The GMDH Algorithm of Ivakhnenko

Stanley J. Farlow

The American Statistician, Vol. 35, No. 4 (Nov., 1981), 210-215.

Stable URL:
http://links.jstor.org/sici?sici=0003-1305%28198111%2935%3A4%3C210%3ATGAOI%3E2.0.CO%3B2-4

The American Statistician is currently published by American Statistical Association.

The GMDH Algorithm of Ivakhnenko

STANLEY J. FARLOW*

In 1966, the Russian cyberneticist, A.G. Ivakhnenko, introduced a technique for constructing an extremely high-order regression-type polynomial. The algorithm, the Group Method of Data Handling (GMDH), builds a multinomial of degree in the hundreds, whereas standard multiple regression becomes bogged down in computation and linear dependence. The GMDH method is ideal for complex, unstructured systems where the investigator is only interested in obtaining a high-order input-output relationship. Also, the Ivakhnenko algorithm is heuristic in nature and is not based on a solid foundation as is regression analysis.

KEY WORDS: GMDH; Ivakhnenko.

1. INTRODUCTION

A major difficulty in modeling complex systems in such unstructured areas as economics, ecology, sociology, and others is the problem of the researcher introducing his or her own prejudices into the model. Since the system in question may be extremely complex, the basic assumptions of the modeler may be vague guesses at best. It is not surprising that many of the results in these areas are vague, ambiguous, and extremely qualitative in nature. It was for this reason that in the mid 1960's the Russian mathematician and cyberneticist, A.G. Ivakhnenko, introduced a method (Ivakhnenko 1966), based in part on the Rosenblatt Perceptron (Rosenblatt 1958), that allows the researcher to build models of complex systems without making assumptions about the internal workings. The idea is to have the computer construct a model of *optimal complexity* based only on data and not on any preconceived ideas of the researcher; that is, by knowing only simple input-output relationships of the system, Ivakhnenko's Group Method of Data Handling (GMDH) algorithm will construct a self-organizing model (an extremely high-order polynomial in the input variables) that can be used to solve prediction, identification, control synthesis, and other system problems. This tutorial will describe Ivakhnenko's basic algorithm and illustrate how this technique can be used to solve complicated system problems.

* Stanley J. Farlow is Associate Professor, Department of Mathematics, University of Maine, Orono, ME 04473. The author is grateful to Tom Probert of the Environmental Data and Information Services, U.S. Department of Commerce, for first introducing this technique to the author. The author would also like to thank the referees and editor for helpful suggestions.

2. BASIC GMDH OVERVIEW

The basic GMDH algorithm is a procedure for constructing a high-order polynomial of the form

$$y = a + \sum_{i=1}^{m} b_i x_i + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_i x_j + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} d_{ijk} x_i x_j x_k + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m}\sum_{l=1}^{m} e_{ijkl} x_i x_j x_k x_l + \cdots, \quad (2.1)$$

which relates m input variables $x_1, x_2, \ldots, x_m$ to a single output variable y. Although (2.1) resembles a high-order regression polynomial, the manner in which the polynomial (which we will call the Ivakhnenko polynomial) is constructed differs from the techniques of standard regression analysis. In fact, it was said (Scott and Hutchinson 1976) that the procedure for constructing the Ivakhnenko polynomial is similar to the way in which nature evolves by natural selection. This procedure will now be described.

3. THE GMDH NUMERICAL ALGORITHM

The basic information one must gather to construct the Ivakhnenko polynomial is a set of n observations like the ones shown in Figure 1. In this figure,

$nt \equiv$ number of observations in the training set (explained later)
$nc \equiv n - nt =$ number of observations in the checking set (explained later)
$n \equiv$ total number of observations
$m \equiv$ number of variables.

The reason for dividing the observations into two distinct sets will be explained shortly. We now describe the basic steps of the GMDH algorithm.

