Intelligent Paralleling of Recurrent Computing in COMBI GMDH Algorithm for Effective Solving Combinatorial Modelling Problems

Serhiy Yefimenko

International Research and Training Centre of Information Technologies and Systems of NAS and MES of Ukraine, Akademik Glushkov prospect 40, 03680, Kyiv, UKRAINE

syefim@ukr.net

Abstract. *The idea of combined use of recurrent and parallel computing in combinatorial GMDH algorithm is investigated. Intelligent technology of inductive modeling on the basis of recurrent-and-parallel computing is developed. Its effectiveness is experimentally examined using a multiprocessor cluster system.*

Keywords

Structural and parametric identification, GMDH, combinatorial algorithm, recurrent and parallel computing, cluster system.

1 Introduction

Group Method of Data Handling (GMDH) [1] is one of the most effective inductive modeling methods. It is widely used for solving broad spectrum problems of artificial intelligence such as identification, forecasting, recognition, clustering, and macromodeling.

For software tools, based on inductive modeling methods, the run-time is one of the most important criteria of their efficiency. The most effective ways to achieve the high performance of such products are recurrent parameter estimation and parallel computing. High-performance software tools on the basis of both recurrent and parallel computing were already developed and proved their efficiency [2-4].

The purpose of this work is to develop an intelligent technology combining both these approaches and allowing to effectively solve (fully or partially) the problem of exhaustive search using GMDH combinatorial algorithm when maximally loading all available computational resources.

2 The Problem of Structural and Parametric Identification

The problem of structural and parametric identification may be considered as follows.

It is necessary to form (based on experimental data set) some discrete set of models (candidates) $\Im$ and to find the optimal model with respect to the value of a given selection criterion *CR*:

$$f^{*} = \arg\min_{f \in \Im} CR(y, f(X, \hat{\theta}_f)). \qquad (1)$$

This corresponds to structure identification problem.

Here the vector of parameters $\hat{\theta}_f$ for every model $f \in \Im$ is estimated as the solution of the task

$$\hat{\theta}_f = \arg\min_{f \in \Im} Q(y, X, \theta_f, s_f), \qquad (2)$$

where $Q \neq CR$ is a quality criterion for the solution of parametric identification problem of every generated model, and s_f is a complexity of model f (i.e. number of its non-zero parameters).

3 Combinatorial GMDH Algorithm with Recurrent Parameters Estimation

The combinatorial GMDH algorithm COMBI [1] is used for solving the problem (1, 2) by exhaustive search of all possible variants and finding the best regressive model containing the most informative subset of input arguments (regressors).

For linear object with *m* input, all possible models are compared in the process of exhaustive search. Total quantity of all generated models of the type

$$\hat{y}_v = X_v \hat{\theta}_v,\ v = 1,...,2^m - 1 \tag{3}$$

is 2^m–1. Decimal number v corresponds to binary number d_v in (3). Unit elements of d_v indicate inclusion regressors with corresponding numbers in the model, whereas zero elements signify exclusion.

Due to the exponential growth of 2^m as function of arguments amount, it is advisable to use algorithms recurrent in the number of parameters in structural identification problems for the parameters estimation of model structures being sequentially complicated.

Efficient recurrent modifications of classic Gauss and Gramm-Schmidt algorithms were offered in [3]. As far as the recurrent variant of Gauss method is useful for combinatorial algorithm paralleling, its short-form description is done below.

