10 коп.

Редактор Б.Б.Колесова. Макет Н.А.Киселевой.

Подписано в печать 20.11.85.
Формат 60х90/16. Офсетная печать. Уч.-изд.листов 0,69.
Тираж 395. Заказ 36994.
Издательский отдел Объединенного института ядерных исследований.
Дубна Московской области.

объединенный
институт
ядерных
исследований
дубна

Р5-85-724

В.А.Сенченко

ПРЕДЕЛЬНО БЫСТРЫЙ АЛГОРИТМ ПОСТРОЕНИЯ ОРТОНОРМИРОВАННОЙ СИСТЕМЫ ПОЛИНОМОВ ОТ МНОГИХ ПЕРЕМЕННЫХ НА ПРОИЗВОЛЬНОМ ДИСКРЕТНОМ МНОЖЕСТВЕ

Направлено в журнал "Вычислительная математика и кибернетика"

1985

Введение

В практике численных расчетов широко используются системы ортонормированных полиномов, удовлетворяющих условию

$$\sum_{i=1}^{N} w_i \cdot e_k(\vec{x}_i)\ e_\ell(\vec{x}_i) = \delta_{k\ell}\ .$$

По таким системам существует обширная литература, значительное внимание уделяется вопросам аппроксимации экспериментальных данных. Среди наиболее важных свойств ортонормированных полиномов при их использовании для аппроксимации экспериментальных данных отмечают простоту статистического анализа при определении длины аппроксимирующего ряда, а также отсутствие необходимости решения системы нормальных линейных уравнений, которая, в общем случае, является плохо обусловленной.

Известны метод рекуррентных соотношений Форсайта (одномерный случай) и Вайсфелда (многомерный случай), а также метод ортогонализации обычных алгебраических одночленов Грамма–Шмидта для построения систем ортонормированных многочленов. Известные алгоритмы программной реализации этих методов имеют соответственно сложность $O(N \cdot M^{3-h})$ /II-I6/ и $O(N \cdot M^2) + O(M^3)$ /5,6/.

В настоящей работе сделана попытка исследовать вопрос, насколько быстрым в принципе может быть подобный алгоритм.

I. Определения, обозначения, краткий обзор известных результатов

На дискретном множестве точек

$$\Omega = \left\{ \vec{x}^{(k)} \in R^n \;\middle|\; k = 1, 2, \ldots, N \right\}$$

будем рассматривать систему алгебраических одночленов

$$F = \left\{ f_i \;\middle|\; f_i = f_i(\vec{x}) = x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}; \quad i = 1, 2, \ldots, M \right\}.$$

Следуя /8/, определим отношение порядка на F : f_i предшествует f_j тогда и только тогда, когда $q(\vec{i}) < q(\vec{j})$ или $q(\vec{i}) = q(\vec{j})$ и $\exists\, k \le n$, что $i_k + \ldots + i_n < j_k + \ldots + j_n$, где $q(\vec{i}) = i_1 + i_2 + \ldots + i_n$.

Порядковый номер i одночлена $f_i = x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}$ можно вычислить с помощью функции нумерации /4,I5/:

$$i = N_n(i_1,i_2,\ldots,i_n) = B^n_{q-1} + N_{n-1}(i_2,i_3,\ldots,i_n) \text{ , где}$$

$$B^n_q = \frac{(n+q)!}{n!\ q!}$$

Скалярное произведение двух произвольных функций $g_i(\vec{x})$ и $g_j(\vec{x})$ на Ω определим как

$$(g_i,g_j) = \sum_{k=1}^{N} w(\vec{x}^{(k)})\cdot g_i(\vec{x}^{(k)})\cdot g_j(\vec{x}^{(k)}) \text{ , где}$$

$w(\vec{x})$ - весовая функция (в данном случае - веса точек).

Функции e_i и e_j ортонормальны на Ω , если $(e_i,e_j)=\delta_{ij}$, где δ_{ij} - символ Кронекера.

