

УДК 681.5.015

В.С. Степашко, А.С. Булгакова

Обобщенный итерационный алгоритм метода группового учета аргументов

Представлена архитектура обобщенного итерационного алгоритма метода группового учета аргументов, частные случаи которого – алгоритмы многорядного и релаксационного типов, а также разновидности итерационно-комбинаторных алгоритмов. На основе предложенной методики численного анализа эффективности итерационных алгоритмов выполнено их сравнительное исследование, включая сходимость к истинной модели.

The architecture of a generalized iterative algorithm of the group method of data handling is presented, special cases of which are algorithms of multilayered and relaxation types and variants of iterative combinatorial algorithms. Based on the proposed technique for numerical analysis of iterative algorithms efficiency, a comparative study of them was carried out including their convergence to the true model.

Представлено архітектуру узагальненого ітераційного алгоритму методу групового урахування аргументів, окремими випадками якого є алгоритми багаторядного та релаксаційного типів, а також різновиди ітераційно-комбінаторних алгоритмів. На основі запропонованої методики чисельного аналізу ефективності ітераційних алгоритмів виконано їх порівняльне дослідження, включно зі збіжністю до істинної моделі.

Введение. В задачах построения моделей сложных систем в условиях неполноты информации активно используются методы и средства индуктивного моделирования, предназначенные, прежде всего, для функционального описания входо-выходных характеристик систем.

Научное направление под названием «индуктивное моделирование сложных процессов и систем», сформированное академиком Национальной академии наук Украины Алексеем Григорьевичем Ивахненко, имеет своим началом статью, опубликованную в 1968 г. в журнале «Автоматика» [1]. Это новое направление автор называл по-разному: вначале «эвристическая самоорганизация» [2], позже – «самоорганизация моделей по экспериментальным данным» [3], далее – «индуктивная самоорганизация моделей сложных систем» [4].

Наконец, в 1998 году был сформулирован термин «индуктивное моделирование», закрепленный в названии серии международных конференций и семинаров по индуктивному моделированию, проводимых, начиная с 2002 г., как в Украине, так и за границей [5].

Среди различных методов моделирования, которые можно отнести к индуктивным, выделяется

метод группового учета аргументов (МГУА), позволяющий строить модели непосредственно по выборке данных, без привлечения дополнительной априорной информации. МГУА успешно применяется в задачах анализа данных и выявления закономерностей, моделирования и прогнозирования, структурной идентификации и кластеризации, классификации и распознавания образов. Он дает возможность автоматически находить взаимозависимости и закономерности, неявно отраженные в данных, и представлять их в явном виде математических моделей оптимальной сложности.

По состоянию на сегодня разработано и исследовано много разновидностей алгоритмов МГУА переборного [6] и итерационного [7] типов. Переборные алгоритмы эффективны как средство структурной идентификации, но лишь для ограниченного числа аргументов, поскольку базируются на полном или направленном переборе всех возможных вариантов структур моделей. Итерационные алгоритмы работоспособны при достаточно большом количестве аргументов, но специфика их архитектуры не гарантирует построения модели истинной структуры, поскольку они базируются на неполных

индуктивных процедурах иерархического усложнения моделей. Долгое время эти два класса алгоритмов развивались независимо, без детального изучения возможности объединения их сильных сторон при одновременном устранении недостатков.

Цель данной статьи – представление обобщенного итерационного алгоритма МГУА, в основе которого лежит сочетание идей сохранения начального базиса моделирования и применения оптимизации сложности частных моделей. Его частными случаями есть алгоритмы многорядного и релаксационного типов, а также некоторые разновидности итерационно-комбинаторных (гибридных) алгоритмов.

Общая постановка задачи моделирования по данным наблюдений

Будем рассматривать класс задач индуктивного моделирования, которые, так или иначе, сводятся к выбору оптимальной по заданному критерию модели из множества генерируемых моделей, линейных по параметрам, для случая объекта с одним выходом.

Пусть имеется n наблюдений за поведением объекта или системы, т.е. задана выборка (матрица) данных $W = (Xy)$, содержащая информацию об изменении m входных переменных $X[n \times m]$ и одной выходной $y[n \times 1]$. Индуктивный процесс решения задачи определения структуры и параметров модели состоит в использовании данных одной части выборки для генерирования постепенно усложняемых гипотез о виде моделей неизвестной закономерности и селекции наиболее правдоподобных из них с применением принципа внешнего дополнения, выражающегося в виде ошибки моделей на данных другой части.

В общем случае задача построения модели сводится к формированию по выборке экспериментальных данных некоторого множества Φ моделей различной структуры [8]:

$$\hat{y}_f = f(X, \hat{\theta}_f), f \in \Phi, \quad (1)$$

и поиску оптимальной модели из этого множества как решения задачи *дискретной* оптимизации по условию минимума заданного внешнего критерия селекции $CR(\cdot)$:

$$f^* = \arg \min_{f \in \Phi} CR(y, f(X, \hat{\theta}_f)), \quad (2)$$

где оценка параметров $\hat{\theta}_f$ каждой $f \in \Phi$ есть решением задачи *непрерывной* оптимизации:

$$\hat{\theta}_f = \arg \min_{\theta \in R^{s_f}} QR(y, X, \theta_f), \quad (3)$$

где $QR(\cdot) \neq CR(\cdot)$ – критерий качества решения задачи параметрической идентификации модели в процессе структурной идентификации, s_f – сложность модели f (число ее параметров).

Постановка задачи (1) – (3) соответствует задачам моделирования с применением переборных алгоритмов МГУА, когда конечное множество Φ можно полностью описать и организовать тем или иным способом поиск модели оптимальной сложности [6]. Специфика итерационных алгоритмов состоит в том, что для такого множества – вообще говоря, бесконечного – можно лишь указать правило построения, однако его нельзя просмотреть, поскольку его формирование происходит в процессе решения задачи [7].

Архитектура классического итерационного алгоритма МИА МГУА

В наиболее известном классическом многорядном алгоритме МГУА [1] задача построения оптимальной модели (1) – (3) решается индуктивным путем: строятся модели постепенно усложняемой структуры, причем процесс усложнения имеет характер итераций, когда лучшие предыдущие результаты используются на следующем ряде (итерации). Усложнение происходит по единому правилу, что позволяет строить сколь угодно сложную модель от большого числа переменных (аргументов), характеризующих объект моделирования.

Согласно [7], определение произвольного итерационного алгоритма МГУА, как и вообще любой итерационной процедуры, предполагает указание: 1) матрицы начального приближения; 2) оператора перехода к следующей итерации; 3) правила остановки. С учетом этого определим базовый многорядный алгоритм МГУА [1], называемый в современной терминологии *многорядным итерационным алгоритмом* МИА МГУА.

