Functional Networks for CAD problems

B. Chandra and S.P. Singh
Department of Mathematics
Indian Institute of Technology,
Hauz Khas, New Delhi 110 016.
bchandra(S)maths.iitd.emet,in

Abstract - There are some basic real life problems that cannot be solved using classical mathematical techniques. In this paper Functional Networks have been effectively used to solve practical CAD problems related to plant engineering Industry. Modular Construction of plants is becoming popular due to severe weather conditions at plant sites. The modules are transported to and assembled at the actual plant site. The temporary structure should be safe during lifting. For this, it is essential to find the rotation position of the model once it is lifted. This rotation position will depend on the center of gravity of the module and the center of rotation about which the module will rotate. If cables meet at a point then this will be the point of rotation. If they do not meet then there is no classical mathematical technique available to find the center of rotation. In this paper functional networks have been successfully applied to solve this problem.

I. INTRODUCTION

Neural Networks has been widely used for solving variety of problems. Back-propagation Network [8] has been extensively used for several applications. But the standard neural network paradigm has certain limitations. Even extensions of neural networks like Higher Order Networks [6], Probabilistic Neural Networks [7], The Fuzzy Neural Networks [5] etc. are same as standard neural networks as far as the functional properties of the function being approximated are concerned. These approaches treat neural networks as black boxes and do not take in to account the functional structure and the functional properties of the domain.

However functional networks, a recent generalization of neural networks combine the domain properties, to determine the structure of the network along with the data, to determine the unknown function. In functional networks, arbitrary neural functions are allowed and initially these functions are assumed to be multi-argument and vector-valued. Functional properties and functional constraints of the domain are defined in the form of functional equations and it is tried to simplify the initial network. Then the simplified network is trained using the data to find out the unknown function.

There are several problems, which can not be modeled using the standard neural networks but can be solved using the functional networks. In this paper it has been shown how Functional Networks can be effectively used for solving CAD problems since these problems generally have well defined physical properties. The chosen CAD problem is highly useful for the plant engineering industry. It involves finding the point about which a temporary structure rotates when it is lifted by cranes. If the cables used to lift the module are physically or virtually meeting at a point then that will be the point of rotation however there is no way to find the point of rotation when cables are not meeting at a point. This problem has been tackled using Functional Networks.

II. FUNCTIONAL NETWORKS - AN OVERVIEW

Functional Networks is an advancement of Neural Networks [2]-[4]. In functional networks domain knowledge is used to determine the structure of network along with data to determine the unknown neuron function. In functional networks there are two types of learning to take care of the domain knowledge and the data:

(a) Structural Learning:

Structural learning includes finding out the initial topology of the network and simplifying the initial network using functional equations.

(b) Parametric Learning:

Parametric learning deals with the estimation of the neuron function using the available data.

Formally, functional networks can be defined as a pair (X, Y), where X is set of nodes and U = {(Y[, Fj, Zj)| 1 = 1 — n } is a set of functional units (neurons) over X such that every node Xj e X is either an input or an output node in at least one functional unit in U.

Functional networks is an extension of neural networks where the activation functions are unknown functions from a given a family, to be estimated during the learning process. Functional networks extend neural networks by allowing neurons to be not only true multi-argument and multivariate functions, but to be different and learnable, instead of fixed functions. In addition, functional networks allow putting constraints on the network by forcing more than one neuron to return the same output.

Working with functional networks involves following steps:

1. First step involves understanding the problem under consideration.
2. Then based on the domain knowledge, an initial topology is selected for the functional network.
3. Functional constraints of the problem are expressed in the form of functional equation [1] and then the initial network is simplified by solving these functional equations.

4. Before learning the unknown functions, we should ensure that there is a unique representation of the given network.
5. This step involves the collection of training data.
6. Linear combination of appropriate function families and using some minimization function to obtain the optimal value of coefficients.
7. Finally the model is validated and if results are found satisfactory it is ready for use.

Generally the functional network is constructed and then simplified based on the problem but some standard functional network models are also available which can be used if the domain properties are satisfied by any of these models. 'The Uniqueness Model', 'The Generalized Associativity Model', 'The Separable Model', 'The Generalized Bisymmetry Model' and "The Serial Function Model' are some of the standard models. Some problems may be solved by using a combination of standard models also.