Метод ортогонализации Грамма-Шмидта позволяет из произвольной линейно-независимой системы функций-векторов $\{f_i\}$ получить ортонормированную систему функций-векторов $\{e_i\}$ по следующим формулам:

$$e_1 = \frac{f_1}{\sqrt{(f_1,f_1)}} = a_{11}\cdot f_1 \qquad (I)$$

$$e_2 = \frac{f_2 - e_1\cdot(f_2,e_1)}{\sqrt{(f_2,f_2)-(f_2,e_1)^2}} = a_{21}\cdot f_1 + a_{22}\cdot f_2 \qquad (2)$$

. . . .

$$e_k = \frac{f_k - e_{k-1}\cdot(f_k,e_{k-1}) - \ldots - e_1\cdot(f_k,e_1)}{\sqrt{(f_k,f_k)-(f_k,e_{k-1})^2 - \ldots - (f_k,e_1)^2}} = a_{k1} f_1 + \ldots + a_{kk} f_k \qquad (3)$$

Используя $\forall i<k$

$$(f_k,e_i) = b_{i1}\cdot(f_k,f_1) + b_{i2}\cdot(f_k,f_2) + \ldots + b_{i,k-1}\cdot(f_k,f_{k-1}) \qquad (4)$$

можно вычислить элементы $\{a_{ij}\}$ матрицы

$$A = \begin{pmatrix} a_{11} & 0 & 0 & \ldots & 0 \\ a_{21} & a_{22} & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ a_{M1} & a_{M2} & a_{M3} & \ldots & a_{MM} \end{pmatrix},$$

связывающей системы $\vec{E} = col(e_1,e_2,\ldots,e_M)$ и $\vec{F} = col(f_1,f_2,\ldots,f_M)$

$$\vec{E} = A\cdot\vec{F}.$$

Как следует из (I)-(4), для вычисления $\{a_{ij}\}$ требуется иметь значения $\{(f_i,f_j) \mid i\le j;\ i,j=1,2,\ldots,M\}$.

Если известны (f_i,f_j) $\forall\ i,j\ (i\le j)$, то для реализации метода Грамма-Шмидта требуется $O(M^3)$ операций. Очевидно, сложность метода ортогонализации не может быть меньше $O(M^3)$, так как каждый из M векторов-функций e_k строится ортогонально подпространству размерностью $1 \div (M-1)$ - в среднем около $M/2$, причем для вычисления скалярных произведений (e_k,f_i) требуется в среднем около $M/2$ операций.

В работе /6/ показано, что метод Грамма-Шмидта предпочтительнее метода Форсайта (Вайсфелда) по требуемому для реализации количеству операций. При этом для вычисления $\{(f_i,f_j)\}$ предлагается вычислять набор значений $\{f_1(\vec{x}^{(k)}),\ f_2(\vec{x}^{(k)}),\ldots,f_M(\vec{x}^{(k)})\}$ в точке $\vec{x}^{(k)}$ (это требует M операций), затем

$$\{f_i(\vec{x}^{(k)})\cdot f_j(\vec{x}^{(k)})\cdot w_k \mid i\le j;\quad i,j=1,2,\ldots,M\}$$

(вместе с предыдущим шагом это требует $O(M^2)$ операций) и произвести суммирование по всем N точкам - всего $O(N\cdot M^2)$ операций.

2. Описание отличительных особенностей алгоритма

Отличительной особенностью предлагаемого в настоящей работе алгоритма является способ вычисления $\{(f_i,f_j)\}$.

Пусть $f_M(\vec{x}) = x_1^{m_1} x_2^{m_2}\ldots x_n^{m_n}$, тогда для вычисления $\{(f_i,f_j) \mid i\le j;\ i,j=1,2,\ldots,M\}$ существует алгоритм, требующий не более $O(2^n\cdot N\cdot M)$ операций.

