АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ В МИКРОЭЛЕКТРОНИКЕ.
ТЕОРИЯ, МЕТОДЫ, АЛГОРИТМЫ
(Вычислительные системы)
1978 год Выпуск 77

УДК 519.95

О СЛУЧАЙНОМ ПОИСКЕ ОПТИМАЛЬНОЙ СИСТЕМЫ ПРИЗНАКОВ

В.Г.Устюжанинов

Задача нахождения достаточно информативной системы признаков широко известна в распознавании образов. Существуют различные методы ее решения. Нас будут интересовать методы,в основе которых лежит случайный поиск [1-6], точнее,вопросы трудоемкости и эффективности этих методов. Можно считать детерминированные алгоритмы частным случаем алгоритмов случайного поиска. Конструкции таких алгоритмов, решающих задачу поиска оптимальной системы признаков, предлагались в работах [1-3,5,6]. Их трудоемкость исследовалась в работе [5]. Не все так благополучно обстоит с недетерминированными алгоритмами. Возможности недетерминированных алгоритмов до сих пор не изучались. Главным препятствием здесь явилось отсутствие метода оценки эффективности произвольного алгоритма случайного поиска. Причем это относится не только к поиску оптимальной системы признаков,но и ко всем задачам,для решения которых используются алгоритмы случайного поиска. Не составляет исключения и глобальный поиск, наиболее полно изучавшийся в работе [7], хотя и там рассматривается не весь класс алгоритмов случайного поиска.

Настоящая статья состоит из двух параграфов. В первом параграфе в общем случае излагается метод оценки эффективности лучшего из всех алгоритмов случайного поиска. Во втором параграфе этот метод используется для оценки эффективности алгоритмов, ищущих оптимальную систему признаков.

§1. Эффективность алгоритмов случайного поиска

Пусть F – класс функций f, определенных на конечном множестве E и принимающих значения из конечного множества L. В дальнейшем элементы множества E будут именоваться точками. Часто требуется разыскать в E точку, обладающую некоторым свойством C. Например, точку глобального максимума f, или точку, в которой значение f превосходит некоторое число. Работу многих алгоритмов случайного поиска, решающих подобные задачи, можно представить следующей неформальной схемой. Алгоритм работает в два этапа. Целью первого этапа является получение информации о функции f. Он состоит из конечного числа тактов. На каждом такте случайным образом, в соответствии с функцией вероятности $p(x)$, заданной на E, выбирается точка $x \in E$ и затем производится измерение f в выбранной точке. Второй этап сводится к выбору ответа. Он состоит из одного такта и заключается в выборе точки $x \in E$, которая объявляется удовлетворяющей свойству C. (При этом алгоритм может ошибиться.) Способ выбора точки $x \in E$ на втором этапе тот же, что и на первом. Предполагается, что функция вероятности $p(x)$, используемая на некотором такте работы алгоритма, однозначно определяется выбором точек измерения функции f и набором исходов этих измерений на всех предыдущих тактах. Никаких ограничений на вид этой зависимости накладывать не будем.

Построим формальную модель алгоритма случайного поиска. С этой целью введем некоторые дополнительные определения. Процесс измерения функции f в некоторой точке x назовем операцией измерения. Две операции измерения будем считать различными, если им соответствуют измерения функции f в разных точках E. Выбор ответа на втором этапе работы алгоритма назовем операцией выбора ответа. Процесс выбора точки для измерения функции f на тактах первого этапа назовем операцией выбора операции (в дальнейшем просто операцией выбора), так как при этом фактически происходит выбор одной из операций измерения. Допустим, сверх того, возможность выбора с ее помощью операции выбора ответа.

При работе алгоритма случайного поиска возможны различные последовательности выполнения перечисленных выше операций. Задание всех допустимых таких последовательностей можно осуществить фиксацией операционной схемы алгоритма. Она представляет собой конечное дерево, вершины которого ассоциируются с отдельными операциями, а цепи – с допустимыми последовательностями операций. В дальнейшем не будем различать вершины этого дерева и соответствующие им операции. Первой выполняется корневая операция, находящаяся на нулевом ярусе. Порядок выполнения остальных операций определяется при прохождении цепи дерева от корневой вершины к концевой. Опишем операционную схему алгоритма случайного поиска.