Although, functional networks is a very new field and not much work has been done yet functional networks can be used to solve many problems, which can not be solved otherwise.

HI. FUNCTIONAL NETWORK FOR A CAD PROBLEM

Functional networks are very useful in Engineering and CAD problems because these problems generally have well defined physical properties. We have selected a practical problem from the field of CAD for which no classical mathematical solution is available.

The problem under consideration is a general CAD problem that has become a practical problem for the plant engineering industry. Nuclear plants are generally constructed in remote areas. Construction and erection costs are significantly higher in these areas because of higher labor costs and severe weather conditions. These factors affect the overall cost and construction schedule of the project. To solve this problem, plant engineers are opting for 'Modular Construction' of the plants. In modular construction, plant is constructed in small volumes called modules. This construction is carried out at a different location with favorable conditions. Then these modules are shipped to the actual plant site. There these modules are combined together. This approach of construction provides reduced construction costs, improved fabrication quality and better construction schedules. But it needs the transportation of module from the fabrication location to the actual plant sites. Transportation involves lifting the module using cranes and then shipping it to the desired location. Her it is required to be ensured that module will remain safe during lifting. For this lifting analysis of the module, it is essential to find the rotated position of the model when it is lifted. This rotated position will depend upon the C.G. of the module and the center of rotation about which the module will rotate. If cables used to lift the module are arranged in such a way that these cables physically or virtually meet at a point then obviously that particular point will be point of rotation. But this is only an ideal condition as most of the times it is not possible to provide cables in ideal way because of interference of cables with the plant commodities.

If cables are not meeting at a point then there is no classical mathematical technique to find the center of rotation. Till now this problem is being taken care of by gradually varying the cable lengths in the field, till the module is balanced. This is the problem; we have solved using the functional networks. The developed functional network takes coordinates of ends of the cables as input and returns the coordinates of the point of rotation.

IV. IMPLEMENTATION

In this section, we will discuss our approach towards the solution of mentioned problem using functional networks.

Problem Description:

There are nine points in the system. These are eight input points Pj (i = 1 to 8) and an output point O.

Pi = Bottom point of cable \\I2\ for odd i's.
Pi = Top point of cable i/2 for even i's .
O = Required output that is the point about which the system rotates.

Initial Topology:

We know that required output is function of eight input points i.e.

$$O = h(P_1,P_2,P_3,P_4,P5,P6,P7,P_8) \quad (1)$$

Further from the domain knowledge, we know that if a small rotation about the point O is given to the system, relative distance between any two points in the system remains unchanged. Let after rotation, points P; take transformed positions P'j; however point O remains unchanged even after the rotation because system has rotated about the point O itself. For the displaced system we can say that:

$$O = h\,(P^1,, P'_2, P'_3, P'_4, P'5, P'_6, P'7, P_8) \quad (2)$$

Here P'; = r(O, P'i), where r(a,b) is the known transformation function for rotation of point b about point a.
From equations (1) and (2), we get equation (3) as:

$$h(P,, P_2, P_3, P_4, P_5, P_6, P_7, Pg) - h\,(P,, P'_2, P'_3, P'4, P'S, P'6, PV, P'K) = 0 \quad (3)$$

One way of coming up with the solution is that we solve the equation (3) to take care of the rotation property of the domain. But this will be quite cumbersome to solve this functional equation because this is cyclic. Hence we have come up with a simpler innovative idea of'Extension of Training Data' to take into account rotation property of the domain. It involves extending the training data by generating six new sets of data for each given set of data. This is achieved by rotating the given

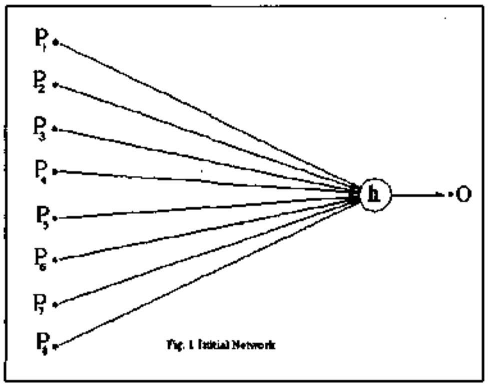

Fig. 1 Initial Network

input points about the given output point by ± 0.1 radian about x, y and z-axis. So we can continue with the initial topology suggested by equation (1) only as shown in figure 1.

