

Рис. 1.

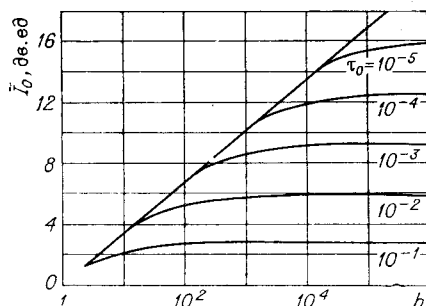


Рис. 2.

График предельной удельной информативности системы в зависимости от относительной величины быстрых флюктуаций времени ( $T_\tau = T_0$ ) представлен на рис. 1. Для сравнения на рис. 2 дан график зависимости удельной информативности от отношения сигнал/шум при фиксированных значениях относительной величины смещений времени.

Из приведенных зависимостей видно, что флюктуации временных соотношений в сигнале принципиально приводят к снижению его информативности. Более того, смещения времени определяют верхнюю границу точности информационно-измерительной системы. Так, например, при величине относительной нестабильности времени  $\tau_0 = 10^{-3}$  информативность системы не может превышать 9 двоичных единиц на один отсчет. Таким образом, используя полученные аналитические выражения и графики, можно определить требования к стабильности временных соотношений в измерительной системе, исходя из требуемой ее информативности.

#### ЛИТЕРАТУРА

1. В. С. Пугачев. Теория случайных функций. М., Физматгиз, 1962.
2. И. М. Гельфанд, А. М. Яглом. О вычислении количества информации о случайной функции, содержащейся в другой такой же функции.— Успехи матем. наук, 1957, т. 12, вып. 1 (73).
3. А. Н. Колмогоров. Стационарные последовательности в гильбертовом пространстве.— Бюлл. МГУ 2, № 6, 1941.
4. А. Н. Колмогоров. Интерполирование и экстраполирование стационарных последовательностей.— Изв. АН СССР, сер. матем., 1941, т. 5, № 1.
5. К. Шеннон. Работы по теории информации и кибернетике. М., «Иностранная литература», 1963.
6. А. Н. Малахов. Флуктуации в автоколебательных системах. М., «Наука», 1968.
7. Б. Р. Левин. Теоретические основы статистической радиотехники, т. 1. М., «Советское радио», 1969.

Поступила в редакцию 24 мая 1973 г.

УДК 62-50

К. К. ПАЩЕНКО, В. А. СТАНОВОВА  
(Караганда)

### О ТЕНЗОРНОЙ СТОХАСТИЧЕСКОЙ АППРОКСИМАЦИИ И ЕЕ ПРИЛОЖЕНИИ К ЗАДАЧАМ ФИЛЬТРАЦИИ, ИДЕНТИФИКАЦИИ И РАСПОЗНАВАНИЯ ОБРАЗОВ

Регрессивные схемы, получившие название алгоритмов стохастической аппроксимации (СА), впервые были применены для скалярных последовательностей [1] и затем обобщены на векторные величины [2].

Существует настоятельная необходимость в распространении этих алгоритмов для самых общих линейных алгебраических объектов, т. е. для тензорных последовательностей.

Сведение стохастических аппроксимаций для тензоров ранга более 1 к векторным алгоритмам приводит к громоздким и трудно обозримым соотношениям, следствием чего могут быть даже ошибки в результирующих схемах.

Постановке вопроса о тензорных СА предположим примеры тензорных процедур валентности 2, 3 и 4, к которым сводятся некоторые задачи фильтрации, идентификации и распознавания образов.

**Линейный обучающийся фильтр.** Ко входу линейного фильтра, представленного уравнениями состояния, приложена аддитивная смесь многомерного сигнала ( $U$ ) и шума ( $V$ ):

$$\begin{aligned} X(t) &= AX(t) + B(U(t) + V(t)), \\ Z(t) &= DX(t), \end{aligned} \quad (1)$$

или для дискретного времени

$$\begin{aligned} X[n] &= FX[n-1] + B(U[n-1] + V[n-1]), \\ Z[n] &= DX[n]. \end{aligned}$$