Step 1 (constructing new variables $z_1, z_2, \ldots, z_{\binom{m}{2}}$). The first step is very simple. We take all the independent variables (columns of the array X) $x_1, x_2, \ldots, x_m$ two at a time and for each of these $\binom{m}{2}$ combinations find the least squares polynomial of the form

$$y = A + Bu + Cv + Du^2 + Ev^2 + Fuv \quad (3.1)$$

that best fits the observations y_i in the training set; that is, we find those $\binom{m}{2} = m(m-1)/2$ polynomial surfaces like those illustrated in Figure 2. Now, for each of the $\binom{m}{2}$ polynomial surfaces shown in Figure 2 evaluate the polynomial at the n data points. For example, for the first surface shown above we would evaluate $y = A + Bx_1 + Cx_2 + Dx_1^2 + Ex_2^2 + Fx_1x_2$ at the data points $(x_{11}, x_{12}), (x_{21}, x_{22}), \ldots, (x_{n1}, x_{n2})$ and store these n values in the first column of an array Z. The remaining $\binom{m}{2} - 1$ columns are constructed in a simi-

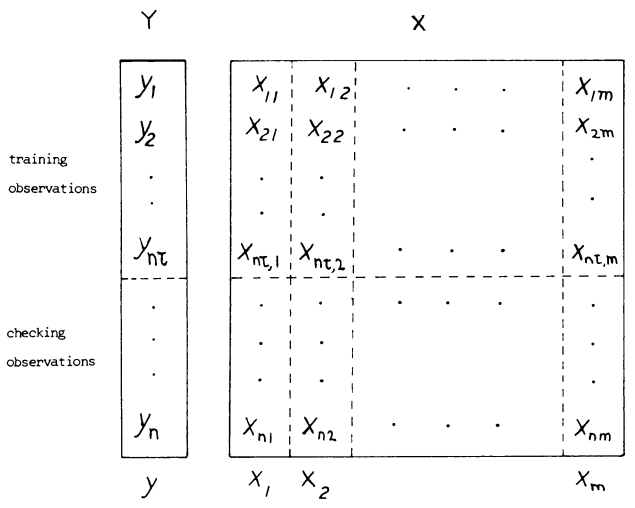


Figure 1. Input to the GMDH Algorithm

lar way (see Figure 3). One should think of these $\binom{m}{2}$ columns of Z as $\binom{m}{2}$ observations of new variables $z_1, z_2, \ldots, z_{\binom{m}{2}}$ where each of these new variables is a polynomial in the original variables $x_1, x_2, \ldots, x_m$. In other words, we have constructed new variables, some of which will replace the original ones. The object is to keep those z_i^s that best estimate the output vector y (and throw the others away). Here is where the checking set enters the algorithm.

Step 2 (screening out z's that are not important). This step replaces the columns of X (old variables) by those columns of Z (new variables that are quadratic polynomials in the old variables) that best predict the observed vector Y. More specifically, for each column j of Z, we compute the *least square error* d_j computed by

$$d_j^2 = \sum_{i=nt+1}^{n} (y_i - z_{ij})^2 \quad j = 1, 2, \ldots, \binom{m}{2},$$

order the columns of Z according to an increasing least square error d_j, and then pick those columns satisfying $d_j < M$ (M some prescribed number) to replace the columns of X (those columns of Z will be the new X). It is important to note here that the number, say m_1, of these new replacement columns may be larger or smaller than the previous number of variables (which was originally m). Also note that the least square errors d_j are summed over the checking observations.