Simplification of the Network:

Initial network consists of a single function h, but this function is multi-argument having 8 arguments. We can get same quality of results even if we proceed with the initial network of single function only. But it is always preferable to have a network with simplified functions to save the computation effort involved in learning the function.

Let us assume that point C_1, C_2, C_3 and C_4 are points returned by cables 1, 2, 3 and 4 respectively as the point of rotation, then we can say that:

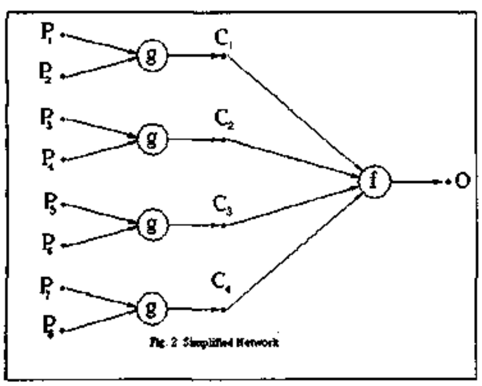

Fig. 2 Simplified Network

$$O = f(C_1, C_2, C_3, C_4) \quad (4)$$

where,

$$C_i = g(P_{2i-1}, P_{2i}) \quad (5)$$

Equation (4) and (5) are an alternative to the initial approach and suggest the simplified network shown in fig. 2.

Data Generation:

We need data to train the model. For the problem under consideration, we know that there is a particular case for which the solution is known i.e. the case in which all the cables are actually or virtually meeting at one point. So we can take data from such cases. Data was generated by writing a program, which randomly fixes the four bottom points of the cables and the top point (O) at which all these cables are meet Then top point of a cable can be found by randomly selecting a point lying on the line joining the point O with bottom of the cable. Using this approach one thousand data sets were generated.

Learning the Model:

We need to learn two functions. These are double argument function g and four argument function f. Since f will be a linear combination of points returned by four cables, hence we can assume f to be a linear function but g will be a quadratic function because it will depend upon length and slope etc. of the cable.

Let us assume that:

$$g(P, Q) = A_o + A_1P + A_2P^2 + A_3PQ + A_4Q + A_5Q^2$$

where P is the bottom point and Q is the top point of the cable. Since P and Q are 3-dimensional so g(P,Q) can be written as

$$g(P,Q) = \begin{bmatrix} V \\ \% \\ \end{bmatrix} + \begin{bmatrix} a_{1x}P_x \\ a_{1y}P_y \\ a_{1z}P_z \end{bmatrix} + \begin{bmatrix} a_{2x}P_x^2 \\ a_{2y}P_y^2 \\ a_{2z}P_z^2 \end{bmatrix} + \begin{bmatrix} a_{3x}P_xQ_x \\ a_{3y}P_yQ_y \\ a_{3z}P_zQ_z \end{bmatrix} + \begin{bmatrix} a_{4x}Q_x \\ "fa \\ \end{bmatrix} + \begin{bmatrix} a_{5x}Q_x^2 \\ if \\ \end{bmatrix} \quad (6)$$

where P_x, P_y and P_z are x, y & z coordinates of point P and Q_x, Q_y and Q_z are x, y & z coordinates of point Q respectively. If P and Q are both origins then g(P, Q) should return origin only as the output. It implies that A_o is redundant

i.e. $a_{ox} = a_{oy} = a_{o2} = 0$ and

$$g(P,Q) = \begin{bmatrix} a_{1x}P_x \\ a_{1y}P_y \\ a_{1z}P_z \end{bmatrix} + \begin{bmatrix} a_{2x}P^2{}_x \\ a_{2y}P^2{}_y \\ a_{2z}P^2{}_z \end{bmatrix} + \begin{bmatrix} a_{3x}P_xQ_x \\ a_{3y}P_yQ_y \\ a_{3z}P_zQ_z \end{bmatrix} . \begin{bmatrix} \\ \%Qy \\ a_{4z}Q_z \end{bmatrix} * \begin{bmatrix} \backslash 0, \\ a_{5y}Q^2{}_y \\ a_{5z}Q^2{}_z \end{bmatrix} \quad (7)$$

Let

f(C,, C_2, C_3, C_4) = A_6 + A_7 C_1+ A_8 C_2+ A_9 C_3+ Aio C_4 (8)

