuzzy LP problem:
Max. $2\tilde{x}_1 \oplus 3\tilde{x}_2$
subject to
$\tilde{x}_1 \oplus 2\tilde{x}_2 \preceq (6,10,12)$
$2\tilde{x}_1 \oplus 1\tilde{x}_2 \preceq (4,11,14)$
$\tilde{x}_1, \tilde{x}_2 \succeq 0$.
Now, using zero as the initial state for all decision variables, we obtain the following solution by the proposed neural network model:
$\tilde{x}_1 = (0.6692, 4.0024, 5.3362)$ and $\tilde{x}_2 = (2.6612, 2.9891, 3.3208)$.
The convergence of the solution of the problem is shown below (Fig-3.):

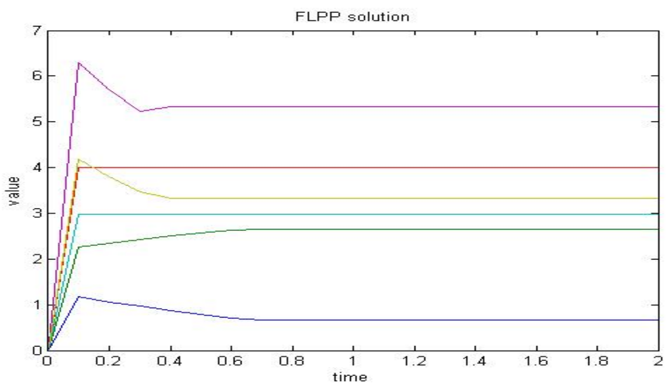

Fig-3 Convergence of the solution for the Example 5.2.

Remark 5.2: By the proposed neural network, the solution of the Example 5.2. is very close to the optimal solution, $\tilde{x}_1 = (0.67,\ 4,\ 5.33)$ and $\tilde{x}_2 = (2.67,\ 3\ ,3.33)$ obtained by bound and decomposition method [11].

Example 5. 3: Consider the following fully fuzzy LP problem:

Max. $(1,2,3) \otimes \tilde{x}_1 \oplus (2,3,4) \otimes \tilde{x}_2$

subject to

$(0,1,2) \otimes \tilde{x}_1 \oplus (1,2,3) \otimes \tilde{x}_2 \preceq (1,10,27)$

$(1,2,3) \otimes \tilde{x}_1 \oplus (0,1,2) \otimes \tilde{x}_2 \preceq (2,11,28)$

$\tilde{x}_1, \tilde{x}_2 \succeq 0$.

Now, using zero as the initial state for all decision variables, we obtain the following solution by the proposed neural network model:

$\tilde{x}_1 = (1.9974, 4.003, 6.009)$ and $\tilde{x}_2 = (0.9948, 2.9920, 4.9837)$.

The convergence of the solution of the problem is shown below (Fig-4.):

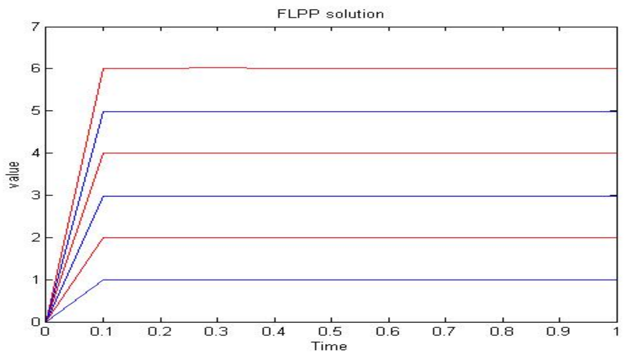

Fig-4 Convergence of the solution for the Example 5.3.

Remark 5.3: The solution of the Example 5.3. by the proposed neural network is very close to the optimal solution $\tilde{x}_1 = (2,\ 4\ ,6)$ and $\tilde{x}_2 = (1,\ 3,\ 5)$ obtained by bound and decomposition method [11].

6. CONCLUSION

In this paper, a fuzzy LP problem is solved by the crisp neural network. In this proposed approach, fuzzy LP problem can be considered as a MOLP problem and then, a neural network is developed using energy function which was constructed by penalty function for solving MOLP problem simultaneously. Based on the solution of the MOLP problem, we obtain a solution to the fuzzy LP problem which is realistic and applicable. We have showed the efficiency of the network by simple numerical examples. The proposed neural network approach is very much helpful to decision makers for solving LP problems in fuzzy environment having large number of parameters.

REFERENCES

[1] Amit Kumar, Jagdeep Kaur and Pushpinder Singh, 2011, “A new method for solving fully fuzzy linear programming problems”, Applied Mathematical Modeling, 35, pp.817-823.