Step 3 (test for convergence). From Step 2 take the smallest of the d_j's, call it DMIN, and plot it on a graph like the one shown in Figure 4. If the value of DMIN is smaller than the previous DMIN (the first time we get to this step we assume this to be true) go back and repeat Steps 1 and 2. If the value of DMIN is greater than the previous DMIN we stop the process (and use the results for the previous minimum value of DMIN). Standard regression uses only the mean-squared error as a criterion and hence it is not possible to determine if the model is too simple or too complex. By testing the Ivakhnenko polynomial against the checking set (universe), it is possible to get a unique polynomial of

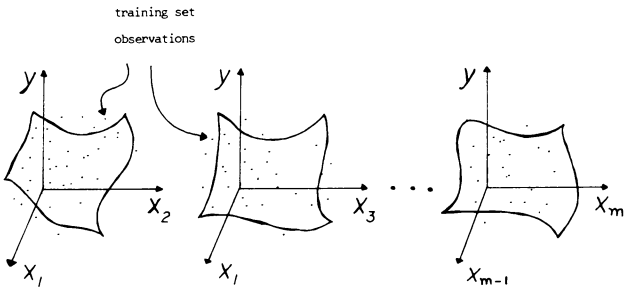


Figure 2. Construction of the New Variables

optimal complexity. Although the theory of the GMDH algorithm does not allow us to prove that the DMIN curve has a single minimum value, all experimental results indicate that this is true.

The general nature of the DMIN curve is that it initially decreases to a minimum and then starts to increase. For the example in Figure 4 we would stop the process after the fifth iteration. This is the end of the Ivakhnenko algorithm. The first column of the Z array will now contain the values $\bar{y}_i$ of the Ivakhnenko polynomial

$$\bar{y}_1 = a + \sum_{i=1}^{m} b_i x_{1i} + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_{1i} x_{1j} + \cdots$$

$$\bar{y}_2 = a + \sum_{i=1}^{m} b_i x_{2i} + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_{2i} x_{2j} + \cdots$$

$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots$$

$$\bar{y}_n = a + \sum_{i=1}^{m} b_i x_{ni} + \sum_{i=1}^{m}\sum_{j=1}^{m} c_{ij} x_{ni} x_{nj} + \cdots$$

evaluated at the original n data points. In other words, the first column should correlate highly to the observed dependent variable y.

In order to find the desired values of the coefficients $a, b_i, c_{ij}, d_{ijk}, e_{ijkl}, \ldots$ in the Ivakhnenko polynomial (2.1), one must save all the regression surfaces $y = A + Bu + Cv + Du^2 + Ev^2 + Fuv$ that were computed at each iteration and systematically evaluate these polynomials of polynomials of polynomials, and so forth, until one eventually gets the Ivakhnenko polynomial. This is a nontrivial programming problem but can be carried out nevertheless. To give the reader more insight to the algorithm we have added a few comments.