Требуется, чтобы выходная величина фильтра ( $Z$ ) заданной размерности воспроизводила некоторое линейное преобразование от полезного сигнала ( $S$ ) наилучшим образом в смысле минимума квадратичного функционала

$$\begin{aligned} I &= M\langle \|Z[n] - S[n]\|_G \rangle = \\ &= M\langle (Z[n] - S[n])^T G (Z[n] - S[n]) \rangle, \\ S[n] &= DY[n], \quad Y[n] = \Phi Y[n-1] + BU[n-1]. \end{aligned} \quad (2)$$

В (1) и (2) использованы следующие матрицы и векторы:

$$\begin{aligned} X &= \{x_i\}; \quad Y = \{y_i\}; \quad A = \{a_{ij}\}; \quad \Phi = \{\varphi_{ij}\}; \quad F = \{f_{ij}\}, \quad i, j = \overline{1, r}; \\ U &= \{u_i\}; \quad V = \{v_i\}; \quad B = \{\beta_{ij}\}; \quad B = \{b_{ij}\}, \quad i = \overline{1, r}, \quad j = \overline{1, s}; \\ G &= \{g_{jk}\}; \quad S = \{s_j\}; \quad Z = \{z_j\}; \quad D = \{d_{ij}\}, \quad i = \overline{1, r}, \quad j = \overline{1, p}; \end{aligned}$$

$M\langle \rangle$  — математическое ожидание, матрицы  $A, F, D, \Phi, B, B$  здесь не зависят от времени,  $G$  — матрица, задающая веса компонент векторов  $Z$  и  $S$  в функционале,  $T$  — знак транспонирования матрицы.

Для достаточно малых дискретов времени  $\Delta t$

$$\Phi \cong (I + A\Delta t); \quad B = B\Delta t,$$

где  $I$  — единичная матрица.

Если вместо априорной информации о статистиках сигнала и помехи имеется лишь набор реализаций полезного сигнала и его смеси с помехой, то, следуя [3], имеем случай обучающегося винеровского фильтра, условием оптимальности которого является

$$\nabla I(A) = M_v \langle \nabla_A \|Z[n] - S[n]\|_G \rangle = 0. \quad (3)$$

В отличие от [4], где вычисляется градиент скалярной функции, заданной в линейном конечномерном векторном пространстве, в (3) необходимо найти производную скалярной функции, определенной в пространстве тензоров второй валентности  $A$ , т. е.

$$\nabla I(A) = Q = \{q_{ij}\}; \quad q_{ij} = \frac{\partial I(A)}{\partial a_{ij}}. \quad (4)$$

Операция тензорного дифференцирования скалярных, векторных и в общем случае тензорных функций тензорной переменной ранга 2 и выше разработана недостаточно хорошо. Лишь в [5] довольно полно рассмотрен один частный, но весьма важный случай тензорной производной скалярной функции тензорной переменной, а именно матричное дифференцирование скалярной функции. Приведем некоторые формулы тензорного дифференцирования, использованные в работе:

$$\left. \begin{aligned} \nabla_A (X^T A^T G A Y) &= (G A Y X^T + G^T A X Y^T), \\ \nabla_A (X^T A Y) &= X Y^T, \end{aligned} \right\} \quad (5)$$

$$\left. \begin{aligned} \nabla_A S_p (B A C) &= A^T C^T, \\ \nabla_A S_p (B A^T A C) &= B A^T C + B C^T A, \end{aligned} \right\} \quad (6)$$

$$\nabla_A (a_{ij\dots p} b_{ij\dots p}) = b_{ij\dots p}. \quad (7)$$

В соответствии с (7)

$$Q[n] = \nabla_A \|Z[n] - S[n]\|_0 = 2\Delta t D^T G (Z[n] - S[n]) X^T [n-1]. \quad (8)$$

Легко проверить, что размерность матрицы  $Q$  совпадает с размерностью матрицы  $A$ .

Для решения (3) в случае функционалов в векторных пространствах и используются алгоритмы СА. В качестве естественного обобщения этих процедур для нашего случая предлагается следующий итеративный алгоритм:

$$A[k] = A[k-1] - \gamma[k] Q(Z[n], A[k-1]), \quad (9)$$

или с учетом (8)

$$A[k] = A[k-1] - \gamma[k] 2\Delta t D^T G (Z[n, k-1] - S[n]) X^T [n-1],$$

где  $k$  — номер итерации, в общем случае не совпадающий с моментом  $n$  дискретного времени,  $\gamma[k]$  — скалярный коэффициент, задающий длину шагов.