[2] Bellman, R. E. and Zadeh, L. A., 1970, “Decision-making in a fuzzy environment”, Management Science, 17, pp.141–161.

[3] Boyd, S. and Vandenberghe, L., 2004, “Convex Optimization”, Cambridge University Press, New York.

[4] Cichocki, A., Unbehauen, R., Weinzierl, K. and Htlzel, R., 1996, “A new neural network for solving linear programming problems”, European Journal of Operational Research, 93, pp. 244-256.

[5] Ebrahimnejad, A., Nasseri, S. H., Hosseinzadeh Lotfi, F. and Soltanifar, M., 2010, “A primal-dual method for linear programming problems with fuzzy variables”, Eur. J. Ind. Eng., 4, 189–209.

[6] Effati, S. and Nazemi, A. R., 2006, “Neural networks models and its application for solving linear and quadratic programming problems”, Appl. Math. Comput., 172, pp.305–331.

[7] Effati, S., Pakdaman, M. and Ranjbar, M., 2011, “A new fuzzy neural network model for solving fuzzy linear programming problems and its applications”, Neural Computation and Application, 20, pp.1285–1294.

[8] Ganesan, K. and Veeramani, P., 2006, “Fuzzy linear programs with trapezoidal fuzzy numbers”, Ann. Oper. Res., 143, pp.305–315.

[9] Gao, X. and Liao, L. Z., 2010, “A new one-layer neural network for linear and quadratic Programming”, IEEE Transactions on Neural Networks, 21, pp. 918-929.

[10] Ghasabi-Oskoei, H., Malek, A. and Ahmadi, A., 2007, “Novel artificial neural network with simulation aspects for solving linear and quadratic programming problems”, Computers and Mathematics with Applications, 53, pp. 1439–1454.

[11] Jayalakshmi, M. and Pandian, P., 2012, “A new method for finding an optimal fuzzy solution for fully fuzzy linear programming problems”, International Journal of Engineering Research and Applications, 2, pp.247-254.

[12] Kennedy, M. P. and Chua, L. O., 1988, “ Neural networks for nonlinear programming”, IEEE Transactions on Circuits and Systems, 35, pp. 554-562.

[13] Maa, C. and Shanbaltt, M., 1992, “A two-phase optimization neural network”, IEEE Trans. On Neural Networks, 3, pp.1003-1009.

[14] Madan M Gupta, Liang Jin, and Noriyasu Homa, 2003, “ Static and Dynamical Neural Networks: From Fundamentals to Advanced Theory”, John Wiley & Sons Inc., New Jersey.

[15] Malek, A. and Alipour, M., 2007, “Numerical solution for linear and quadratic programming problems using a recurrent neural network”, Applied Mathematics and Computation, 192, pp. 27–39.

[16] Malek, A. and Yari, A., 2005, “Primal–dual solution for the linear programming problems using neural networks”, Applied Mathematics and Computation, 167, pp. 198–211.

[17] Nguyen, K. V., 2000, “A nonlinear neural network for solving linear programming problems”, In: Proceedings of the 17th International Symposium on Mathematical Programming, Atlanta, Georgia.
[18] Pandian, P., 2013, “Multi-objective programming approach for fuzzy linear programming problems”, Applied Mathematical Sciences, 7, pp. 1811 – 1817.
[19] Pehlivan, N. Y. and Apaydin, A., 2005, “Artificial neural networks approach to fuzzy linear programming”, Selcuk journal of applied mathematics, 6, pp.9-26
[20] Shaocheng, T. 1994, “Interval number and fuzzy number linear programming”, Fuzzy Sets Syst, 66:301–306
[21] Tanaka, H. and Asai, K., 1984, “Fuzzy linear programming with fuzzy numbers”, Fuzzy Sets. Syst., 13, pp.1–10.
[22] Tank, D. W. and Hopfield, J. J., 1986, “Simple neural optimization networks: An A/D converter, signal decision circuit and a linear programming circuit”, IEEE Transactions on circuits and systems (CAS), 33, pp. 533-541.
[23] Xia, Y., 1996, “A new neural network for solving linear programming problems and its application”, IEEE Transactions on Neural Networks, 7, pp. 525-529.
[24] Yinzhen, L. Mitsuo, G. and Kenichi, I., 1996, “Solving fuzzy shortest path problems by neural network”, Comput. Eng., 31, pp.861–865.
[25] Zadeh, L.A., 1965, “Fuzzy Sets”, Information and Control, 8, pp.338-353.
[26] Zimmerman, H. J., 1983, “Fuzzy mathematical programming”, Comput. Ops. Res., 10, pp.291–298