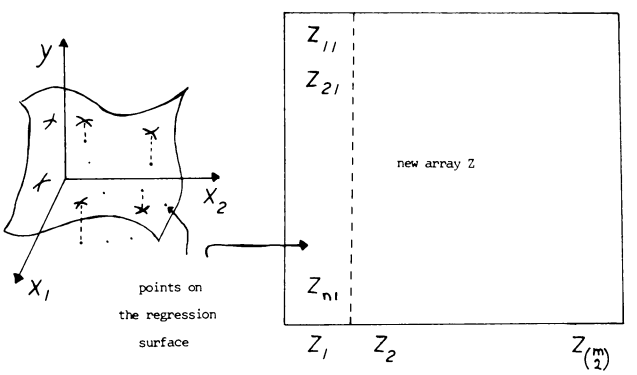


Figure 3. Construction of the New Array Z

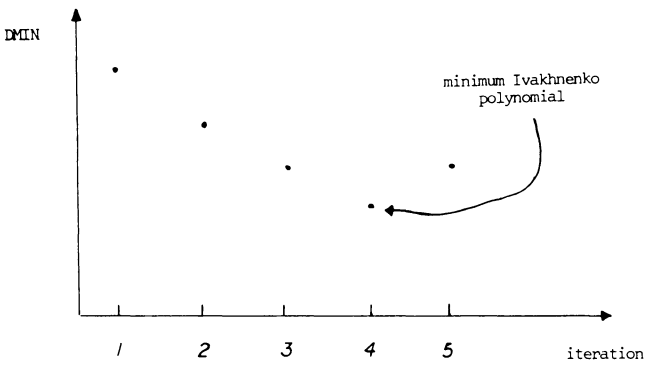


Figure 4. Stopping Criterion

4. NOTES ON THE GMDH ALGORITHM

1. The rationale for subdividing the n data points into the training and checking sets can be easily understood from the following analogy. Suppose one wishes to find a least squares polynomial $y(x) + a_0 + a_1x + a_2x^2 + \cdots + a_mx^m$ of some degree m that fits a set of data points $\{(x_i, y_i): i = 1, 2, \ldots, n\}$ (see Figure 5). The point here is that if the degree m of the polynomial is sufficiently large (namely, $m = n - 1$), then the least squares polynomial will pass through every data point. However, that does not mean this polynomial will accurately describe or model the true relationship between the variables x and y since if we were to compute the sum of squares of errors from our given polynomial and a new set of observations $\{(x_i, y_i): i = 1, 2, \ldots, n\}$ (checking set), we could get a very large sum of squares (DMIN). In other words, the model is too complicated. What we want is a polynomial constructed from the original data (training set), but checked against an independent set of observations (checking set). This is called the polynomial of *optimal complexity*.

2. After each iteration, the Ivakhnenko (evaluated) polynomial is stored in the first column of Z and the degree of the polynomial doubles at each iteration. After the first iteration, the degree is two, after the second it is four, then eight, 16, and so on.

3. The Ivakhnenko process was compared to the self-organizing process of natural selection (Scott and Hutchinson 1976). As these authors described, suppose a horticulturalist wishes to selectively breed a type of tulip in order to obtain a hybrid variety. He or she begins by growing a given number of tulips (original data X), after which all combinations are cross-fertilized. The resulting seeds are retrieved and planted, and the new generation of tulips is examined for the desirable trait. He or she then discards those that do not constitute an improvement. This experiment up to now corresponds to one iteration of the GMDH algorithm. The horticulturalist now cross-fertilizes again and again until a single tulip is grown that meets his or her criterion.

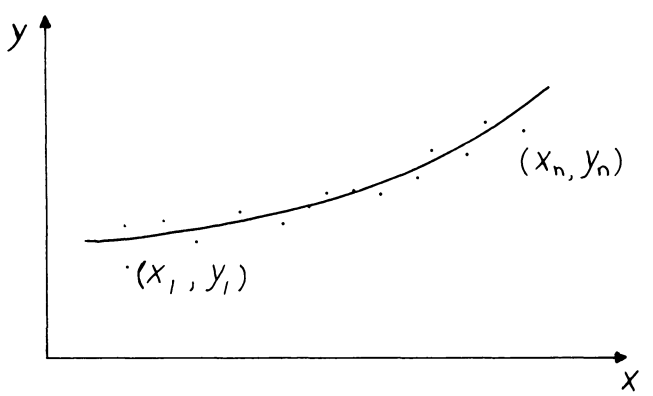


Figure 5. Least Squares Polynomial Fit

4. The Ivakhnenko polynomial

$$y = a + \sum_i b_i x_i + \sum_i \sum_j c_{ij} x_i x_j + \cdots$$

can be used in the following types of analyses.

Estimation. If one were given a set of numbers $x_1, x_2, \ldots, x_m$, one could substitute these into the polynomial to estimate the output to the system. After some modifications, the GMDH algorithm can be used to predict future values in a dynamic system where the previous n input-output relations are given. One can see how these modifications were carried out in a paper by Duffy and Franklin (1975).

Identification. The coefficients a, b_i, c_{ij}, d_{ijk}, ... measure the importance of the input variables $x_1, x_2, \ldots, x_m$ and their nonlinear interactions. A large coefficient c_{13} indicates an interaction of x_1 and x_3, while a large d_{114} points out an interaction of $x_1^2x_4$.