Корневой вершиной является операция выбора. На ярусах с нечетными номерами располагаются операции выбора ответа и операции измерения. На ярусах с четными номерами (за исключением предпоследнего яруса) располагаются операции выбора и концевые вершины. На предпоследнем ярусе располагаются операции выбора ответа.

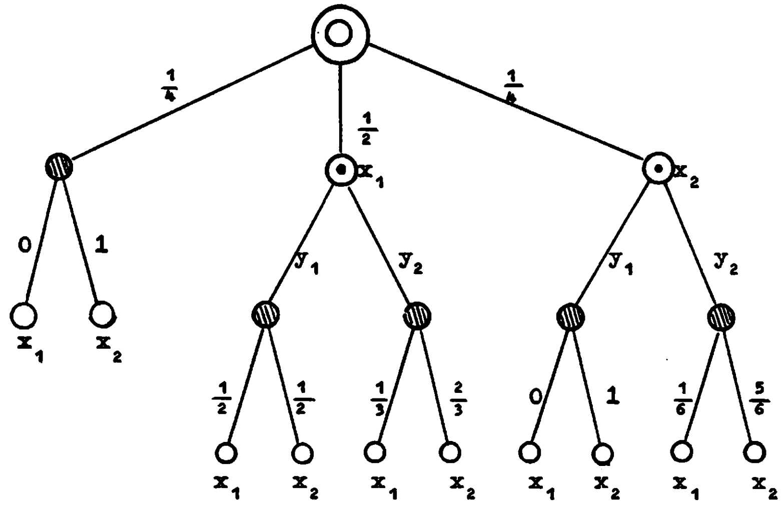
Из каждой операции выбора ξ исходит $\|E\|+1$ ребро. Одно из ребер входит в операцию выбора ответа, а $\|E\|$ других – в попарно различные операции измерения. Каждому ребру v соответствует вероятность $p(v)$. Если $V(\xi)$ – множество исходящих из ξ ребер, то потребуем, чтобы $\sum_{v\in V(\xi)} p(v) = 1$. Выбор операции, в которую входит ребро v, осуществляется с вероятностью $p(v)$. Следующей за операцией выбора выполняется выбранная ею операция.

Пусть L есть область значений функции f. Из каждой операции измерения исходит $\|L\|$ ребер, которые могут входить в операции выбора и в операции выбора ответа. Ребра метятся элементами множества L. Метки различных ребер различны. Сама операция измерения метится точкой $x \in E$, в которой согласно этой операции производится измерение функции f. Пусть в результате измерения установлено, что $f(x) = y$. Тогда следующей за операцией измерения выполняется та операция, в которую входит ребро, помеченное меткой y.

Из каждой операции выбора ответа ξ исходит $\|E\|$ ребер, входящих в концевые вершины дерева. Каждому исходящему ребру v соответствует вероятность $p(v)$ при выполнении равенства $\sum_{v\in V(\xi)} p(v) = 1$.

Концевые вершины метятся точками из E. Метки концевых вершин, подчиненных одной операции выбора ответа, попарно различны. В процессе выполнения операции выбора ответа осуществляется случайный выбор одной из концевых вершин и соответствующая ей метка объявляется результатом работы алгоритма. Выбор концевой вершины, в которую входит ребро v, осуществляется с вероятностью $p(v)$. После выполнения операции выбора ответа алгоритм прекращает работу.

ПРИМЕР. На рисунке изображена операционная схема алгоритма, производящего не более одного измерения функции. Имеем множества



$L=\{y_1,y_2\}$ и $E=\{x_1,x_2\}$. Вершина ◎ соответствует операции выбора, вершины ⊙ – операциям измерения, вершины ● – операциям выбора ответа и ○ – концевые вершины.

Пусть R есть множество операций выбора и операций выбора ответа в операционной схеме. Варьируя функции вероятности $p(v)$, заданные на ребрах, исходящих из вершин $\xi \in R$, можно получать произвольные алгоритмы случайного поиска.

В частности, когда все эти распределения вырождены, получаем детерминированный алгоритм. В противном случае получаем алгоритмы, которые будем называть вероятностными. Будем считать различными алгоритмы, отличающиеся функциями вероятности, заданными на ребрах, исходящих из вершин $\xi \in R$. Поэтому в дальнейшем для этих функций вероятности будет использоваться обозначение $p(v/a)$, где a – обозначение алгоритма.