1. Матрицей начального приближения алгоритма МИА есть исходная матрица из m столбцов измерений входных переменных

$$X = (x_1, \dots, x_m), x_j [n \times 1], j = \overline{1, m}.$$

2. Общим видом оператора перехода к следующей итерации есть некоторая функция $z = f(u, v)$, где u, v – произвольная пара векторов решений предыдущей итерации, z – вектор решения (выход модели) следующей итерации. Переходная функция $z = f(u, v)$, называемая *частным описанием*, может быть линейной, билинейной или квадратичной:

$$\begin{aligned} z &= f(u, v) = a_0 + a_1 u + a_2 v; \\ z &= f(u, v) = a_0 + a_1 u + a_2 v + a_3 uv; \\ z &= f(u, v) = a_0 + a_1 u + a_2 v + \\ &+ a_3 uv + a_4 u^2 + a_5 v^2. \end{aligned} \quad (4)$$

Стандартным частным описанием базового алгоритма МГУА [1] была квадратичная функция, т.е. он предназначался для построения моделей сложных нелинейных объектов.

3. Алгоритм останавливается при выполнении условия $CR_{r+1} \geq CR_r$, где CR – критерий качества модели, r – номер итерации (ряда). Обычно применяется критерий регулярности:

$$AR_{B|A} = \|y_B - \hat{y}_B\|^2 = \|y_B - X_B \hat{\theta}_A\|^2, \quad (5)$$

соответствующий разбиению выборки данных $W = (Xy)$ на две непересекающихся подвыборки – обучающую A и проверочную B , $W^T = (A^T B^T)$, где \hat{y}_B – оценка выхода на B по модели, вектор параметров которой $\hat{\theta}_A$ вычислен на A .

Приведенные основные характеристики МИА не описывают полностью работу алгоритма, поэтому рассмотрим подробнее операции, выполняемые на первом и произвольном $(r+1)$ -м рядах.

Первая итерация (первый ряд). Из входных векторов–аргументов x_1, x_2, \dots, x_m выбираются все возможные пары $x_i, x_j, i \neq j$, составляются частные описания вида (4), т.е. $y_l^1 = f(x_i, x_j), l = 1, 2, \dots, C_m^2$, и по МНК на обучающей выборке находятся оценки неизвест-

ных параметров (коэффициентов) $\hat{a}_0, \hat{a}_1, \hat{a}_2, \dots$. По заданному критерию на проверочной выборке отбираются F лучших моделей $\hat{y}_k^1, k = \overline{1, F}$, т.е. выполняется селекция, где F называется *свободой выбора*. Выходы этих F моделей служат аргументами–входами для формирования моделей следующего ряда. Далее находится минимальное CR_{\min}^1 среди всех F значений критерия на первом ряду.

Произвольная $(r+1)$ -я итерация (ряд $r+1$).

Из векторов–аргументов $\hat{y}_k^r, k = \overline{1, F}$, предыдущего r -го ряда формируются все возможные частные описания вида (4), т.е.

$$y_l^{r+1} = f(y_i^r, y_j^r), l = 1, 2, \dots, C_F^2, i, j = \overline{1, F}, \quad (6)$$

и по МНК на A находятся оценки параметров. Затем по критерию (5) отбираются F лучших моделей $\hat{y}_k^{r+1}, k = \overline{1, F}$, и находится CR_{\min}^{r+1} . Проверяется условие $CR_{\min}^{r+1} \geq CR_{\min}^r$, при выполнении которого итерационный процесс останавливается, иначе – переход к следующему ряду. В случае остановки в качестве оптимальной принимается модель, соответствующая значению CR_{\min}^r на предыдущем r -м ряду. Структура этого алгоритма представлена на рис. 1.

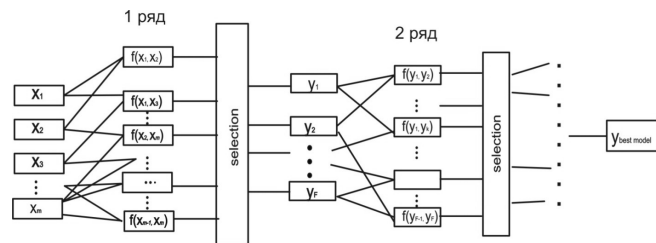


Рис. 1. Структура многорядного итерационного алгоритма МИА МГУА

Отметим, что такая многорядная процедура последовательной генерации структур моделей имитирует процесс биологической селекции с попарным скрещиванием родительских индивидуумов, т.е. этот алгоритм относится к группе эволюционных «инспирированных природой» (*nature inspired*) методов вычислительно-интеллекта [9].

Высокую эффективность итерационных алгоритмов демонстрируют многочисленные при-

меры решения практических задач моделирования для целей прогноза и управления. Они работоспособны при весьма больших m – порядка 1000 – и позволяют строить линейные и нелинейные модели, причем даже в вырожденных задачах, когда длина выборки $n < m$ [7].

Этот классический алгоритм МИА МГУА все же не лишен существенных недостатков:

- возможность потери информативных аргументов, если они были исключены в начале перебора;
- возможность закрепления неинформативных аргументов, если они были включены в начале перебора;
- экспоненциальный рост степени полинома: 1, 2, 4, 8, ... при квадратичном описании;
- с ростом числа рядов векторы выходов лучших моделей становятся все более коррелированными, что ухудшает обусловленность систем уравнений для оценивания параметров.

Далее рассмотрим некоторые известные пути преодоления этих недостатков.

Развитие архитектур итерационных алгоритмов МГУА

С целью повышения эффективности применения МГУА в разные периоды развития метода были предложены различные модификации базового алгоритма. Одним из очевидных путей улучшения было увеличение свободы выбора F во избежание потери оптимальной модели [10]. Поскольку при таком подходе время поиска лучшей модели тоже растет, в [11] для уменьшения объема вычислений используется подход к выбору промежуточных переменных на основе разнообразия критериев: промежуточная переменная будет выбрана как оптимальная на ряде r , если ее эффективность по разным критериям улучшится в $(r + 1)$ -м ряде.

Чтобы исключить рост сложности модели, применялось варьирование частных описаний: в [12] используют многочлен второго порядка на первом ряде и линейный на следующих; в [13], наоборот, вводят вначале линейное, а на дальнейших рядах – квадратичное частное описание.

Многорядно-комбинаторный алгоритм МГУА. Для устранения третьего недостатка ба-

зового алгоритма – экспоненциального роста степени полинома – в [14] было предложено применение полного перебора вариантов частной модели – так называемый многорядно-комбинаторный алгоритм, подробно описанный в [15]. Частное описание здесь имеет вид $f(u, v) = a_1 + a_2u + a_3v + a_4uv$, и для каждой пары аргументов выполняется перебор различных вариантов сочетаний по 1, 2, 3 и 4 одночлена с выбором наилучшего по внешнему критерию. Однако в этом алгоритме, как и в классическом МИА, все же не устраняется возможность потери информативных аргументов. Подобный алгоритм был предложен также в [16].