Control. Some of the variables $x_1, x_2, \ldots, x_m$ might be subject to our control and hence we can determine how to operate the system in an optimal manner. See Ivakhnenko and Stepashko (1977).

5. Note that the GMDH algorithm is what one might call a heuristic procedure; that is, it is not based on a solid theoretical foundation (as is regression analysis). For example, one cannot prove reliability estimates for the coefficients of the Ivakhnenko polynomial, nor can one prove how to pick the number of observations in the training and checking sets. One of the purposes of this paper is to stimulate statistical thinking in relationship to this algorithm.

6. Keep in mind that the GMDH algorithm provides a nice input-output relationship between the input and output variables, but will not necessarily give the correct structural model. For example, one could construct (although not likely in practice) a hypothetical set of observations that could be described accurately by as few as three of the independent variables, but none of these three variables, when paired, would produce a small d_j. For this reason we would then discard them and miss the model.

5. ADVANTAGE OF THE GMDH OVER HIGH-ORDER REGRESSION

To illustrate the advantage of the GMDH algorithm over standard nonlinear regression, suppose one wishes

to find a regression polynomial in the $m = 4$ variables x_1, x_2, x_3, x_4 of degree $p = 4$, that is,

$$y = r_0 + r_1x_1 + r_2x_2 + r_3x_3 + r_4x_4 + r_5x_1^2 + r_6x_2^2 + r_7x_3^2 + r_8x_4^2 + r_9x_1x_2 + r_{10}x_1x_3 + r_{11}x_1x_4 + r_{12}x_2x_3 + r_{13}x_2x_4 + r_{14}x_3x_4 + r_{15}x_1^3 + \cdots + r_{69}x_1x_2x_3x_4. \quad (5.1)$$

This polynomial has one constant term, four first-order terms, 10 second-order terms, 20 third-order terms, 35 fourth-order terms for a total of 70 coefficients. To find these 70 regression constants $r_0, r_1, \ldots, r_{69}$ by the usual regression technique, one must solve a system of 70 linear equations with 70 unknowns. Although this is not extremely hard by today's computing standards, one should keep in mind that the equations are ill conditioned and that one is working with a very small example.

On the other hand, the Ivakhnenko polynomial can be obtained from the GMDH algorithm by solving far fewer linear systems of order six.

The real advantage of the GMDH algorithm comes about when one models larger systems. In general, if the polynomial contains m variables, then the pth-order polynomial would contain a total of $(m + 1)(m + 2) \ldots (m + P)/(m!)$ terms. For example, even in the moderate sized problem where $m = 10$, the number of terms in the general polynomial of degree $p = 8$ would be 43,758. To find these coefficients by solving the normal equations is out of the question. On the other hand, using the GMDH algorithm, one can generally get by with solving only a few hundred normal systems of order six. By today's standards, one is talking about only a few minutes.

Of course, if one kept all the quadratic polynomials at each of the three iterations in the Ivakhnenko algorithm, one would have to solve

$$[\binom{10}{2} = 45] + [\binom{45}{2} = 990] + [\binom{990}{2} = 489{,}555] = 490{,}590$$

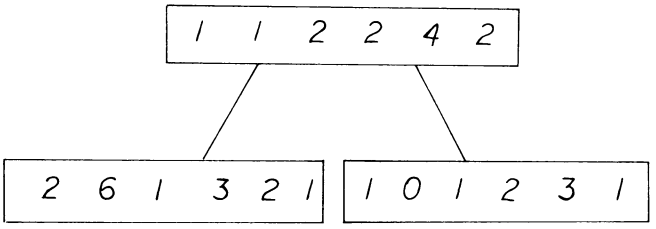
normal systems of order six, but in practice the actual number is much less. It is clear that when one has many variables and the degree of the polynomial is large, the GMDH algorithm has a distinct computational advantage.