Пусть δ есть произвольная цепь, соединяющая в операционной схеме корневую вершину с концевой. Проходя ее от корневой вершины к концевой, составим упорядоченный список $x_1y_1x_2y_2\ldots x_ty_t$ меток операций измерения и меток исходящих из них ребер, принадлежащих этой цепи. Будем считать, что функция f допускается цепью δ, если для всех $i = 1,\ldots,t$ выполняется равенство $f(x_i) = y_i$ }. Пусть $v_1,\ldots,v_l$ – список всех ребер, лежащих на δ и исходящих из операций множества R. Будем считать, что в результате работы алгоритма a над функцией f имело место событие δ, если в процессе работы алгоритма были выполнены все операции, принадлежащие δ. В соответствии с этим вероятность этого события положим равной $p(\delta/a) = \prod_{i=1}^{l} p(v_i/a)$, если $l \geq 1$, и $p(\delta/a) = 1$, если $l=0$. Будем считать, что в результате работы алгоритма a над функцией f имело место событие x, если в качестве ответа была названа точка x. Обозначим через $\Delta(f)$ множество всех цепей, допускающих функцию f. Так как при работе алгоритма над функцией f проходится лишь одна из цепей множества $\Delta(f)$, то при различных цепях $\delta, \delta' \in \Delta(f)$ события δ и δ' несовместимы. Поэтому вероятность $p(x/f,a)$ того, что, работая над функцией f алгоритм a в качестве ответа назовет точку x, равна

$$p(x/f, a) = \sum_{\delta \in \Delta(f) \cap \Delta(x)} p(\delta/a). \qquad (1)$$

Здесь $\Delta(x)$ - множество цепей, оканчивающихся концевыми вершинами с меткой x. Легко проверить, что в каждом детерминированном алгоритме a_D существует единственная цепь δ, допускающая функцию f, вероятность которой $p(\delta/a_D)$ равна I. Откуда для алгоритма a_D вероятность $p(x/f, a_D) = 1$, если эта цепь заканчивается концевой вершиной, помеченной x, и $p(x/f,a_D) = 0$ - в противном случае.

Обозначим через A_D множество всех детерминированных алгоритмов, которые можно получить, используя описанную выше операционную схему. Очевидно, что это множество конечно.

ТЕОРЕМА I. *Для всякого алгоритма a существует такая функция вероятности $p_a(a_D)$, определенная на множестве A_D и зависящая от a, что для любых $x \in E$ и $f \in F$ выполняется равенство*

$$p(x/f,a) = \sum_{a_D \in A_D} p_a(a_D) \cdot p(x/f, a_D). \qquad (2)$$

ДОКАЗАТЕЛЬСТВО. Построим по алгоритму a функцию вероятности $p_a(a_D)$. Пусть R - множество операций выбора и операций выбора ответа и a_D - произвольный детерминированный алгоритм. Обозначим через $V(a_D)$ множество ребер v, исходящих из операций множества R, на которых в алгоритме a_D вероятность $p(v/a_D) = 1$. Алгоритму a_D сопоставим число $p_a(a_D)$, определяемое формулой

$$p_a(a_D) = \prod_{v \in V(a_D)} p(v/a) , \tag{3}$$

где $p(v/a)$ есть вероятность,соответствующая ребру v в алгоритме a. Теорема будет доказана, если будет показано, что для чисел $p_a(a_D)$ справедливы равенства (2) и

$$\sum_{a_D \in A_D} p_a(a_D) = 1 .$$

Пусть δ – цепь, соединяющая корневую вершину с некоторой (не обязательно концевой) вершиной; $\zeta = \{\xi_1,\dots,\xi_e\}$ – список операций из множества R, принадлежащих этой цепи; $\gamma = \{v_1,\dots,v_e\}$ – список ребер, исходящих соответственно из $\xi_1,\dots,\xi_e$ и также принадлежащих цепи δ. Пусть $R \setminus \zeta = \{\eta_1,\dots,\eta_m\}$ и $V(\eta_i)$ есть множество ребер, исходящих из вершины η_i $(i=1,\dots,m)$. Образуем множество детерминированных алгоритмов $A_D(\delta)$, у которых вероятности на ребрах $v_1,\dots,v_e$, лежащих на цепи δ, равны 1. Очевидно, каждый алгоритм $a_D \in A_D(\delta)$ может быть получен выбором по одному ребру w_i из множеств $V(\eta_i)$ и приданием этим ребрам вероятностей $p(w_i/a_D)$, равных 1. Опираясь на это замечание и равенство (3), получим в случае $\gamma \neq \emptyset$:

$$\sum_{a_D \in A_D(\delta)} p_a(a_D) = \sum_{a_D \in A_D(\delta)} \prod_{v \in V(a_D)} p(v/a) = \{ \prod_{i=1}^{e} p(v_i/a)\} \times$$

$$\times \{ \sum_{w_1 \in V(\eta_1)} \dots \sum_{w_m \in V(\eta_m)} \prod_{j=1}^{m} p(w_j/a)\} =$$

$$= \{ \prod_{i=1}^{e} p(v_i/a)\} \cdot \{ \prod_{j=1}^{m} \{ \sum_{w_j \in V(\eta_j)} p(w_j/a)\}\} =$$

$$= \prod_{i=1}^{e} p(v_i/a) = p(\delta/a). \tag{4}$$

В случае $\gamma = \emptyset$ аналогично показывается, что $\sum_{a_D \in A_D(\delta)} p_a(a_D) = 1$. Если $\gamma = \emptyset$, то множество $A_D(\delta) = A_D$. Поэтому

$$\sum_{a_D \in A_D} p_a(a_D) = 1.$$

Таким образом, набор чисел $p_a(a_D)$ действительно представляет набор значений функции вероятности, определенной на A_D .

Пусть δ есть цепь, соединяющая корневую вершину с концевой. Тогда выполняется равенство

$$\bigcup_{\delta \in \Delta(f)} A_D(\delta) = A_D.$$

Действительно, его нарушение повлекло бы существование детерминированного алгоритма, не имеющего ни одной цепи δ , допускающей функцию f. А этого не может быть. Если разные цепи δ_1 и δ_2 допускают одну и ту же функцию f, то выполняется равенство $A_D(\delta_1) \cap A_D(\delta_2) = \emptyset$. Нарушение этого равенства повлекло бы существование детерминированного алгоритма, у которого две разные цепи δ_1 и δ_2 допускают одну и ту же функцию f . Этого также не может быть. Используя сделанные замечания для изменения порядка суммирования, из (I), (4) получаем

$$p(x/f,a) = \sum_{\delta \in \Delta(f) \cap \Delta(x)} p(\delta/a) = \sum_{\delta \in \Delta(f) \cap \Delta(x)} \; \sum_{a_D \in A_D(\delta)} p_a(a_D) =$$

$$= \sum_{\delta \in \Delta(f)} \; \sum_{a_D \in A_D(\delta)} p_a(a_D) p(x/f,a_D) = \sum_{a_D \in A_D} p_a(a_D) p(x/f,a_D).$$

Теорема доказана.

Применим доказанную теорему к оценке эффективности алгоритмов случайного поиска. Будем представлять процесс эксплуатации алгоритма a в виде конечной последовательности из q независимых испытаний. В каждом испытании случайным образом в соответствии с функцией вероятности $p(f)$, определенной на F , выбирается некоторая функция f , которая затем исследуется алгоритмом a . Пусть s – число испытаний, в которых алгоритм правильно указал точку множества E , обладающую свойством C . В качестве меры эффективности работы алгоритма a при фиксированной функции вероятности $p(f)$ примем вероятность правильного ответа $\Phi(p(f),a)$, к которой по вероятности при $q \to \infty$ стремится частота s/q :

$$\Phi\,(p(f),a) = \sum_{f \in F} p(f)\; p(x \in M(f)/f,a). \qquad (5)$$

Здесь $M(f)$ – множество точек из E , обладающих свойством C, и $p(x \in M(f)/f,a)$ – вероятность того, что, работая над функцией f , алгоритм a выберет в качестве ответа точку $x \in M(f)$. Примером $M(f)$ может служить множество точек из E , в которых функция f достигает глобального максимума. Ранее говорилось, что выдача алгоритмом точки x в качестве ответа рассматривается как наступление события x. Так как алгоритм не может в ответе указать одновременно две разные точки x и x', то события x и x' несовместимы. По этой причине

$$p(x\in M(f)/f,a) = \sum_{x\in M(f)} p(x/f,a) .$$

Пусть A есть множество всех алгоритмов, которые можно получить, используя описанную выше операционную схему, и A_D – подмножество детерминированных алгоритмов, входящих в A. Пусть также F_0 есть произвольный подкласс класса F. Будем считать, что алгоритм a_D правильно сработал на функции f, если в качестве ответа он выбрал точку $x\in M(f)$. Обозначим через $F_0(a_D)$ множество функций $f\in F_0$, на которых правильно срабатывает алгоритм a_D.