Многорядные алгоритмы с сохранением базиса. Разработанный в [17] так называемый «упрощенный многорядный алгоритм МГУА» содержал оригинальную процедуру формирования двухчленного частого описания вида

$$y^{r+1} = f(y^r, x) = a_r y^r + b_r \varphi(x_1, \dots, x_m), \quad (7)$$

где y^r, y^{r+1} – лучшие модели соседних рядов, $\varphi(\cdot)$ – вообще говоря, нелинейная функция от заданного числа входных (первичных) аргументов, причем ее усложнение по некоторому правилу выполняется до тех пор, пока уменьшается критерий селекции. Введение первичных аргументов на каждом ряде этого алгоритма устраняет возможность потери информативных аргументов, однако его недостаток – отсутствие свободы выбора – здесь $F = 1$. Отметим, что пары тут образуются только из промежуточных и начальных аргументов – такого типа алгоритмы со временем стали называться *релаксационными* [7].

Следующий шаг по пути улучшения классического алгоритма был сделан в [18], где предложен *многорядный алгоритм с селекцией первичных аргументов*. В нем фактически комбинируются структуры алгоритмов [1] и [17]: на каждом ряде пары для формирования частного описания образуются не только из промежуточных аргументов, но и из промежуточных и начальных, т.е. в нем описания (6) дополняются также описаниями вида

$$y_l^{r+1} = f(y_i^r, x_j), l = 1, 2, \dots, Fm, i = \overline{1, F}, j = \overline{1, m}. \quad (8)$$

Очевидно, что здесь также исключается потеря информативных аргументов, однако не устраняется сохранение неинформативных аргументов и возрастание степени полинома.

Подобного класса комбинированные алгоритмы с включением начальных аргументов в перебор на всех рядах итерационной процедуры были существенно усовершенствованы в таких алгоритмах: *CML* – многорядный алгоритм МГУА, в котором входная матрица аргументов для ряда $r + 1$ имеет вид $X = (\mathbf{0}, \mathbf{1}, y_1^r, \dots, y_F^r, x_1, \dots, x_m)$, где $\mathbf{0}$ и $\mathbf{1}$ – нулевые и единичные $n \times 1$ векторы, а частные описания – линейные вида $y_i^{r+1} =$

$= a^r u_i^r + b^r u_j^r$, где u_i^r, u_j^r – произвольные векторы из матрицы X [19]; *GN* – полиномиальный алгоритм с геделевской нумерацией [20] (для автоматического сворачивания иерархической структуры в явную формулу), где на ряде r применяются частные описания не от двух, а от трех входов (векторов матрицы X):

$$z_l^{r+1} = a^r u_l^r + b^r u_j^r u_k^r, l = 1, 2, \dots, C_F^3, \quad i, j, k = \overline{1, F}. \quad (9)$$

Очевидно, что *CML* – частный случай алгоритма *GN* (при $u_k = 1$ или $u_j = 1$). Алгоритм типа *GN* получил дальнейшее развитие в [21] как *GMDH-PNN*.

Для итерационных алгоритмов важна их сходимость [22], однако она доказана только для так называемой «внутренней сходимости» [19], когда анализируется не внешний критерий (5), а «внутренний», т.е. ошибка модели на всей выборке без разбиения. Дальнейшие результаты по анализу внутренней сходимости получены в [23] для обобщенного релаксационного алгоритма МГУА [24], который представляет собой существенно усовершенствованный вариант [17].

Современные тенденции в развитии алгоритмов самоорганизации моделей

МГУА как полиномиальная нейросеть. В 1990-х годах, когда искусственные нейросети получили широкую известность, А.Г. Ивахненко и пользователи его метода начали называть типичную структуру алгоритма МГУА также нейросетью. Более того, в последние годы специалисты по нейросетям за границей чаще все-

го называют алгоритм МГУА *Polynomial Neural Network (PNN)*, т.е. Полиномиальная Нейронная Сеть (ПНС). При этом основной элемент итерационных алгоритмов – частное описание – есть элементарным нейроном ПНС МГУА, его структура для квадратичного частного описания изображена на рис. 2. Оригинальность и эффективность нейросети из таких нейронов заключается в скорости процесса локальной настройки весов нейронов и автоматической глобальной оптимизации (т.е. *самоорганизации*) структуры сети – числа узлов и количества рядов (скрытых слоев).

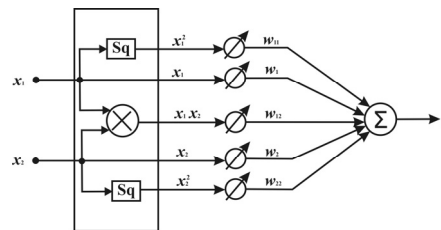


Рис. 2. Структура нейрона МГУА с квадратичным частным описанием

Нейросеть с активными нейронами. Нейроны ПНС МГУА (частные описания) можно назвать «пассивными» – все они имеют одинаковую структуру, т.е. полиномиальная нейросеть гомогенна. В 1990-х годах А.Г. Ивахненко предложил новый тип сети МГУА – с активными нейронами [25], каждый из которых есть переборным или другим алгоритмом МГУА, благодаря чему структура каждого нейрона оптимизируется. При этом все нейроны могут получить разную структуру, что повышает гибкость настройки гетерогенной сети на конкретную задачу. Такие сети называются также *дважды многорядными* [26].

Существуют и другие варианты МГУА-подобных нейросетей с активными нейронами.

МГУА-подобная нейросеть с обратной связью. Нейронная сеть с обратной связью, когда выходы нейронов данного слоя (ряда) «скрещиваются» с исходными переменными была предложена в [27]. Оптимальная архитектура сети и ее структурные параметры строятся на основе самоорганизации: автоматически выбирается одна из трех архитектур для каждого нейрона – сигмоидная, радиальная или полиномиальная

функция активации, а также количество слоев, нейронов в слоях и входные переменные.

Эволюция групп адаптивных моделей. Эволюционный алгоритм под названием *GAME* (*Group of Adaptive Models Evolution*), базирующийся на архитектуре МГУА, разработан в Чешском техническом университете в Праге. Главные модификации этой МГУА-подобной системы [28] также базируются на идее активных нейронов:

- передаточная функция каждого узла этой гетерогенной сети может быть линейной, полиномиальной, логистической, РБФ и другой, а также в виде перцептрона, причем выбор типа узлов, образующих эту сеть, определяется минимальным значением критерия селекции;

- количество входов каждого узла равно номеру слоя сети, где находится этот узел;

- в сети существуют междурядные связи, и в процессе построения сети находятся не все возможные размещения узлов, а только их случайные подмножества;

- оригинальный МГУА дает одну оптимальную модель, а *GAME* – группу лучших моделей, локально оптимальных для своего подмножества размещений.

В этом МГУА-подобном алгоритме передаточные функции, начальные значения весов и коэффициентов узлов, а также их входы выбираются случайно, поэтому топология моделей, построенных по одной обучающей выборке, может отличаться.