Для доказательства этого утверждения воспользуемся тем, что $(f_i,f_j) = (f_1,f_\ell)$, где

$$i = N_n(i_1,i_2,\ldots,i_n),\quad j = N_n(j_1,j_2,\ldots,j_n),$$

$$\ell = N_n(i_1+j_1,\ i_2+j_2,\ldots,i_n+j_n).$$

Последнее очевидно из соотношений

$$(f_i,f_j) \overset{def}{=} \sum_{k=1}^{N} w_k\left[(x_1^{(k)})^{i_1}\cdot(x_2^{(k)})^{i_2}\ldots(x_n^{(k)})^{i_n}\right]\left[(x_1^{(k)})^{j_1}(x_2^{(k)})^{j_2}\ldots(x_n^{(k)})^{j_n}\right] = \sum_{k=1}^{N} w_k\left[(x_1^{(k)})^{i_1+j_1}\cdot(x_2^{(k)})^{i_2+j_2}\ldots(x_n^{(k)})^{i_n+j_n}\right] \overset{def}{=} (f_1,f_\ell)$$

Докажем теперь, что если $i,j=1,2,\ldots,M$, то

$$\ell \le N_n(2m_1,2m_2,\ldots,2m_n) = M'. \qquad (5)$$

Предположим $\exists\ \vec{i}' = (i_1', i_2', \ldots, i_n')$ и $\vec{j}' = (j_1', j_2', \ldots, j_n')$, удовлетворяющие условиям

$$N_n(i_1', i_2', \ldots, i_n') \le N_n(m_1, m_2, \ldots, m_n) = M \qquad (6)$$

$$N_n(j_1', j_2', \ldots, j_n') \le N_n(m_1, m_2, \ldots, m_n) = M \qquad (7)$$

и в то же время такие, что

$$N_n(i_1'+j_1', i_2'+j_2', \ldots, i_n'+j_n') > N_n(2m_1, 2m_2, \ldots, 2m_n). \qquad (8)$$

Тогда, по определению отношения порядка на F либо

$$q(\vec{i}' + \vec{j}') > q(2\vec{m}), \qquad (9)$$

либо

$$q(\vec{i}' + \vec{j}') = q(2\vec{m}) \quad \text{и} \quad \exists\ k \le n, \quad \text{что} \qquad (10)$$

$$\left[(i_k' + j_k') + (i_{k+1}' + j_{k+1}') + \ldots + (i_n' + j_n')\right] > 2(m_k + m_{k+1} + \ldots + m_n). \qquad (11)$$

Поскольку $\vec{i}'$ и $\vec{j}'$ удовлетворяют по предположению условиям (6) и (7), то $q(\vec{i}') \le q(\vec{m})$, $q(\vec{j}') \le q(\vec{m})$ и, следовательно, (9) не может выполняться. Рассмотрим теперь (10) и (11). Неравенство (11) представим в виде

$$\left[(i_k'-m_k) + (i_{k+1}' - m_{k+1}) + \ldots + (i_n' - m_n)\right] + \left[(j_k'-m_k) + (j_{k+1}' - m_{k+1}) + \ldots + (j_n' - m_n)\right] > 0,$$

откуда следует, что хотя бы одно из слагаемых в квадратных скобках -величина положительная. Без ограничения общности будем считать, что

$$(i_k' - m_k) + (i_{k+1}' - m_{k+1}) + \ldots + (i_n' - m_n) > 0,$$

но это, в соответствии с определением порядка для f_i, и f_M, эквивалентно неравенству

$$N_n(i_1', i_2', \ldots, i_n') > N_n(m_1, m_2, \ldots, m_n)$$

и противоречит предположению (6).

Полученные противоречия показывают, что предположение (8) неверно, и утверждение (5) доказано. Таким образом,

$$\forall\ i,j = 1,2,\ldots,M \quad \exists\ \ell \le N_n(2m_1, 2m_2, \ldots, 2m_n) = M',$$

что

$$(f_i, f_j) = (f_1, f_\ell).$$

Для вычисления (f_1, f_ℓ), $\ell = 1,2,\ldots,M'$, как следует из [6], можно затратить не более $N \cdot M'$ операций.