ЛЕММА. Существует такая функция вероятности $p_0(f)$, что, каков бы ни был алгоритм $a\in A$, выполняется неравенство

$$\Phi(p_0(f),a)\le \max_{a_D\in A_D} \frac{\|F_0(a_D)\|}{\|F_0\|} .$$

ДОКАЗАТЕЛЬСТВО. Построим функцию $p_0(f)$. Положим $p_0(f)=1/\|F_0\|$, если $f\in F_0$, и $p_0(f)=0$, если $f\notin F_0$. Пусть a есть произвольный алгоритм из множества A. Из равенств (2), (5) и определения вероятности $p(x\in M(f)/f,a)$ следует, что

$$\Phi(p_0(f),a) = \sum_{f\in F} p_0(f) \sum_{x\in M(f)} \sum_{a_D\in A_D} p_a(a_D)\, p(x/f,a_D) .$$

Учитывая, что функция вероятности $p_a(a_D)$ едина для всех $f\in F$ и $x\in E$, последнее выражение можно переписать в виде

$$\Phi(p_0(f),a) = \sum_{a_D\in A_D} p_a(a_D)\{\sum_{f\in F} p_0(f) \sum_{x\in M(f)} p(x/f,a_D)\} =$$

$$= \sum_{a_D\in A_D} p_a(a_D)\, \frac{\|F_0(a_D)\|}{\|F_0\|} \le \max_{a_D\in A_D} \frac{\|F_0(a_D)\|}{\|F_0\|} .$$

Лемма доказана.

§2. О поиске оптимальной системы признаков

Пусть объекты описываются набором из n признаков и E есть множество подмножеств этого набора. Меру качества систем признаков $\gamma \in E$ зададим как числовую функцию f из некоторого класса, принимающую значения из конечного множества L. Обозначим число признаков в системе γ значком $\|\gamma\|$. Ограничимся рассмотрением трех классов функций: классом монотонных функций $F_М$; классом цепных унимодальных функций $F_У$; классом цепных многоэкстремальных функций $F_{МЭ}$. Последовательность систем признаков $\gamma_1,\dots,\gamma_n$ назовем цепью, если для всех $i=1,\dots,n-1$ верно $\|\gamma_i\|=i$ и $\gamma_i \subset \gamma_{i+1}$. Функцию f назовем монотонной, если на произвольной цепи $\gamma_1,\dots,\gamma_n$ для всех $i=1,\dots,n-1$ имеем место $f(\gamma_i) \le f(\gamma_{i+1})$. Функцию f назовем цепной унимодальной, если на произвольной цепи $\gamma_1,\dots,\gamma_n$ найдется такая система признаков γ_r, что для всех $i<r$ выполняется неравенство $f(\gamma_{i-1}) < f(\gamma_i)$, а для всех $i \ge r$ – неравенство $f(\gamma_i) > f(\gamma_{i+1})$. Функцию f назовем цепной многоэкстремальной при условии, что на произвольной цепи $\gamma_1,\dots,\gamma_n$ она удовлетворяет следующим требованиям: если γ_i и γ_{i+j} есть соседние минимумы f на этой цепи, то на куске цепи $\gamma_i,\dots,\gamma_{i+j}$ она унимодальна; если γ_i и γ_{i+j+1} есть соседние максимум и минимум (либо наоборот) функции f на этой цепи, то $j \ge d$. Нас будут интересовать следующие задачи.

Пусть $f \in F_М$. Требуется указать систему признаков γ^* минимальной мощности, для которой $f(\gamma^*) \ge \max_{\gamma \in E} f(\gamma) - k$, где $k \ge 0$ и $k=\mathrm{const}$.