Ci, C_2, C_3 and C4 are 3-dimensional points so f can be expressed as:

If C, = C_2 = C_3 = C_4= C then

f(C,C,C,C) = C

If C is origin then equation (8) implies that A6 is zero i.e.

a«x = aey = a«$_z$ = 0 and

$$f(C_1,C_2,C_3,C_4) = \begin{bmatrix} a_{7x}C_{1x} \\ a_{7y}C_{1y} \\ a_{7z}C_{1z} \end{bmatrix} + \begin{bmatrix} a_{8x}C_{2x} \\ a_{8y}C_{2y} \\ a_{8z}C_{2z} \end{bmatrix} + \begin{bmatrix} a_{9x}C_{3x} \\ a_{9y}C_{3y} \\ a_{9z}C_{3z} \end{bmatrix} + \begin{bmatrix} a_{10x}C_{4x} \\ a_{10y}C_{4y} \\ a_{10z}C_{4z} \end{bmatrix}$$

$$f(C,C,C,C) = \begin{bmatrix} a_{7x}C_x \\ a_{7y}C_y \\ a_{7z}C_z \end{bmatrix} + \begin{bmatrix} a_{8x}C_x \\ \%^c y \\ a_{8z}C_z \end{bmatrix} + \begin{bmatrix} a_{9x}C_x \\ a_{9y}C_y \\ a_{9z}C_z \end{bmatrix} + \begin{bmatrix} a_{10x}C_x \\ a_{10y}C_y \\ a_{10z}C_z \end{bmatrix} = \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} \quad (9)$$

But all the cables have same weightage, so

a7x = a_{8x} = a9X = ai$_{Ox}$,

a7y = a_{8y} = $a9_y$ = ajoy

a7z = a_{8z} = a_{9z} = ai$_{Oz}$

⇒

$$\begin{bmatrix} 4«7,c; \\ 4a_{7y}C_y \\ 4a_{7z}C_z \end{bmatrix} = \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix} \quad (10)$$

=>$a_{7x} = a_{7y} = a_{7z} = 1/4$

So function f reduces to following form

f(C, C_2, C_3, C_4) = l/4(d+ C_2+ C_3+ C_4)

ButC$_i$ = g(P_{2i-1},P_{2i})

=> fl[C,, C_2, C, C_4) = l/4(g(P$_1$, P_2) + g(P_3, P_4) + g(P$_{S)}$ P_6) + g(P_7, Ps))

=> KCu C_2, C_3, C_4) = l/4(g(P$_1$, P_2) + g(P_3, P_4) + g(P_5, P_6) + g(P_7, P_8))

=> BiA, +B_2A_2 + B_3A_3 + B_4A_4 + B_4A_5 = 4O

where,

B, = P, + P_3 + P_5 + P_7

B_2=P^2, + $P^2{}_3 + P^2{}_5 + P^2{}_7$

B_3=P_1P_2 + $P_3P_4 + P_5P_6 + P_7P_8$

B_4=P_2 + $P_4 + P_6 + P_8$

$B_5 = P^2{}_2 + P^2{}_4 + P^2{}_6 + P^2{}_8$

There are 1000 sets of training data out of which five hundred were used for learning the model. Rest of the data was used for validation. Error for data set of data can be measured as:

ej = B,jA, + $B_{2j}A_2$ + $B_{3j}A_3$ + $B_{4j}A_4$ + $B_{4j}A_5$-4Oj (11)

Sum of square of errors = Q = $\sum e^2_j$

We can find the optimal values of coefficients a_1; a_2, a_3, a» and a_5 by differentiating Q with respect to each of the coefficient and then solving the resulting system of simultaneous equations as follows:

DiiAj -l- $D_{12}A_2$Hh $D_{13}A_3$ + $D_{14}A_4$ + $D_{15}A_5$ = K_1
D_{21}A, -H $D_{22}A_2$ -I- $D_{23}A_3$ -(- D_{24}A4 + D_5A_5 = K_2
D_{31}A, -i-$D_{32}A_2$Hh $D_{33}A_3$ -I- D_{34}A4 + D_{35}A5 = K_3
D_4,A, -I- $D_{42}A_2$ -^$D_{43}A_3$-1- D^A^ + $D_{45}A_5$ = K4
DsiA] -I- $D_{52}A_2$H^ $D_{53}A_3$ -i-$D_{54}A_4$ + $D_{55}A_5$ = K5 **(12)**

where,

$D_{ij} = D_{ji} = C_iC_j$ and Kj =4QO

VI. RESULTS

Results obtained are quite encouraging. After training the model, ten points were randomly selected from the validation data and their point of rotation was computed using the code developed in Java.. Maximum difference between the actual and computed values of coordinates of point of rotation was found to be less than 0.1 percent. Input, actual point of rotation and point of rotation returned by functional network have been presented in Table 1.