Другие пути развития идей самоорганизации. Имеется много новых тенденций в развитии методов самоорганизации моделей на основе МГУА: создание гибридных архитектур в сочетании с алгоритмами вычислительного интеллекта [29], применение распараллеливания операций [30], построение нейросетей с нечеткими функциями активации [31] и мн. др.

Архитектура обобщенного итерационного алгоритма ОИА МГУА

Подход к конструированию обобщенного алгоритма. Из анализа следует, что для дальнейшего развития архитектур итерационных алгоритмов необходимо использовать две основ-

ные идеи [32]: добавление на каждом ряду исходных аргументов (селекция первичных аргументов); реализация «активных нейронов» в форме комбинаторной оптимизации структуры частных моделей. Рассмотрим, как комплексная реализация этих двух идей позволяет обобщать большинство архитектур имеющихся алгоритмов МГУА и конструировать новые.

По аналогии с аббревиатурой МИА МГУА алгоритм типа [18], в котором возможны сочетания в частных описаниях пар аргументов как только из промежуточных переменных (6), так и из промежуточных и исходных (8), следует назвать «комбинированным итерационным алгоритмом» КИА МГУА. Если же разрешить только второй вариант описания вида (8), получим новый алгоритм релаксационного типа, который естественно назвать РИА МГУА, и он вместе с МИА будет очевидным частным случаем алгоритма КИА.

Если в каждой из этих трех структур алгоритмов применить комбинаторную оптимизацию сложности частных описаний вида (4) – как линейных, так и нелинейных, – получим еще три варианта итерационно-комбинаторных алгоритмов МГУА: многорядный итерационно-комбинаторный алгоритм МИКА; релаксационный итерационно-комбинаторный алгоритм РИКА; комбинированный итерационно-комбинаторный алгоритм КИКА.

Таким образом, в архитектуре КИКА можно выделить все шесть названных вариантов структур алгоритмов, три из которых – МИА [1], КИА [18] и МИКА [14] – были известны ранее, а еще три, а именно РИА, РИКА и собственно КИКА, следует считать новыми. Это типичный обобщенный алгоритм, построенный на основе идеи гибридизации структур итерационных алгоритмов МГУА и комбинаторного алгоритма *COMBI* МГУА [33].

Комбинаторная оптимизация частных описаний. В обобщенном алгоритме КИКА на каждом ряде для формирования моделей применяются частные описания (4) в форме (6) или (8), и для любого из них можно выполнить перебор всех возможных вариантов описаний

меньшей сложности и выбрать лучший. Покажем это на примере линейного описания вида

$$f(u, v) = a_0 d_1 + a_1 d_2 u + a_2 d_3 v, \quad (10)$$

где $d_k, k=1,2,3$, $d_k = \{0, 1\}$ – элементы структурного вектора $d = (d_1 d_2 d_3)$, принимающие значения единица или ноль, что указывает на включение или исключение соответствующего члена в (10). Очевидно, что в данном случае имеется семь возможных вариантов составления различных структурных векторов: (100), (010), (001), (110), (101), (011), (111), а в общем случае описания, состоящего из p одночленов, их будет $q = 2^p - 1$. Тогда оптимальную частную модель можно представить как $f(u, v, d_{\text{opt}})$, где структурный вектор d_{opt} соответствует тому варианту частной модели меньшей сложности, для которого значение критерия (5) – наименьшее:

$$d_{\text{opt}} = \arg \min_{l=1, q} CR_l, \quad q = 2^p - 1, \\ f_{\text{opt}}(u, v) = f(u, v, d_{\text{opt}}). \quad (11)$$

Это выражение описывает оптимизацию сложности частной модели по комбинаторному алгоритму для формирования активных нейронов в полиномиальной нейросети, которая здесь представлена итерационно-комбинаторным алгоритмом КИКА МГУА. Его дальнейшее обобщение, учитывающее как КИКА и все его частные случаи, так и различные варианты режимов работы, реализуемые с помощью тем или иным способом построенного интерфейса, будем называть «Обобщенным итерационным алгоритмом» ОИА МГУА.

Обобщенный итерационный алгоритм. Формально в общем случае для ряда r определим ОИА МГУА так:

- 1) входной матрицей есть $X_{r+1} = (y_1^r, \dots, y_F^r, x_1, \dots, x_m)$;
- 2) применяются операторы перехода вида (6) и (8) с квадратичным частным описанием из (4);
- 3) для каждого описания находится оптимальная структура (11);
- 4) алгоритм останавливается при выполнении условия $CR_{\min}^{r+1} \geq CR_{\min}^r$, и оптимальная

модель соответствует значению CR_{\min}^r на r -м ряде.

Разработанная гибридная архитектура ОИА МГУА обеспечивает эффективный комплекс новых свойств: восстановление информативных аргументов, отсеянных на первых этапах алгоритма; исключение неинформативных аргументов, оставленных на первых этапах; предотвращение переусложнения модели вследствие оптимизации сложности частных моделей; устранение «вырождения» частных моделей, т.е. данная алгоритмическая структура позволяет не только обобщить основные структуры разработанных ранее итерационных алгоритмов МГУА и одновременно получить новые их варианты, но и в комплексе устранить все четыре отмеченных ранее недостатка классического многорядного алгоритма МГУА.

Упорядоченное кодирование структур итерационных алгоритмов. Определим ОИА МГУА как множество итерационных и итерационно-комбинаторных алгоритмов, описываемое вектором из трех элементов DM (*Dialogue Mode*), IC (*Iterative-Combinatorial*), MR (*Multi-layered-Relaxative*), т.е. любой итерационный алгоритм определяется как частный случай обобщенного ОИА (DM, IC, MR). При этом DM принимает три значения: 1 – стандартный автоматический режим, 2 – планируемый автоматический режим, 3 – интерактивный режим; IC – два значения: 1 – итерационный, 2 – итерационно-комбинаторный алгоритмы; MR – три значения: 1 – классический итерационный, 2 – релаксационный, 3 – комбинированный алгоритмы.

Такой вектор полностью определяет собой тот или иной алгоритм: при $DM = 1$ имеем три стандартных варианта итерационных алгоритмов МИА = ОИА (1, 1, 1), РИА = ОИА (1, 1, 2), КИА = ОИА (1, 1, 3), и три итерационно-комбинаторных: МИКА = ОИА (1, 2, 1); РИКА = ОИА (1, 2, 2); КИКА = ОИА (1, 2, 3). При DM , равном двум или трем, сформируются новые варианты этих алгоритмов.

Соответствующая иерархическая структура обобщенного итерационного алгоритма изображена на рис. 3 (белые блоки – известные алго-

ритмы, серые – новые структуры), причем указанные алгоритмы исчерпывают множество возможных частных случаев ОИА МГУА. Однако существуют и другие варианты итерационных алгоритмов МГУА, не укладывающиеся в эту архитектуру – например, сюда не вписываются алгоритмы [17, 20, 27, 28].