Оценим теперь M' через M. В соответствии с определением

$$M' = N_n(2m_1, 2m_2, \ldots, 2m_n) = B^n_{2q_1-1} + B^{n-1}_{2q_2-1} + \ldots + B^1_{2q_n-1} + 1,$$

где $q_i = \sum_{k=i}^{n} m_k$. Воспользуемся неравенством $\forall\ n,\ q \ge 1$

$$B^n_{2q-1} = \frac{(n+2q-1)!}{n!\,(2q-1)!} = \frac{(2q)\ (2q+1)\ \ldots\ (2q+n-1)}{n!} =$$

$$= 2^n \frac{q\ (q+\frac{1}{2})\ \ldots\ (q+\frac{n-1}{2})}{n!} < 2^n \frac{q(q+1)\ \ldots (q+n-1)}{n!} = 2^n B^n_{q-1}.$$

Тогда

$$M' < 2^n \cdot B^n_{q_1-1} + 2^{n-1} B^{n-1}_{q_2-1} + \ldots + 2B^1_{q_n-1} + 1$$

или, на основе еще более слабой, но наглядной оценки,

$$M' < 2^n (B^n_{q_1-1} + B^{n-1}_{q_2-1} + \ldots + B^1_{q_n-1} + 1) = 2^n \cdot M. \qquad (12)$$

Из (12) следует, что предложенный алгоритм вычисления $\{(f_i, f_j) \mid i \le j;\ i,j = 1,2,\ldots,M\}$ требует не более $O(2^n \cdot N \cdot M)$ операций.

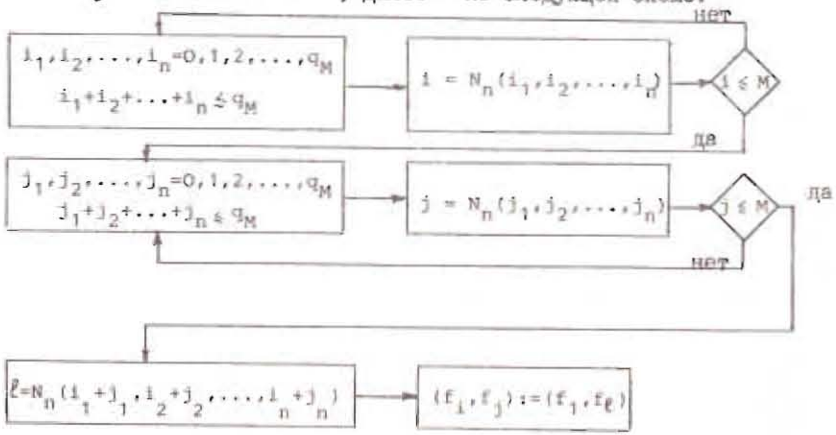
Покажем теперь, что для рассматриваемых алгебраических одночленов и скалярного произведения предложенный алгоритм вычисления $\{(f_i, f_j) \mid i \le j;\ i,j = 1,2,\ldots,M\}$ является предельно быстрым в зависимости от M, то есть показатель степени M в оценке сложности алгоритма $O(2^n \cdot N \cdot M)$ не может быть меньше единицы.

Предположим, что при фиксированном n существует алгоритм вычисления $\{(f_i, f_j)\}$ сложности $O(N \cdot M^{1-\alpha})$, $\alpha > 0$. В силу изотропности множества Ω относительно количества операций над координатами каждой точки требуется $O(M^{1-\alpha})$ операций в каждой точке. Поскольку в каждой точке вычисляется значение каждой из M функций $\{f_i\}$, необходимо $O(M^{1-\alpha})/M = O(M^{-\alpha})$ операций для вычисления одной функции в точке. Но $O(M^{-\alpha}) \xrightarrow[M \to \infty]{} 0$, что противоречит неявному предположению о необходимости как минимум одной операции для вычисления одночлена в точке.