Comparison of coordinates of actual and computed points of rotations for these ten points has been presented in graphical form in Fig. 3,4 and 5.

#		Cable 1		Cable 2		Cable 3		Cable 4		Point of Rotation	
		Top	**Bottom**	**Top**	**Bottom**	**Top**	**Bottom**	**Top**	**Bottom**	**Actual**	Network
1.	X	8.84	4.60	10.41	12.43	10.52	13.01	8.76	4.20	9.900	9.900
	Y	7.47	0.21	7.49	0.31	7.61	0.90	7.70	1.33	9.290	9.288
	Z	4.90	0.63	4.98	1.00	6.56	8.93	6.46	8.41	5.970	5.972
2.	**X**	8.50	5.22	10.26	14.03	10.40	14.72	8.42	4.82	9.320	9.320
	Y	7.92	1.92	7.94	2.02	7.84	1.52	8.12	2.93	9.420	9.419
	Z	5.43	2.23	5.31	1.62	7.09	10.53	7.01	10.12	6.230	6.231
3.	**X**	8.63	4.94	10.20	12.78	10.23	12.91	8.55	4.54	9.560	9.553
	Y	7.53	0.11	7.55	0.21	7.76	1.24	7.85	1.68,	9.390	9.388
	Z	4.69	0.98	4.76	1.34	6.35	9.28	6.16	8.31	5.620	5.617
4.	**X**	9.00	5.72	10.67	14.03	10.80	14.71	8.92	5.32	9.820	9.822
	Y	7.91	1.91	7.93	2.01	7.93	2.02	8.11	2.93	9.410	9.407
	Z	5.43	2.23	5.41	2.12	7.09	10.53	7.01	10.11	6.230	6.231
5.	**X**	9.16	5.88	10.70	13.58	10.85	14.35	**9.08**	5.48	9.980	9.979
	Y	7.55	1.55	7.57	1.65	7.67	2.18	7.73	2.48	9.050	9.046
	Z	4.98	1.78	5.08	2.28	6.64	10.08	6.57	9.75	5.780	5.779
6.	**X**	9.45	6.17	10.85	13.19	11.13	14.57	9.37	5.77	10.270	10.270
	Y	7.77	1.77	7.79	1.87	7.91	2.47	7.83	2.09	9.270	9.268
	Z	4.59	1.39	4.83	2.57	6.25	9.69	6.31	9.97	5.390	5.394
7.	**X**	8.93	4.45	10.57	12.62	10.66	13.09	8.85	4.05	10.050	10.052
	Y	7.43	0.29	7.45	0.39	7.52	0.75	7.67	1.52	9.210	9.213
	Z	4.79	0.82	4.80	0.85	6.45	9.12	6.33	8.49	5,780	5.785
8.	**X**	9.16	5.88	10.73	13.72	10.92	14.67	9.08	5.48	9.980	9.980
	Y	7.87	1.87	7.89	1.97	7.93	2.18	8.02	2.62	9.370	9.369
	Z	5.12	1.92	5.19	2.28	6.78	10.22	6.75	10.07	5.920	5.919
9.	**X**	8.75	4.75	10.39	12.99	10.40	13.00	8.67	4.35	9.750	9.746
	Y	7.48	0.20	7.50	0.30	7.65	1.05	7.82	1.89	9.300	9.301
	Z	4.57	1.19	4.56	1.15	6.23	9.49	6.01	8.40	5.410	5.413
10.	**X**	8.59	5.31	10.19	13.30	10.31	13.94	8.51	4.91	9.410	9.410
	Y	7.14	1.14	7.16	1.24	7.23	1.61	7.35	2.20	8.640	8.638
	Z	4.70	1.50	4.74	1.71	6.36	9.80	6.27	9.34	5.500	5.500