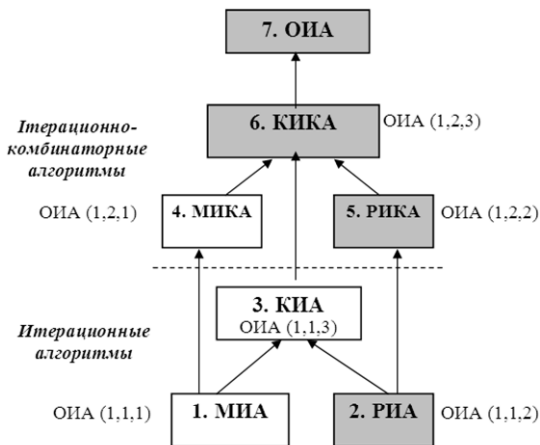


Рис. 3. Иерархия разработанной архитектуры итерационных алгоритмов МГУА

Методика численного анализа эффективности итерационных алгоритмов МГУА

Цель экспериментов – сравнительное исследование эффективности вариантов итерационных алгоритмов МГУА как частных случаев обобщенного. Методика проведения экспериментов аналогична разработанной в [34] для переборных алгоритмов МГУА. Каждый из итерационных алгоритмов характеризуется своими входными (управляющими) параметрами, а также промежуточными и выходными показателями качества процесса построения модели.

Входные параметры алгоритмов: число начальных аргументов m ; число точек в выборке n ; способ разбиения выборки W на обучающую A и проверочную B подвыборки; критерий селекции моделей; свобода выбора моделей F .

Промежуточные показатели качества: график изменения минимальных и максимальных значений критерия по рядам; изменение состава аргументов в лучшей модели по рядам.

Выходные показатели качества: лучшая модель и ее параметры; соответствующее значение критерия селекции; состав аргументов лучшей модели и ее близость к истинной; время

работы программы (время работы центрального процессора).

Методика численного исследования влияния управляющих параметров на эффективность алгоритмов состоит из следующих основных этапов:

- генерации матрицы входов $X[n \times m]$ из случайных чисел с заданным распределением;
- задания конкретного вида зависимости выхода не от всех, а только от определенной части «истинных» аргументов $X = [X \tilde{X}]$, где X – истинные, \tilde{X} – лишние аргументы;
- вычисления значений истинного выходного вектора $y = Xb$, где b – вектор точных параметров, и формирования выборки $W = (Xy)$, где y может содержать наложенный шум;
- построения оптимальных моделей по каждому алгоритму при различных значениях управляющих параметров, основным из которых есть свобода выбора F .

Методика позволяет изучать влияние управляющих параметров на промежуточные и выходные показатели процесса построения модели и выбирать наиболее эффективный алгоритм. На ее основе были исследованы свойства типовых итерационных алгоритмов. Некоторые из полученных результатов описаны в [35].

Исследование эффективности алгоритмов в задаче восстановления истинной модели

Цель – сравнение эффективности итерационных алгоритмов в зависимости от их управляющих параметров. Все эксперименты проведены с применением критерия регулярности $AR(5)$ на компьютере с процессором *AMD Phenom™ 9650 Quad-Core Processor*.

Задача выявления истинной структуры модели при наличии лишних аргументов, случай линейной зависимости. Матрица X получена с применением датчика случайных равномерно распределенных чисел в интервале от нуля до пяти. Выборка состоит из 200 аргументов, из которых 10 информативных и 190 лишних. Истинная модель зависит от 10 первых аргументов:

$$y = -3x_1 - 3x_2 + 5x_3 - x_4 - x_5 + 3x_6 + x_7 - 2x_8 + x_9 + x_{10} . \quad (12)$$

Длина выборки наблюдений $n = 243$, разбиение на обучающую и проверочную подвыборки: $n_A = 173$, $n_B = 60$. Изучено влияние свободы выбора F на эффективность восстановления структуры истинной модели. Проведено сравнение четырех алгоритмов: многорядного МИА; релаксационного РИА; комбинированного КИА; обобщенного ОИА.

На рис. 4 показано изменение значений критерия селекции для разных алгоритмов при росте свободы выбора. Поскольку уже при $F = 20$ результаты практически не меняются, для такого F в табл. 1 приведена характеристика эффективности всех алгоритмов. Истинную зависимость воспроизводят все алгоритмы, кроме классического многорядного, в котором были потеряны истинные аргументы x_4 , x_7 , x_9 , и ОИА имеет наименьшее значение критерия AR .

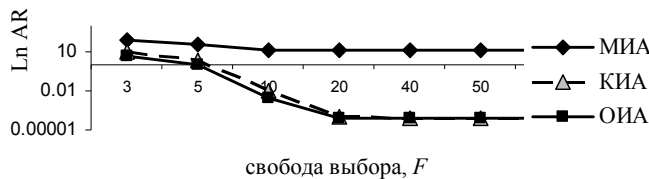


Рис. 4. Влияние свободы выбора на изменения значений критерия селекции

Таблица 1. Характеристика эффективности алгоритмов

Алгоритмы	F = 20			Истинные одночлены										Лишние одночлены
	r	AR	Время, мин	x ₁	x ₂	x ₃	x ₄	x ₅	x ₆	x ₇	x ₈	x ₉	x ₁₀	
1	9	12,988	0,09	+	+	+		+	+		+	+	+	9
2	23	0,0002	3,52	+	+	+	+	+	+	+	+	+	+	0
3	21	0,0001	4,10	+	+	+	+	+	+	+	+	+	+	0
4	36	0,00008	15,4	+	+	+	+	+	+	+	+	+	+	0

Выявление нелинейной истинной структуры модели, случай квадратичной зависимости. Исследования выполнялись по данным той же матрицы X . Выборка имеет 200 аргументов, из них четыре информативных и 196 лишних, и делится на три части: $n_A = 173$, $n_B = 60$, $n_C = 10$, где C – экзаменационная часть. Выходная величина представлена квадратичной зависимостью:

$$y = -3x_1 - 2x_2x_3 + x_{25}^2 + x_{25}x_1 + 12 . \quad (13)$$

Для сравнения рассмотрены все шесть основных вариантов итерационных алгоритмов: многорядный, релаксационный, комбинированный, многорядно-комбинаторный, релаксационно-комбинаторный, обобщенный итерационный.

Сравнивая результаты (табл. 2), можно сделать вывод об эффективности ОИА МГУА. Из табл. 1 и 2 видно, что этот алгоритм как для линейной, так и для квадратичной зависимостей восстанавливает истинную модель и по структуре, и по параметрам.