Прежде чем перейти к следующему примеру, заметим, что лишь в простейшем случае  $\gamma[k]$  из (9) является скаляром. В общем случае, когда приращение каждого элемента матрицы  $A$  определяется всеми компонентами матрицы  $Q$ , матричный алгоритм СА имеет следующий вид:

$$a_{ij}[k] = a_{ij}[k-1] - \gamma_{ijlm}[k] q_{lm}[n, k-1]. \quad (10)$$

В (10) использована сокращенная форма тензорной записи;  $\gamma_{ijlm}[k]$  — тензор четвертой валентности, аналог матрицы шагов  $\Gamma[k]$  в векторных алгоритмах стохастической аппроксимации, например:

$$C[k] = C[k-1] - \Gamma[k] \nabla I[k-1].$$

Очевидно, только в форме (10) могут быть реализованы оптимальные алгоритмы матричной стохастической аппроксимации.

**Идентификация линейной нестационарной системы с помощью линейной обучающейся модели.** Рассмотрим для простоты случай свободного движения линейной нестационарной системы, многомерный выходной сигнал которой для  $m$  идущих подряд дискретных моментов вре-

мени регистрируется в виде следующей матрицы:

$$S = \begin{bmatrix} S^T [1] \\ S^T [2] \\ \vdots \\ S^T [m] \end{bmatrix}; \quad S^T [j] = \{s_i [j]\} = \{s_{ij}\}, \quad i = \overline{1, r}, \quad j = \overline{1, m}. \quad (11)$$

Для идентификации системы используем линейную нестационарную модель с наперед заданной размерностью пространства состояний ( $s$ ):

$$X[n] = F[n]X[n-1]; \quad Z[n] = DX[n], \quad (12)$$

причем  $F[n] = \sum_{p=1}^{p=\pi} F_p l_p [n]$ ,

где  $X[n] = \{x_i [n]\}$ ;  $F = \{F_p\} = \{f_{ijp}\}$  — линейное приближение фундаментального тензора системы, тензор третьей валентности ( $i, j = \overline{1, s}$ ), подлежащий оценке;  $\{l_p [n]\}$  — конечный набор произвольных линейно-независимых функций дискретного времени  $p = \overline{1, \pi}$ .

Аналогично (11)  $m$  идущих подряд значений вектора  $Z[n]$  объединим в матрицу

$$Z = \begin{bmatrix} z^T [1] \\ z^T [2] \\ \vdots \\ z^T [m] \end{bmatrix}, \quad (13)$$

строками которой являются транспонированные векторы  $Z[n]$ .

Определим, что система идентифицирована, если матрицы  $Z$  и  $S$  неразличимы, иначе — достигнут минимум следующего скалярного функционала:

$$I(F[n]) = M \langle S_p (Z - S)^T (Z - S) \rangle, \quad (14)$$

где математическое ожидание взято по ансамблю ошибок, сопровождающих регистрацию выхода системы.

Как и в предыдущем примере, найдем производную скалярной функции, стоящей в (14) под знаком математического ожидания, однако теперь речь идет о дифференцировании скалярной функции тензорной переменной третьей валентности. Если воспользоваться одной несколько искусственной конструкцией, то можно продолжить выкладки, оставаясь в рамках матричной символики.

Симметрическая матрица в (14) может быть записана следующим образом:

$$(Z - S)^T (Z - S) = \left( \sum_p L_p Z(-1) F_p^T - S \right)^T \left( \sum_p L_p Z(-1) F_p^T - S \right), \quad (15)$$

где

$$L_p = \begin{bmatrix} l_p [1] & 0 & \dots & 0 \\ 0 & l_p [2] & & \\ \vdots & \vdots & \ddots & \\ 0 & 0 & \dots & l_p [m] \end{bmatrix}; \quad Z(-1) = \begin{bmatrix} z^T [0] \\ z^T [1] \\ \vdots \\ z^T [m-1] \end{bmatrix}.$$

Производная функционала (14) по  $F$  (тензору третьей валентности) может быть представлена  $p$ -компонентами, производными по  $F_p$  (тензорам второй валентности), т. е.

$$\nabla F I = \{Q_p\}, \quad \text{где } Q_p = \nabla_{F_p} I; \quad Q_p = \{q_{ijp}\}, \quad i, j = \overline{1, s}.$$