The reason the GMDH algorithm requires fewer computations lies in the fact that it throws out harmful information (variables that do not correlate highly with the checking set). This makes the computations feasible and also helps make the systems of normal equations well conditioned.

6. COMPUTER PROGRAMS FOR GMDH

The author currently knows of no computer program for carrying out the GMDH algorithm that is available free of charge to the public. The author is preparing a series of programs for publication at the present time. The computations of the GMDH algorithm might be carried out by the following three subroutines.

Subroutine GMDH (X, Y, N, M, NT). Subroutine GMDH uses the data arrays X and Y of Figure 1 along with the numbers n, m, nt to compute a *tree* of quadratic polynomials of the form shown in Figure 6. For example, if the tree were computed to be



then the Ivakhnenko polynomial would be of degree four and given by

$$y = 1 + u + 2v + 2u^2 + 4v^2 + 2uv$$

where

$$u = 2 + 6x_1 + x_2 + 3x_1^2 + 2x_2^2 + x_1x_2$$

$$v = 1 + x_4 + 2x_3^2 + 3x_4^2 + x_3x_4.$$

Which of the four variables x_1, x_2, x_3, and x_4 of the m variables $x_1, x_2, \ldots, x_m$ should be included in the polynomial is something the programmer must determine from the computations. For every level of the tree, the Ivakhnenko polynomial increases by a factor of two. It is convenient to keep the Ivakhnenko polynomial stored as this tree of polynomials rather than in the conventional manner of (2.1).

Subroutine COMP. This subroutine uses the tree of polynomials generated in GMDH to evaluate the Ivakhnenko polynomial (2.1) for some value of the variables $x_1, x_2, \ldots, x_m$.

Subroutine COEF. This subroutine uses the same tree of polynomials computed by GMDH to compute the coefficients $a, b_i, c_{ij}, \ldots$ in the Ivakhnenko polynomial (2.1).

These three programs might be called by a main program with the restriction that GMDH must be called before the other two. In many cases a preprocessor subroutine might be called before calling any of these

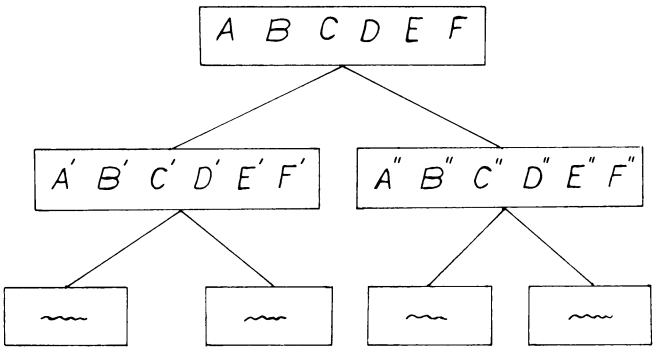


Figure 6. Tree of Polynomials Describing the Ivakhnenko Polynomial

subroutines. This subroutine would be for the purpose of detrending data, orthogonalizing data, and so forth. Any reader wishing to be informed of the publication date and name of the journal, if and when these programs are published, should send a self-addressed envelope to the author.

7. APPLICATIONS

This section briefly summarizes some applications of the GMDH algorithm published within the last 10 years. In many of these examples modifications of the basic GMDH algorithm were used.