Таблица 2. Сравнение модификаций многорядного алгоритма для квадратичной зависимости

Алгоритмы	AR(B)	AR(C)	Модель
1	11,309	2,786	$\hat{y} = -3x_1 - 1,231x_2 + 2,711x_{25} + 0,998x_{32} + x_{37} - 2,001x_{45} + 12$
2	3,031	0,841	$\hat{y} = -3x_1 - 1,231x_2 + 6,700x_{25} - 2,001x_{45} + 11,99$
3	3,011	0,841	$\hat{y} = -3x_1 - 1,231x_2 + 6,711x_{25} - 2,001x_{45} + 12$
4	0,462	0,322	$\hat{y} = -2,0001x_2x_3 + x_{25}^2 + x_{25}x_1 - 2,999x_{45} + 12$
5	$4 \cdot 10^{-7}$	0,089	$\hat{y} = -3,000x_1 - 2,000x_2x_3 + x_{25}^2 + 0,985x_{25}x_1 + 12,000$
6	$3 \cdot 10^{-8}$	0,001	$\hat{y} = -3,000x_1 - 2,000x_2x_3 + x_{25}^2 + x_{25}x_1 + 12,000$

Эффективность алгоритмов в условиях зашумленных данных. Матрица получена по датчику случайных равномерно распределенных чисел в интервале от нуля до единицы. Выборка состоит из 40 аргументов, из которых три информативных и 37 лишних, и делится на $n_A = 40$, $n_B = 20$. Выходная величина, не содержащая шума, представлена зависимостью

$$y = 0,5 - 1,2x_2 + 5x_{10} - 3,4x_{25} . \quad (14)$$

Эксперименты показали, что при наложении на выход равномерного шума с отношением шум/сигнал 10% и 30% лучшим алгоритмом, воспроизводящим истинную структуру (14), остается обобщенный, хотя точность оценок параметров уменьшается. На рис. 5 показана эффективность обобщенного алгоритма в сравнении с многорядным и комбинированным.

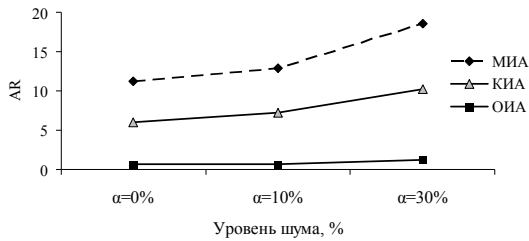


Рис. 5. Значения критерия регулярности при разном уровне шума

Эффективность построения существенно нелинейной зависимости от малого числа аргументов. Исследование проводилось на данных, полученных по датчику случайных чисел в интервале от нуля до единицы. Выборка содержит $m = 3$ аргумента и $n = 100$ точек и делится на две части: $n_A = 65$, $n_B = 35$. Исследовано влияние свободы выбора F на эффективность алгоритмов.

Отметим, что в многорядном алгоритме использовано расширение начального базиса до полного куба с последующим использованием линейных частных описаний, поскольку при $m = 3$ возможны только три комбинации пар на всех рядах, и F ограничена значением три, а это недостаточно для исследования. Истинная модель имеет вид нелинейной функции от трех аргументов:

$$y = 7 - 6x_1 + 5x_3 - 4x_2^2 + 3x_1x_3 - 2x_1^2x_2 + x_3^3. \quad (15)$$

На рис. 6 показано изменение значений критерия селекции AR для различных алгоритмов в зависимости от свободы выбора. Наименьшие значения критерия были достигнуты для ОИА, причем была получена модель истинной структуры.

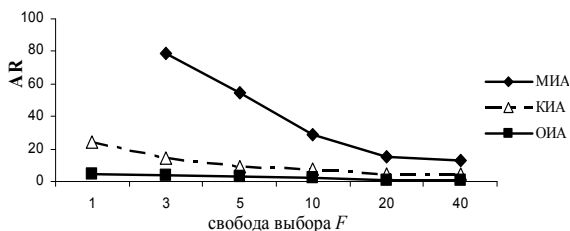


Рис. 6. Изменение значений критерия при увеличении свободы выбора моделей

Для свободы выбора $F = 20$ в табл. 3 представлена полная характеристика эффективности обобщенного алгоритма по рядам селекции, причем показано изменение минималь-

ных и максимальных значений критерия селекции для разных рядов r .

Таблица 3. Характеристика алгоритма ОИА по рядам r при $F = 20$

r	Показатели при $F = 20$		Наличие истинных одночленов						Число лишних одночленов
	AR_{\min}	AR_{\max}	x_1	x_3	x_2^2	x_1x_3	$x_1^2x_2$	x_3^3	
1	2035,3	4288,7		+					1
2	70,58	152,3	+	+	+	+			4
3	5,76	17,44	+	+	+	+		+	5
4	4,04	28,12	+	+	+	+	+	+	4
5	3,37	5,70	+	+	+	+	+	+	3
...
10	0,88	2,36	+	+	+	+	+	+	1
11	1,22	4,58	+	+	+	+	+	+	3

На рис. 7 показана зависимость значений критерия от номера ряда при различных F для обобщенного алгоритма. Видно, что при $F = 3$ и $F = 5$ минимум достигается на девятом ряду, а при $F = 10$ и $F = 20$ – на 10-м, т.е. при увеличении F число рядов, необходимых для достижения минимума критерия, растет, а собственно значение минимума уменьшается, пока точка минимума и минимальное значение не стабилизируются, что говорит о нахождении истинной модели.

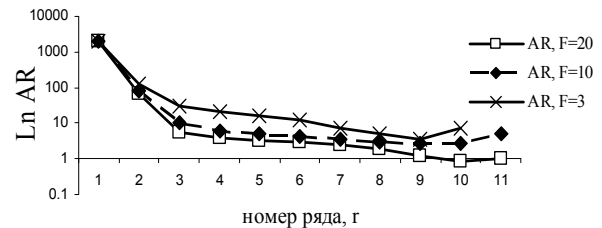


Рис. 7. Изменение критерия AR по рядам ОИА при разной свободе выбора

Лучшая модель получена за 2 мин. 23 с. по обобщенному алгоритму МГУА на 10-м ряде при оптимальной свободе выбора $F = 20$, когда оптимальная структура содержит семь аргументов, из которых шесть истинных и только один лишний. В других алгоритмах затрачено меньше времени, но количество рядов было больше, причем полученные модели содержат не все информативные одночлены, однако включают в себя много лишних.

Исследование внутренней сходимости итерационных алгоритмов МГУА

Понятие сходимости – одно из центральных в математической теории итерационных алго-

ритмов МГУА. Теоретически проблема сходимости алгоритмов МГУА рассматривалась в литературе лишь в нескольких работах [7, 19, 22, 23], причем только для двух алгоритмов [19, 23] строго доказана «внутренняя» сходимость.

Итерационный алгоритм имеет внутреннюю сходимость, если в результате итерационного процесса по критерию селекции $CR = RSS$, где RSS – остаточная сумма квадратов на всей выборке W (без разбиения), оценки параметров оптимальной модели совпадают с оценками параметров модели истинной структуры, полученными по МНК.