The modification, in a nutshell, is as follows. The matrix $H_s = X_s^T X_s$ of the size $s \times s$ is reduced to superdiagonal form by computing only elements $h_{i\,s}^s,\quad i = \overline{2,\, s-1},\ h_{s\,i}^s,\quad i = \overline{2,\, s}$, and $g_s = X_s^T y$ at every step $s,\ s = \overline{1, m}$ during the direct motion. The elements of the nested matrix H_{s-1} of size (s–1)×(s–1) (reduced to superdiagonal form on the previous step) remain changeless. So only “bordering elements” (bold fonts) are computed on step s:

$$\left[\begin{array}{cccccccc|c}
h_{11} & h_{12} & h_{13} & \dots & h_{1s\text{-}1} & \boldsymbol{h_{1s}} & \dots & h_{1m} & g_1 \\
h_{21} & h_{22} & h_{23} & \dots & h_{2s\text{-}1} & \boldsymbol{h_{2s}} & \dots & h_{2m} & g_2 \\
h_{31} & h_{32} & h_{33} & \dots & h_{3s\text{-}1} & \boldsymbol{h_{3s}} & \dots & h_{3m} & g_3 \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
h_{s\text{-}1,1} & h_{s\text{-}1,2} & h_{s\text{-}1,3} & \dots & h_{s\text{-}1,s\text{-}1} & \boldsymbol{h_{s\text{-}1,s}} & \dots & h_{s\text{-}1,m} & g_{s\text{-}1} \\
\boldsymbol{h_{s1}} & \boldsymbol{h_{s2}} & \boldsymbol{h_{s3}} & \boldsymbol{\dots} & \boldsymbol{h_{s,s\text{-}1}} & \boldsymbol{h_{ss}} & \dots & h_{sm} & \boldsymbol{g_s} \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\
h_{m1} & h_{m2} & h_{m3} & \dots & h_{m,s\text{-}1} & h_{ms} & \dots & h_{mm} & g_m
\end{array}\right]$$

4 Paralleling of COMBI Algorithm with Standard Binary Counter

The scheme of COMBI paralleling based on the modified recurrent Gauss algorithm with standard binary counter is described in [6]. The sequence of all possible combinations for models comprising e.g. m=3 arguments will be as follows (with corresponding binary structural vector):

$y_1 = a_1x_1$ $\quad\{1, 0, 0\}$

$y_2 = a_2x_2$ $\quad\{0, 1, 0\}$

$y_3 = a_1x_1 + a_2x_2$ $\quad\{1, 1, 0\}$

$y_4 = a_3x_3$ $\quad\{0, 0, 1\}$

$y_5 = a_1x_1 + a_3x_3$ $\quad\{1, 0, 1\}$

$y_6 = a_2x_2 + a_3x_3$ $\quad\{0, 1, 1\}$

$y_7 = a_1x_1 + a_2x_2 + a_3x_3$ $\quad\{1, 1, 1\}$

Tab. 1 represents approximate dependence of modeling time on arguments number and used processors for constructed algorithm. Already for more than 50 arguments, an exhaustive search (in acceptable modeling time) becomes impossible even for cluster system containing one hundred processors. Any effective reducing of exhaustive search is impossible due to the feature of the standard binary generator: complexity of structural vectors changes inconsequentially.

Tab. 1. Approximate time of exhaustive search.

Arguments	Models	Time	
		1 processor	**100 processors**
20	1 048 575	1 s	0,01 s
21	2 097 151	2 s	0,02 s
…	…	…	…
40	1,1E+12	~ 12 days	~ 3 hours
…	…	…	…
50	1,1E+15	~ 34 years	~ 124 days

For this case some another scheme of COMBI paralleling was constructed.

5 Paralleling of COMBI Algorithm with Sequential Binary Counter

This scheme uses such sequence of binary numbers generation when all combinations with one unit in structural vector appears first of all (totally $C_m^1 = m$ possible variants is generating), then with two units ($C_m^2 = \frac{m(m-1)}{2}$ possible variants), and so on to complete model ($C_m^m = 1$) comprising all arguments.