1. Ivakhnenko and Vysotshiy (1975) constructed a model of the dynamics of a plankton ecological system, using only raw data and no internal assumptions.
2. Ivakhnenko and Ovchinnidov (1975) constructed a model of a hydroelectric power station, using the GMDH algorithm for the purpose of optimally controlling the system to take into consideration both power and fishery needs.
3. Parks, Ivakhnenko, and Boychuk (1975) used a GMDH-type algorithm to find a set of linear differential equations that modeled the British economy.
4. Ivakhnenko and Mikrykov (1975) used a modified GMDH algorithm to construct a prediction model of a blast furnace.
5. Ivakhnenko and Stepashko (1975) used a self-organized-type algorithm to produce the long-range prediction of river runoff.
6. Tamura and Aotani (1976) constructed a model for static large spatial pattern of air pollution concentration, using the GMDH algorithm to identify the nonlinear portion of the model.
7. Ihara (1975) used an improved GMDH algorithm to identify world population models.
8. Dubrovin and Stepashko (1974) extended the basic GMDH algorithm to adaptive identification problems. They used these techniques to find parameters in a controlled plant.
9. Stepanov (1974) used the GMDH algorithm to predict consumer goods.
10. Ivakhnenko and Todua (1973) showed how GMDH type self-organizing algorithms can be used for long-range prediction. The prediction of the flow rate of the Dnieper river and the British economy is given.
11. Duffy and Franklin (1975) used a slight modification of the GMDH algorithm to identify the sources of high nitrate levels in an agricultural area.
12. Ivakhnenko et al. (1971) applied GMDH algorithms to the problem of predicting the quantity of bacteria in the Ryninsk reservoir.
13. Pak and Kiryakov (1977) used the GMDH algorithm to reorganize the black and white letters that are invariant under shifts and rotations. The algorithm has been programmed in Algol 60.
14. Kozubovskiy (1977) reviewed the applications for the GMDH algorithm in microbiology.
15. Ivakhnenko and Vostrov (1977) identified structure and parameters of models of oil deposits, using mineral seismic data. The authors used first and second tabulated differences along with the GMDH algorithm to find the optimal model structure.
16. Silis and Rozenblitt (1976) built a complex decision function in the form of a disjunctive normal form by using the GMDH algorithm. The obtained decision function was evaluated by the number of correct answers in the training set and the statistical significance of the individual conjunctions.
17. Ichikawa and Vansteenkiste (1978) described the day-to-day variation in runoff after intensive rainfalls. They used GMDH algorithms.
18. Ivakhnenko and Stepashko (1977) constructed self-organizing dynamic models, using GMDH algorithms. These models predicted the growth of the spring wheat in terms of several climatic factors. The models can be used as control models since several inputs (such as soil moisture) can be controlled.

The serious reader can obtain a listing of 250 titles (plus abstracts) covering the GMDH algorithm from *The Information Service for Physicists, Electrotechnology, and Control* (INSPEC) index. For 10 dollars the reader can make a computer search using key words GMDH and IVAKHNENKO from the Bibliographic Retrieval Services, Inc., Corporation Park, Building 702, Scotia, NY 12302. There are other retrieval services that index the INSPEC journals and many libraries are connected on-line for the reader to use. INSPEC is an index that contains journals in computer science, cybernetics, mathematics, and modeling.

[*Received May 1980. Revised February 1981.*]

REFERENCES

DUBROVIN, O.F., and STEPASHKO, V.S. (1974), "An Adaptive GMDH Filter in a Closed Loop," *Soviet Automatic Control*, 7, 11–16.

DUFFY, J., and FRANKLIN, M. (1975), "A Learning Identification Algorithm and Its Application to an Environmental System," *IEEE Transactions on Systems, Man, Cybernetics*, 5, 226–239.

ICHIKAWA, A., and VANSTEENKISTE, G.C. (1978), "Prediction and Simulation of River Water Quality by Using GMDH," Ghent, Belgium: Proceedings of International Federation of Information Processing Societies.

IHARA, J. (1975), "Improved GMDH—A Case of Dynamical World Population Models," *System and Control*, 19, 201–210.

IVAKHNENKO, A.G. (1966), "Group Method of Data Handling—A Rival of the Method of Stochastic Approximation," *Soviet Automatic Control*, 13, 43–71.