Численное исследование внутренней сходимости трех итерационных алгоритмов МГУА – классического МИА, комбинированного КИА и обобщенного ОИА – проводилось по данным, полученным по датчику случайных чисел в интервале $[0, 5]$. Выборка содержит $m = 10$ аргументов и $n = 50$ точек, разбиение не применяется.

Истинная модель имеет вид линейной функции от десяти аргументов:

$$y = 3 - 2x_1 + 5x_2 - x_3 + 7x_4 - 2x_5 + 4x_6 - 3x_7 + 2,5x_8 - 1,8x_9 + x_{10}. \quad (16)$$

На рис. 8 отражен процесс сходимости алгоритмов.

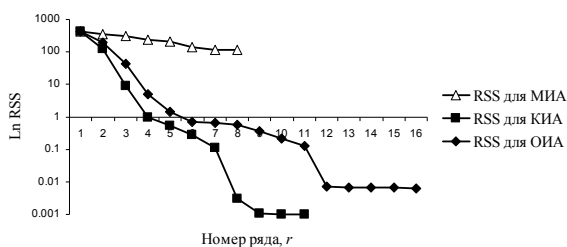


Рис. 8. Изменение RSS по рядам селекции для трех сравниваемых алгоритмов при $F = 20$

Уравнение множественной регрессии по этим данным имеет вид:

$$\hat{y}_{\text{рег}} = 3 - 2x_1 + 4,999x_2 - x_3 + 6,999x_4 - 2x_5 + 3,999x_6 - 3x_7 + 2,499x_8 - 1,8x_9 + x_{10}. \quad (17)$$

По трем сравниваемым алгоритмам получены следующие модели:

$$y_{\text{МИА}} = -22,387 + 5,334x_2 - 0,535x_3 + 5,266x_4 + 0,812x_5 + 4,726x_6 + 1,278x_8 - 2,643x_9 + 0,437x_{10}, \quad (18)$$

$$y_{\text{КИА}} = 2,913 - 1,978x_1 + 4,999x_2 - 1,015x_3 + 7,004x_4 - 1,988x_5 + 3,997x_6 - 2,994x_7 + 2,514x_8 - 1,808x_9 + 1,003x_{10}, \quad (19)$$

$$y_{\text{ОИА}} = 3,001 - 1,999x_1 + 4,999x_2 - 1,002x_3 + 7,004x_4 - 1,998x_5 + 4,000x_6 - 2,999x_7 + 2,500x_8 - 1,801x_9 + 1,003x_{10}. \quad (20)$$

Очевидно, что алгоритм МИА сходится по критерию (рис. 8), но не имеет внутренней сходимости по структуре, поскольку в (18) отсутствуют истинные аргументы x_1 и x_7 .

Уравнения (19), (20) в целом подтверждают внутреннюю сходимость двух итерационных алгоритмов КИА и ОИА как по критерию, так и по структуре и параметрам, поскольку значения оцененных параметров близки к истинным (16) и (17). Обобщенный алгоритм достигает минимума на девятом ряду, а комбинированный – только на 12-м (см. рис. 8). Точность полученных моделей также выше у обобщенного: для ОИА $RSS = 0,001$, для КИА $RSS = 0,006$. Это демонстрирует эффективность использования оптимизации частных моделей.

Для обобщенного алгоритма в табл. 4 показано изменение структуры и параметров лучших

Таблица 4. Изменение структуры и параметров лучшей модели по рядам ОИА МГУА

Ряд, r	RSS для ОИА	Модель
1	416,15	$\hat{y} = -15,806 + 5,521x_2 + 6,581x_4$
2	121,75	$\hat{y} = -7,220 + 4,434x_2 + 5,286x_4 + 5,604x_6 - 1,988x_9$
3	9,20	$\hat{y} = 7,084 + 4,782x_2 + 5,700x_4 - 1,073x_5 + 4,467x_6 - 3,412x_7 + 2,177x_8 - 2,647x_9 + 0,905x_{10}$
4	0,98	$\hat{y} = 2,245 - 1,953x_1 + 5,028x_2 - 1,019x_3 + 7,054x_4 - 2,008x_5 + 4,016x_6 - 2,968x_7 + 2,492x_8 - 1,801x_9 + 0,962x_{10}$
...
8	0,003	$\hat{y} = 3,001 - 1,999x_1 + 4,999x_2 - 1,003x_3 + 7,005x_4 - 1,998x_5 + 4,000x_6 - 2,999x_7 + 2,501x_8 - 1,801x_9 + 1,003x_{10}$
9	0,0011	$\hat{y} = 3,001 - 1,999x_1 + 4,999x_2 - 1,002x_3 + 7,004x_4 - 1,998x_5 + 4,000x_6 - 2,999x_7 + 2,501x_8 - 1,801x_9 + 1,003x_{10}$
10	0,0010	$\hat{y} = 3,001 - 1,999x_1 + 4,999x_2 - 1,002x_3 + 7,004x_4 - 1,998x_5 + 4,000x_6 - 2,999x_7 + 2,500x_8 - 1,801x_9 + 1,003x_{10}$

моделей по рядам. Видно, что истинная структура была найдена уже на четвертом ряду, после чего происходило только уточнение значений параметров, т.е. ОИА МГУА можно применять не только для структурно-параметрической идентификации, но и как итерационную процедуру оценивания параметров моделей по условию минимизации остаточной суммы квадратов RSS .

Заключение. В результате сравнительного анализа преимуществ и недостатков имеющихся алгоритмов индуктивного моделирования выявлены возможности повышения эффективности решения задач построения моделей на основе обобщения типичных структур итерационных алгоритмов МГУА. Разработан обобщенный итерационный алгоритм ОИА МГУА, частными случаями которого есть как известные, так и новые разновидности многорядных, релаксационных и итерационно-комбинаторных алгоритмов, что дает возможность сравнительного исследования эффективности разных алгоритмов и решения реальных задач моделирования.

Комплексная методика численного анализа эффективности итерационных алгоритмов МГУА позволяет всесторонне изучать влияние ключевых параметров сравниваемых алгоритмов на основные показатели качества результатов моделирования. В частности, для обобщенного итерационного алгоритма в результате экспериментов установлен факт сходимости к истинной полиномиальной модели как по структуре, так и по параметрам.

1. *Ивахненко О.Г.* Метод группового учета аргументов – конкурент методу стохастичної апроксимації // Автоматика. – 1968. – № 3. – С. 58–72.
2. *Ивахненко А.Г.* Системы эвристической самоорганизации в технической кибернетике. – Киев: Техніка, 1971. – 392 с.
3. *Ивахненко А.Г.* Долгосрочное прогнозирование и управление сложными системами. – Там же, 1975. – 311 с.
4. *Ивахненко А.Г.* Индуктивный метод самоорганизации моделей сложных систем. – Киев: Наук. думка, 1982. – 296 с.
5. www.mgua.irtc.org.ua
6. *Ивахненко А.Г., Степашко В.С.* Помехоустойчивость моделирования. – Киев: Наук. думка, 1985. – 216 с.