В соответствии с (6) и учетом (15)

$$\nabla_{F_p} S_p (Z - S)^T (Z - S) = 2 (Z - S)^T L_p Z (-1),$$

откуда

$$\nabla F I = M \langle \{2 (Z - S)^T L_p Z (-1)\} \rangle. \quad (16)$$

Решение уравнения

$$\nabla F I = 0$$

с помощью метода СА приводит к следующему итеративному алгоритму относительно тензоров третьей валентности:

$$\hat{f}_{ijp}[k] = \hat{f}_{ijp}[k-1] - \gamma[k] (z_{ji}[k-1] - s_{ji}) l_{pj} z_{ij} (-1). \quad (17)$$

Не вдаваясь в подробности относительно существа процесса идентификации в соответствии с (17), отметим лишь, что аналогом необходимых условий идентифицируемости [4] такой системы будет

$$m \geq ps.$$

В общем случае в (17)  $\gamma[k]$  — тензор шестой валентности.

**Обучающаяся двухальтернативная система распознавания образов, описываемых матрицами с квадратичной решающей функцией.** Пусть реакция рецепторного поля на предъявляемый для обучения (и распознавания) образ есть матрица (не обязательно двоичных чисел

$$X = \{x_{ij}\}, \quad i = \overline{1, r}, \quad j = \overline{1, s},$$

причем ожидается наиболее эффективное распознавание при квадратичной решающей функции. Сконструируем ее для матричного описания образов по аналогии с квадратичными формами векторных пространств (запись с использованием тензорного соглашения):

$$\hat{y} = a + b_{ij} x_{ij} + c_{ijlm} x_{ij} x_{lm}. \quad (18)$$

Опуская все рассуждения о построении алгоритма обучения, которые здесь не имеют принципиального значения, отметим, что, например, при квадратичном функционале, определяющем близость «истинной» решающей функции  $y$  и ее аппроксимации  $\hat{y}$ , алгоритмы обучения каждого из объектов свертки (18) сводятся к следующим алгоритмам СА:

$$T[k] = T[k-1] - \gamma[k] [y[k] - \hat{y}(T[k-1], X[k])] \nabla_T \hat{y}(T[k-1], X[k]), \quad (19)$$

где  $T[k] = a[k]; \{b_{ij}[k]\}; \{c_{ijlm}[k]\}$  соответственно,  $X[k], y[k], k = 1, 2, \dots$  — обучающие выборки.

В соответствии с (7)

$$\nabla_a \hat{y} = \frac{\partial \hat{y}}{\partial a} = 1; \quad \nabla_b \hat{y} = \frac{\partial \hat{y}}{\partial b_{ij}} = x_{ij}; \quad \nabla_c \hat{y} = \frac{\partial \hat{y}}{\partial c_{ijklm}} = x_{ij} x_{lm}.$$

В (19) максимальная валентность тензорной последовательности равна 4.

Итак, из рассмотренных достаточно нетривиальных примеров вытекает естественность обобщения алгоритмов СА на тензорные, рекуррентные последовательности. Следует ожидать, что состоятельная оценка (состоятельное обучение) тензорных неизвестных параметров возможна лишь при использовании соответствующего по рангу тензорного алгоритма СА. Остановимся на этом утверждении более подробно.

Пусть минимизируемый функционал зависит от неизвестной матрицы  $A = \{a_{ij}\}$ , но оценка (обучение) всех или части элементов ее проводится с помощью векторного алгоритма СА:

$$\vec{C}[k] = \vec{C}[k-1] - \gamma_k \nabla_c I(A, \vec{C}[k-1]); \quad (20)$$

здесь  $\vec{C} = \{C_i\}$  — вектор неизвестных параметров, который всегда может быть описан путем следующего линейного преобразования матрицы  $A$  (тензорное умножение со сверткой по одной паре индексов):

$$c_i = r_{ij} a_{ij}, \quad (21)$$

где  $r_{ij}$  — тензор третьей валентности.