The sequence of all possible combinations for models comprising three arguments will be the following:

$y_1 = a_1x_1$ $\{1, 0, 0\}$

$y_2 = a_2x_2$ $\{0, 1, 0\}$

$y_3 = a_3x_3$ $\{0, 0, 1\}$

$y_4 = a_1x_1 + a_2x_2$ $\{1, 1, 0\}$

$y_5 = a_1x_1 + a_3x_3$ $\{1, 0, 1\}$

$y_6 = a_2x_2 + a_3x_3$ $\{0, 1, 1\}$

$y_7 = a_1x_1 + a_2x_2 + a_3x_3$ $\{1, 1, 1\}$

The scheme can be easily used for COMBI paralleling on the given amount of processors [7].

It allows to partially solve the problem of exhaustive search when arguments number exceeds capability of the algorithm with a standard binary generator. In this case it is advisable to execute an exhaustive search not among all possible models but only for models of the restricted complexity.

6 Algorithm COMBI Based on Recurrent-and-Parallel Computing

We will estimate modeling time of COMBI algorithm with successive complication of structures on the basis of the recurrent and parallel computing (see Tab. 2). Let acceptable modeling time (for paralleling on 100 processors) will be no more than 10 hours. Under such constraint it is possible to build all models of complexity no more than l=15 for the total amount of arguments m=50 (i.e. to build all models for which 50-elements binary structural vectors contain from 1 to 15 units). For 100 arguments it is possible to reach complexity 9, for 150 and 200 arguments – complexity 7.

Tab. 2. Approximate time of restricted search.

Arguments, *m*	**Complexity, *l***	**Models,** $\sum_{i=1}^{l} C_m^i$	**Time, hours**	
			1 processor	**100 processors**
50	15	3,7E+12	984	~ 10
100	9	2,1E+12	558	~ 6
150	7	3,1E+11	82	~ 1
200	7	2,4E+12	628	~ 6

7 Intelligent COMBI Algorithm

Thus, taking into account mentioned features for the two methods of COMBI paralleling, it is possible to propose a technology of intelligent model building using combinatorial GMDH algorithm on the basis of recurrent-and-parallel computing. Fig. 1 presents the respective scheme with automatic tacking into account number of arguments, number of available processors and running time restriction.

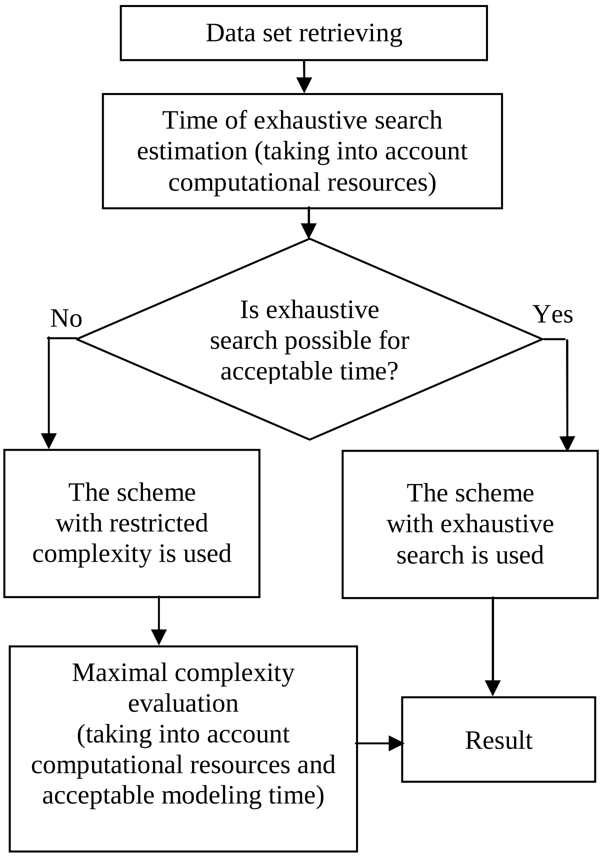

Fig. 1. Scheme of COMBI intelligent paralleling.

8 Results of Experiments

We will show, how proposed technology works by giving an example. The computer cluster system SCIT-4 [8] will be used for solving of two combinatorial modeling problems. Let acceptable modeling time will be no more than 1 minute.