——— (1971), "Polynomial Theory of Complex Systems," *IEEE Transactions on Systems, Man, Cybernetics*, 1, 364–378.

——— (1978), "The Group Method of Data Handling in Long-Range Forecasting," *Technological Forecasting and Social Change*, 12, 213–227.

IVAKHNENKO, A.G., KOPPA, YU., TODUA, M., and PETRACHE, G. (1971), "Mathematical Simulation of Complex Ecological Systems," *Soviet Automatic Control*, 4, 15–26.

IVAKHNENKO, A.G., and MIKRYUKOV, D.G. (1975), "Ob-

jective Identification of Thermal State of Blast Furnace by Self-Organization Methods," *Soviet Automatic Control*, 8, 44–49.

IVAKHNENKO, A.G., and OVCHINNIDOV, V.A. (1975), "Control of Dniepr Power Station Water Reservoirs With Two Optimally Criteria Based on Self-Organization," *Soviet Automatic Control*, 8, 49–59.

IVAKHNENKO, A.G., PEKA, P. YU, and KOSHULKO, A.I. (1976), "Simulation of the Dynamics of the Mineralization Field of Aquifers With Optimization of Porosity Estimate of the Medium," *Soviet Automatic Control*, 9, 35–44.

IVAKHNENKO, A.G., and STEPASHKO, V.S. (1975), "Self-Organization of Models and Long-Term Prediction of River Runoff by the Balance Criterion," *Soviet Automatic Control*, 8, 34–41.

——— (1977), "Self-Organization of Dynamic Models of Growth of Agricultural Crops for Control of Irrigated Crop Rotation," *Soviet Automatic Control*, 10, 32–44.

IVAKHNENKO, A.G., and TODUA, M.M. (1973), "Statistical Prediction of Random Processes Using Self-Organization of the Prediction Equations," *Soviet Automatic Control*, 5, 15–36.

IVAKHNENKO, A.G., and VOSTROV, N.N. (1977), "Identification of Oil Horizons by Computer-Aided Self-Organization of Mathematical Models," *Soviet Automatic Control*, 10, 3–12.

IVAKHNENKO, A.G., and VYSOTSHIY, V.N. (1975), "Simulation of the Dynamics of the Environment-Plankton Ecological Systems of the White Sea and Analysis of Its Stability," *Soviet Automatic Control*, 8, 9–18.

KOZUBOVSKIY, S.F. (1977), "Use of the GMDH in Microbiology," *Soviet Automatic Control*, 10, 82–85.

PAK, V.G., and KIRYAKOV, Yu (1977), "GMDH Algorithm for Recognition of Black-and-White Letters or Numbers," *Soviet Automatic Control*, 10, 8–12.

PARKS, P., IVAKHNENKO, A.G., and BOYCHUK, L.M. (1975), "A Self-Organizing Model of the British Economy for Control With Optimal Prediction Using the Balance-of-Variables Criterion," *International Journal of Computer and Information Science*, 4, 349–379.

ROSENBLATT, F. (1958), "The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain," *Psychological Review*, 68, 386–408.

SCOTT, D.E., and HUTCHINSON, C.E. (1976), "The GMDH Algorithm—A Technique for Economic Modeling," Report No. ECE-SY-67-1, University of Massachusetts, Dept. of Computer Science.

SILIS, YA, and ROZENBLITT, A.B. (1976), "Algorithms for Construction of Decision Function in the Form of a Complex Logic Proposition," *Soviet Automatic Control*, 9, 1–5.

STEPANOV, V.A. (1974), "Results of Mass Checking of GMDH Efficiency in Long-Term Prediction of Demand for Consumer Goods," *Soviet Automatic Control*, 7, 60–66.

TAMURA, H., and AOTANI, T. (1976), "Large-Spatial Air Pollution Identification by Combined Approach of Source-Receptor Matrix and GMDH," *Transactions of the Society of Instrumentation and Control in Engineering*, 12, 121–126.