TABLE 1

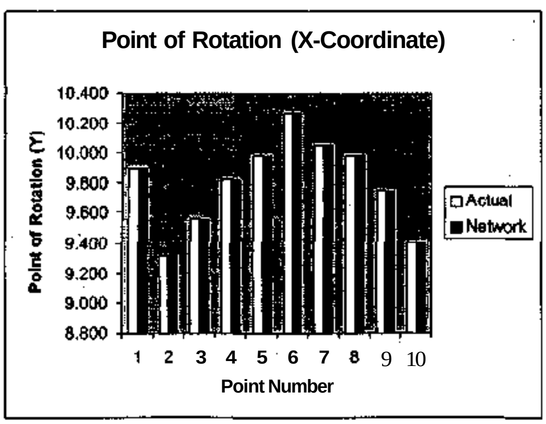

Fig3

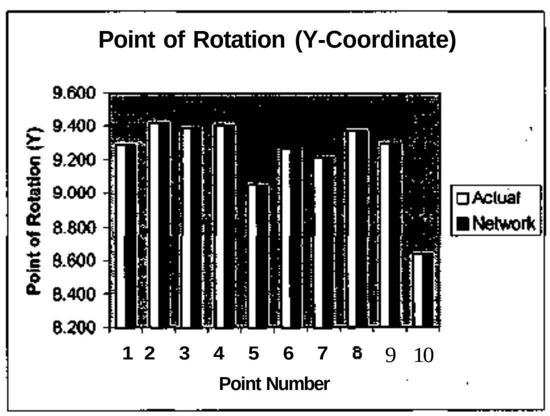

Fig. 4

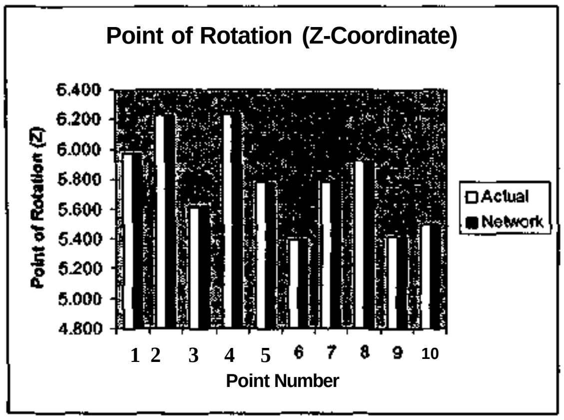

Fig. 5

VII. CONCLUSIONS

The results show how Functional Networks can be effectively employed for solving CAD problems especially when no classical mathematical technique is available. The advantage of functional Networks is that it takes the domain knowledge also into account.

References

[1] J Aczel , "Functional Equations in Various Variables", *Cambridge Univ. Press,* 1989.

[2] E Castillo and J M Gutie'rrez , Nonlinear Time Series Modeling and Prediction Using Functional Networks Extracting Information Masked by Chaos", *Physical Letters A,* Vol. 244, 71-84, 1998.

[3] E Castillo, J M Gutie'rrez and A S Haidi, "A Comparison of Functional Networks and Neural Networks", *Proceedings of the IASTED International Conference on Artificial Intelligence and Soft Computing,* IASTED/ACTA press, 439-442, 1998.

[4] E Castillo, A Cobo, J M Gutie'rrez and E Pruneda, Introduction to Functional Networks with Applications - A Neural Based Paradigm", *Kluwer International Publishers,* 1998.

[5] M. M. Gupta and D. H. Rao, "On the principles of fuzzy Neural networks," *Fuzzy Sets and Systems,* 61, 1 - 18 (1994)

[6] Y. C. Lee, G. Doolen, H. Chen, G. Sun, T. Maxwell, H. Lee, and C. L. Giles, "Machine learning using a higher order correlation network", *Physica D: Nonlinear Phenomena,* vol. 22, pp. 276-306, 1986.

[7] D F Specht , "Probabilistic Neural Networks and the Polynomial Adaline as a Complementary Technique for Classification", *IEEE Transactions on Neural Networks,* 1(1):111-121 March 1990.

[8] P.J.Werbos, "Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences.", Ph.D. thesis, Harvard University, 1974.