Так как минимизируемый функционал задан в виде функции  $A$ , то

$$\nabla_A I(C(A)) = \{q_{ij}\} = \nabla_c I \nabla_A C = \left\{ \frac{\partial I}{\partial c_i} \right\} \left\{ \frac{\partial c_i}{\partial a_{ij}} \right\}. \quad (22)$$

Из (21) следует, что

$$\frac{\partial c_i}{\partial a_{ij}} = r_{ij}.$$

Для того чтобы реализовать (20), необходимо существование величины, обратной  $\nabla_A C$ , так как

$$\nabla_c I = \nabla_A I (\nabla_A C)^{-1}. \quad (23)$$

Если существует  $\nabla_A I$ , то можно реализовать матричный алгоритм СА, хотя в общем случае это не означает, что он может быть сведен к векторному алгоритму СА. По этой же самой причине векторный алгоритм СА никогда не может быть сведен к скалярному алгоритму, так как при этом аналогом (21) является

$$c = r_{ij} a_{ij}$$

или в векторно-матричных обозначениях  $c = R^T A$ , где  $c$  — скаляр, а  $R$  и  $A$  — векторы, и для существования алгоритма необходимо существование  $R^{-1}$ .

В самом общем виде тензорный алгоритм СА (аналог процедуры Кифера-Вольфовица) имеет вид

$$a_{ij\dots p}[k] = a_{ij\dots p}[k-1] - \gamma_{ij\dots p, i'j'\dots p'}[k] \frac{\partial I(a_{ij\dots p}[k-1])}{\partial a_{i'j'\dots p'}}. \quad (24)$$

По одинаковым индексам в (24) производится свертка. Валентность тензора шагов в общем случае равна удвоенному рангу алгоритма.

Не останавливаясь на условиях сходимости тензорных алгоритмов СА, можно считать, что при исследовании сходимости полезно использовать линейное преобразование (21), сводящее любой тензор к вектору.

В заключение приведем численный пример, иллюстрирующий эффективность матричного алгоритма СА.

Свободное движение линейной системы и ее модели описывается соответственно уравнениями

$$\begin{aligned}\dot{X}(t) &= A^* X(t); & X|_{t=0} &= X^0; \\ \dot{Y}(t) &= AY(t); & Y|_{t=0} &= Y^0.\end{aligned}$$

Перейдем к разностным уравнениям

$$\begin{aligned}\dot{X}[n+1] &= (I + A^*h)X[n]; & X[0] &= X^0; \\ Y[n+1] &= (I + Ah)Y[n]; & Y[0] &= Y^0.\end{aligned}$$

Необходимо обучить модель таким образом, чтобы в каждый момент времени

$$I[n] = \|Y[n] - X[n]\| = (Y[n] - X[n])^T (Y[n] - X[n]) = I_{\min}.$$

В соответствии с (24) алгоритм адаптации модели имеет вид

$$A[k, n] = A[k-1, n] - \gamma[k] 2h (Y[n, k-1] - X[n]) Y^T[n-1]. \quad (25)$$

При адаптации без помех нетрудно найти условие, которому должны удовлетворять значения  $\gamma[k]$  с тем, чтобы

$$I[n, k] < I[n, k-1],$$

т. е. процесс адаптации сходился.

Для этого необходимо, чтобы

$$\gamma[k] \leq \frac{1}{2h^2 \|Y[n-1]\|}, \quad (26)$$

причем знак равенства соответствует полной адаптации на данном шаге, т. е.  $I[k+1, n] = 0$ , что достижимо только при измерении состояний системы без помех.