Пусть $f \in F_У$ или $f \in F_{МЭ}$. Требуется указать систему признаков γ^*, для которой $f(\gamma^*) = \max_{\gamma \in E} f(\gamma)$. Объединим в класс A_m все алгоритмы случайного поиска, производящие не более чем m операций измерения.

ТЕОРЕМА 2. Существует такая функция вероятности $p_0(f)$, заданная на классе F, что для любого алгоритма $a \in A_m$ выполняются следующие неравенства:

если $F=F_М$, то $\Phi(p_0(f),a) \le (m+1)/C_n^{[n/2]}$;

если $F=F_У$, то $\Phi(p_0(f),a) \le (m+1)/\sum_{i=-1}^{1} C_n^{[n/2]+i}$;

если $F=F_{МЭ}$, то при $d=0$ $\Phi(p_0(f),a) \le (m+1)/2^n$

и при $0<d\le n$ $\Phi(p_0(f),a)\le (m+1)/\sum_{j=-r}^{r}\sum_{i=-1}^{1} C_n^{q(j)+i}$, где $r = \left]\frac{[n/2]}{2(d+2)}\right[$ и $q(j) = [n/2] + 2j(d+2)$.

ДОКАЗАТЕЛЬСТВО. Ограничимся рассмотрением лишь случая монотонных функций. Доказательство оценок для других типов функций проводится аналогично. Построим вспомогательную функцию f_0. Положим $f_0(\gamma) = k+1$, если число признаков $\|\gamma\|\ge [n/2]+1$, и $f_0(\gamma)=0$ – в противном случае. Образуем класс функций F_0 из всех попарно различных функций f, отличающихся от f_0 на единственной системе признаков γ^* с числом признаков $\|\gamma^*\| = [n/2]$ и равных на γ^* числу $k+1$. Всего таких функций можно образовать $C_n^{[n/2]}$. Покажем, что произвольный детерминированный алгоритм $a_D\in A_D$ может правильно указать систему признаков γ^* не более чем у $m+1$ функции из класса F_0. Пусть δ есть цепь, допускающая в алгоритме a_D функцию f_0, и $x_1y_1x_2y_2\ldots x_ty_t$ есть упорядоченный список меток операций измерения и меток исходящих из них ребер, принадлежащих этой цепи. Причем $t\le m$ и для всех $i=1,\ldots,t$ выполняется равенство $f_0(x_i) = y_i$. Легко видеть, что по крайней мере для $C_n^{[n/2]}-t$ функций f из класса F_0 равенство $f(x_i)=y_i$ также будет выполняться при всех $i=1,\ldots,t$. Объединим эти функции в класс F'. Таким образом, при работе алгоритма a_D над любой функцией $f\in F'$ будет проходиться одна и та же цепь δ и в качестве ответа выдаваться метка ее концевой вершины. При этом неизбежно для $C_n^{[n/2]}-t-1$ функций из F' системы признаков γ^* будут указаны неверно. Откуда вытекает, что алгоритм a_D может правильно назвать системы признаков γ^* не более чем для $t+1\le m+1$ функций из класса F_0. Далее остается лишь воспользоваться вышедоказанной леммой.

Теорема доказана.

Литература

1. Вопросы статистической теории распознавания. М., "Сов.радио", 1967, с.127-131.

2. ВАСИЛЬЕВ В.И. Распознающие системы. Киев, "Наукова думка", 1969, с. 39-42.

3. ТУРБОВИЧ И.Т., ГИТИС В.Г., МАСЛОВ В.К. Опознание образов. М., "Наука", 1971, с.109-111.

4. ЛБОВ Г.С. Выбор эффективной системы зависимых признаков. -В кн.: Вычислительные системы. Вып. 19. Новосибирск, 1965, с.21-34.

5. КАТЕРИНОЧКИНА Н.Н. Поиск максимального верхнего нуля монотонной функции алгебры логики. -"Докл. АН СССР", 1975, т.224,№3, с. 557-560.

6. КУЛЬЯНОВ Е.Г. Алгоритмы поиска максимального верхнего нуля произвольной монотонной функции логики. -"Журн. вычислит. математики и мат.физики", 1975, т.15, № 4, с.1080-1082.

7. СУХАРЕВ А.Г. Об оптимальных стратегиях поиска экстремума. -"Журн.вычислит.математики и мат. физики", 1971, т.11, №4, с.910-925.

Поступила в ред.-изд.отд.
24 февраля 1978 года