7. *Ивахненко А.Г., Юрачковский Ю.П.* Моделирование сложных систем по экспериментальным данным. – М.: Радио и связь, 1987. – 120 с.
8. *Степашко В.С.* Элементы теории индуктивного моделирования. Стан та перспективи розвитку інформатики в Україні / Кол. авт. – К.: Наук. думка, 2010. – С. 481–496.
9. *Gupta S., Bhardwaj S., Bhatia P.K.* A reminiscent study of nature inspired computing // Int. J. of Advances in Engin. & Technol. – 2011. – 1. – Issue 2. – P. 117–125.
10. *Ивахненко А.Г.* Метод группового учета аргументов в задачах прогнозирования // Автоматика. – 1976. – № 6. – С. 24–33.
11. *Трисеев Ю.П.* Алгоритм МГУА с изменением свободы выбора по рядам селекции на основе критерия разнообразия переменных // Там же. – 1977. – № 4. – С. 37–42.
12. *Parker R.G.J., Tummala M.* Identification of Volterra systems with a polynomial neural network / Proc. of the 1992 IEEE Int. Conf. on Acoustics, Speech and Signal Processing ICASSP'92. – 1992. – 4. – P. 561–564.
13. *Hara K., Yamamoto T., Terada K.* Improved dual mode GMDH with automatic switch // Int. J. of Systems Science. – 1990. – 21, N 8. – P. 1553–1565.
14. *Ивахненко Н.А., Марчев А.А.* Самоорганизация математической модели для перспективного планирования строительно-монтажных работ // Автоматика. – 1978. – № 3. – С. 12–18.
15. *Справочник по типовым программам моделирования* / Под ред. Ивахненко А.Г. – Киев: Техника, 1980. – 184 с.
16. *Ikeda S., Fujishige S., Sawaragi Y.* Non-linear prediction model of river flow by selforganization method // Int. J. of Systems Science. – 1976. – 7, N 2. – P. 165–176.
17. *Шелудько О.И.* Алгоритм МГУА с ортогонализированным полным описанием для синтеза моделей по результатам планируемого эксперимента // Автоматика. – 1974. – № 5. – С. 32–42.
18. *Светальский Б.К., Ковальчук П.И.* Многорядный алгоритм МГУА с селекцией первичных аргументов // Там же. – 1979. – № 4. – С. 31–35.
19. *Юрачковский Ю.П.* Сходимость многорядных алгоритмов МГУА // Там же. – 1981. – № 3. – С. 32–36.
20. *Юрачковский Ю.П.* Восстановление полиномиальных зависимостей на основе самоорганизации // Там же. – № 4. – С. 15–20.
21. *Aksyonova T.I., Volkovich V.V., Tetko I.V.* Robust Polynomial Neural Network in Quantative-structure Activity Relationship Studies // Systems analysis modeling simulation. – 2003. – 43, N 10. – P. 1331–1341.
22. *Ивахненко О.Г., Ковальчук П.И., Тодуа М.М.* Про єдиність відновлення кривої регресії за малим числом точок // Автоматика. – 1973. – № 5. – С. 35–49.
23. *Павлов А.В., Кондрашова Н.В.* О сходимости обобщенного релаксационного итерационного алгоритма

- метода групового учета аргументов // УСиМ. – 2012. – № 3. – С. 24–29, 38.
24. Павлов А.В. Обобщенный релаксационный итерационный алгоритм МГУА / Индуктивное моделирование складных систем: Зб. наук. праць. – К.: МННЦІТ та С НАНУ, 2011. – 4. – С. 121–134.
 25. Ivakhnenko A.G., Ivakhnenko G.A., Muller J.A. Self-Organization of Neuronets with Active Neurons // Patt. Recognition and Image Analysis. – 1994. – 4, N 4. – P. 177–188.
 26. Muller J.-A., Lemke F. Self-Organizing data mining. An intelligent approach to extract knowledge from data. – Berlin, Dresden, 1999. – 225 p.
 27. Kondo T., Ueno J. Feedback GMDH-Type Neural Network Self-Selecting Optimum Neural Network Architecture and its Application to 3-Dimensional Medical Image Recognition of the Lungs / Proc. of the II Int. Workshop on Inductive Modelling IWIM–2007, 19–23 Sept. 2007, Prague. – Prague: Czech Technical University, 2007. – P. 63–70.
 28. Kordik P. Fully automated knowledge extraction using group of adaptive model evolution: PhD thesis / Electrical Engineering and Information Technology. – Prague: CTU, 2006. – 150 p.
 29. Kovarik O., Kordik P. Optimizing Models Using Continuous Ant Algorithms / Proc. of the 2nd Int. Conf. on Inductive Modelling ICIM–2008. – K.: IRTC ITS NANU, 2008. – P. 124–128.
 30. Lemke F. Parallel Self-Organizing Modeling / Ibid. – P. 176–184.
 31. Bodyanskiy Ye.V., Zaychenko Yu.P., Pavlikovskaya Ye. The Neo-Fuzzy Neural Network Structure Optimization Using the GMDH for the Solving Forecasting and Classification Problems / Proc. of the 3rd Int. Workshop on Inductive Modelling IWIM–2009, 14–19 Sept. 2009, Krynica, Poland. – Prague: Czech Technical University, 2009. – P. 100–107.
 32. Степашко В.С., Булгакова О.С., Зосімов В.В. Гібридні алгоритми самоорганізації моделей для прогнозування складних процесів / Индуктивное моделирование складных систем: Зб. наук. праць. – К.: МННЦІТ та С НАНУ, 2010. – 2. – С. 236–246.
 33. Степашко В.С. Комбинаторный алгоритм МГУА с оптимальной схемой перебора моделей // Автоматика. – 1981. – № 3. – С. 31–36.
 34. Степашко В.С., Костенко Ю.В. Исследование свойств комбинаторно-селекционного (многоэтапного) алгоритма МГУА / Моделирование и управление состоянием эколого-экономических систем региона: Сб. науч. тр. – К.: ИК НАНУ, 2001. – С. 69–76.
 35. Булгакова О.С., Степашко В.С. Порівняльний аналіз ефективності ітераційних алгоритмів МГУА за допомогою обчислювальних експериментів // Вісн. ЧДТУ. – 2011. – № 1. – С. 41–44.

Тел. для справок: +38 044 266-3028, +38 0512 37-8809,
+38 068 267-5118 (Київ, Николаев)

E-mail: stepashko@irtc.org.ua

© В.С. Степашко, А.С. Булгакова, 2013

Внимание !

**Оформление подписки для желающих
опубликовать статьи в нашем журнале обязательно.**

В розничную продажу журнал не поступает.

Подписной индекс 71008