Experiment 1. The problem of structural and parametric identification on exhaustive search for 30 arguments was solved. A test task was formed as follows: the design matrix *X* of size 50×30 (50 records for 30 arguments) was generated for the system of conditional equations $X\theta = y$. Vector y was formed as a linear combination of only five arguments:

$$y = x_{11} + x_{12} + x_{13} + x_{14} + x_{15}\ . \tag{4}$$

The results of experiment as run-time of COMBI algorithm with standard binary counter, depending on amount of used processors, are given in Tab. 3. Thus under modeling time constraint it is quite enough 16 processors to obtain model (4).

Tab. 3. Run-time of exhaustive search.

Processors	1	2	4	8	10	16	20
Run-time, s	855	429	214	108	87	54	41

Experiment 2. The design matrix X of size 70×50 (70 records for 50 arguments) was generated. Vector y was formed as a linear combination:

$$y = x_{10} + x_{20} + x_{30} + x_{40} + x_{50}\ . \tag{5}$$

Time of exhaustive search by COMBI algorithm with standard binary generator on the basis of recurrent computing for 50 arguments may last about 34 years (see Tab. 1). However this problem can be solved by algorithm with successive complication of structures restricted to models of complexity 7 (i.e. considering all models containing no more than 7 arguments).

For structural parametric identification 6 nodes containing 24 processor cores were used. Model (5) by such computer system was obtained for less than 2 seconds.

9 Conclusion

The paper describes 2 methods of computing paralleling in combinatorial GMDH algorithm and analyses their advantages and limitations.

Effective intelligent technology of inductive modeling is developed on the basis of recurrent-and-parallel computing. It allows fully or partially solving the problem of exhaustive search, depending on available computational resources and acceptable modeling time.

Test experiment on a cluster multiprocessor system confirmed high efficiency of the technology.

References

[1] Madala H.R., Ivakhnenko A.G. Inductive Learning Algorithms for Complex Systems Modeling. – New York: CRC Press, 1994. – 368 p.

[2] Stepashko, V. S., A Combinatorial Algorithm of the Group Method of Data Handling with Optimal Model Scanning Scheme // *Soviet Automatic Control*, 14, No. 3, (1981). – P. 24-28.

[3] Stepashko V. S. , and Efimenko S. N. Sequential Estimation of the Parameters of Regression Model // *Cybernetics and Systems Analysis, Springer New York*, July, 2005, Vol. 41, Num. 4. – P. 631-634.

[4] Stepashko V., Yefimenko S. Parallel algorithms for solving combinatorial macromodelling problems // *Przegląd Elektrotechniczny* (*Electrical Review*), ISSN 0033-2097, R. 85. – NR 4. – 2009. – P. 98-99.

[5] Yefimenko S. Comparative Effectiveness of Parallel and Recurrent Calculations in Combinatorial Algorithms of Inductive Modelling // *Proceedings of the 4th International Conference on Inductive Modelling ICIM*'2013. – Kyiv, 2013. – P. 231-234.

[6] Єфіменко С.М., Степашко В.С. Рекурентно-паралельні обчислення в комбінаторному алгоритмі МГУА для задач індуктивного моделювання // *Матеріали 21-ї Міжнародної конференції з автоматичного управління „Автоматика-2014”, Київ, 23-27 вересня 2014 р.* – К.: Вид-во НТУУ „КПІ” ВПІ ВПК „Політехніка”, 2014. – С. 200-201.

[7] Єфіменко С.М. Комбінаторний алгоритм МГУА з послідовним ускладненням структур моделей на основі рекурентно-паралельних обчислень // *Індуктивне моделювання складних систем: Зб. наук. пр.* – К.: МННЦ ІТС НАН та МОН України, 2014. – Вип. 6. – С. 64-71.

[8] *Internet source* http://icybcluster.org.ua.