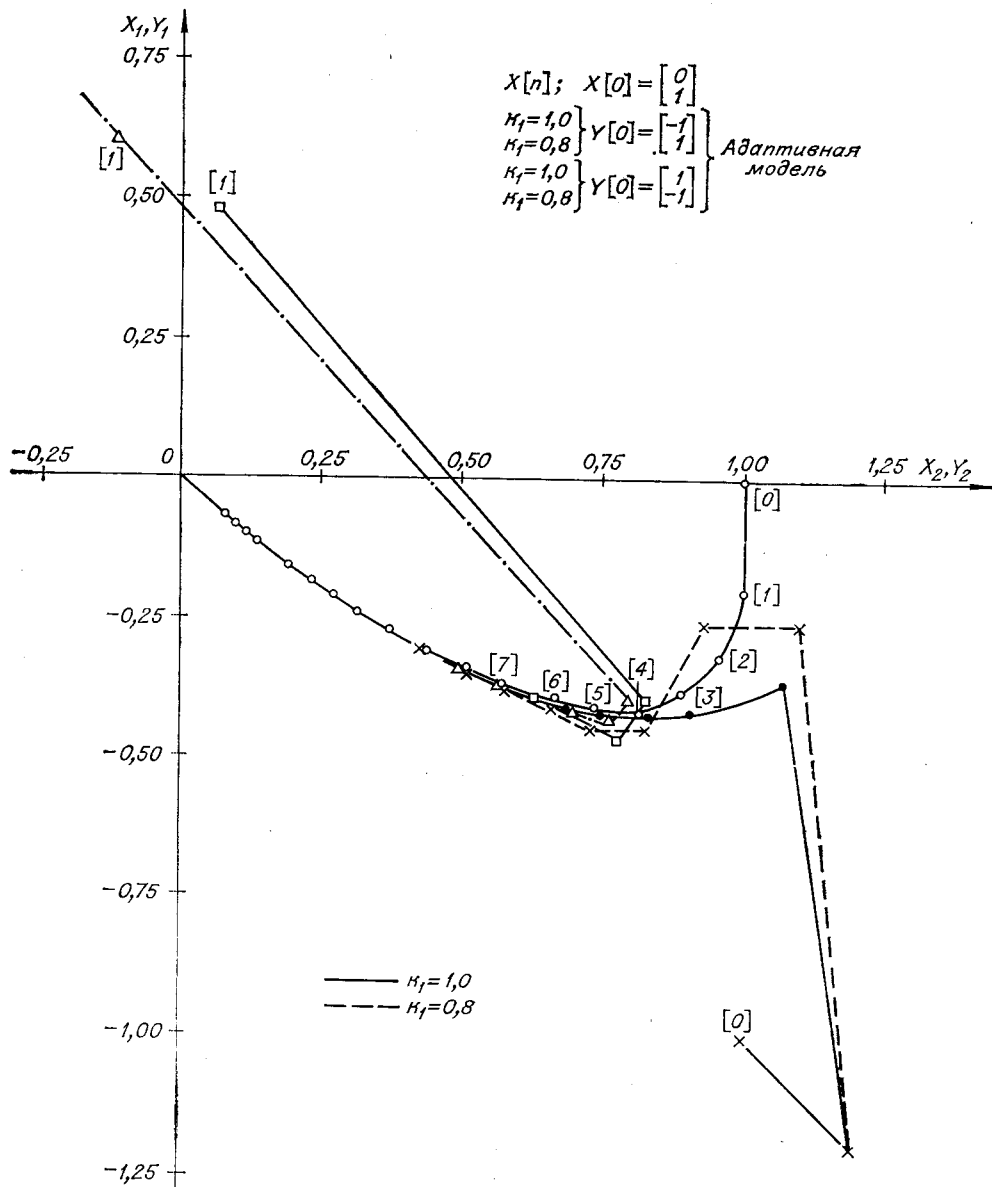
Условие (25) остается в силе и при регистрации выхода системы с некоррелированной помехой.

На рисунке, где

$$\begin{aligned}A^* &= \begin{bmatrix} -2 & 1 \\ 1 & 0 \end{bmatrix}; & A[0] &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; & X[0] &= \begin{bmatrix} 0 \\ 1 \end{bmatrix}; \\ h &= 0,2; & \gamma[k] &= \frac{k_1}{2h^2 \|Y[n-1]\|},\end{aligned}$$

приведен процесс адаптации линейной модели из двух различных начальных точек  $Y[0] = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$  и  $Y[0] = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ .

Для каждого  $n$  выполняется три итерации алгоритма (25).



### ЛИТЕРАТУРА

1. Н. Robbins, S. Monro. A Stochastic Approximation Method.— Ann. Math. Stat., 1951, v. 22, N 1.
2. I. R. Blum. Multidimensional Stochastic Approximation Methods.— Ann. Math. Stat., 1954, v. 25, N 4.
3. Я. З. Цыпкин. Основы теории обучающихся систем. М., «Наука», 1970.
4. Р. Ли. Оптимальные оценки, определение характеристик и управление. М., «Наука», 1966.
5. А. А. Амосов, В. В. Колпаков. Скалярно-матричное дифференцирование и его приложения к конструктивным задачам теории связи.— Проблемы передачи информации, 1972, т. 8, № 1.

*Поступила в редакцию 25 июня 1973 г.*