Для реализации изложенного способа вычисления (f_i, f_j) в методе ортогонализации Грамма-Шмидта необходимо обеспечить способ доступа к элементам $\{(f_i, f_j)\}$, имея в качестве исходных данных массив $\{(f_1, f_\ell),\ \ell = 1,2,\ldots,M'\}$ и пары индексов (i,j). В [15] описано правило определения для произвольного индекса i компонент соответствующего вектора $(i_1, i_2, \ldots, i_n)$ такого, что $f_i(\vec{x}) = x_1^{i_1} x_2^{i_2} \ldots x_n^{i_n}$. Следуя этому правилу, можно было бы обеспечить доступ к (f_i, f_j) по схеме

$$(i,j) \begin{matrix} \nearrow (i_1,i_2,\ldots,i_n) \searrow \\ \searrow (j_1,j_2,\ldots,j_n) \nearrow \end{matrix} N_n(i_1+j_1,i_2+j_2,\ldots,i_n+j_n) = \ell \to (f_1,f_\ell) = (f_i,f_j),$$

но вычисление компонент векторов $\vec{i}$ и $\vec{j}$, а также функции нумерации от их суммы является процедурой сложности $O(n^3)$. Ее использование внутри трех вложенных циклов длиной $O(M^3)$ мультипликативно увеличило бы сложность метода ортогонализации. По этой причине предлагается предварительное формирование треугольной матрицы, элементами которой являются $\{(f_i,f_j)\}$, $i \leq j$; это лишь аддитивно увеличивает сложность метода ортогонализации. На первом шаге формирования матрицы определим максимальную степень одночленов q_M по известному значению M /I5/, далее – по следующей схеме:



Предложенный способ организации доступа к элементам (f_i,f_j) требует $M(M+1)/2$ слов памяти для хранения значений $(f_i,\ f_j)$, $i \leq j$, имеет сложность $O(M^2)$ и всего лишь аддитивно увеличивает сложность метода ортогонализации: $O(M^2)+O(M^3)=O(M^3)$.

3. О программной реализации алгоритма

Для программной реализации алгоритма требуется пространство памяти размером $n \times N$ – для входного массива, содержащего координаты точек множества Ω, $M(M+1)/2$ – для выходного массива, содержащего элементы матрицы A, связующей построенную систему ортонормированных полиномов Π с системой алгебраических одночленов F, $2^{n+1}\cdot M + M(M+1)/2$ – для рабочих массивов $\{f_\ell(\vec{x}),\ \ell=1,2,\ldots,M' \leq 2^n \cdot M\}$, $\{(f_1,f_\ell),\ \ell=1,2,\ldots,M' \leq 2^n\cdot M\}$ и $\{(f_i,f_j)\mid i,j=1,2,\ldots,M;\ i \leq j\}$, причем массивы $\{f_\ell(\vec{x})\}$ и $\{(f_1,f_\ell)\}$ можно разместить в пространстве памяти, отведенном для матрицы A, зарезервировав для нее пространство памяти длиной $\max(2^{n+1}\cdot M,\ M(M+1)/2)$ слов. Если $2^{n+2} \leq M$, то потребуется $M(M+1)$ слов памяти для рабочих массивов и размещения элементов матрицы A, что меньше $N \cdot M$ слов, как в программе /5/.

Как известно /20/, метод ортогонализации обеспечивает очень высокую степень устойчивости в смысле малости эквивалентных возмущений, и неустойчив в смысле ортогональности системы получаемых векторов. Для устранения отмеченной неустойчивости применяется переортогонализация получаемых векторов /20/. Хорошие результаты дает также применение режима вычислений с двойной точностью.

Алгоритм реализован на языке FORTRAN-IV для ЭВМ СМ-4* в программе, выполняющей аппроксимацию набора экспериментальных данных $(u_i,\ v_i)$, $i=1,2,\ldots,N$ на множестве (x_i,y_i), $i=1,2,\ldots,N$. Для $n=2$, $N=225$, $M=21$ время счета (при вычислении с двойной точностью) составило 45 секунд, что примерно в 3,5 раза быстрее, чем при счете по алгоритму /6/ в тех же условиях.

Заключение

Ортогональные полиномы от многих переменных получают широкое применение при обработке экспериментальных данных (калибровки оптических и магнитных систем, аппроксимация физических величин и т.п.), а также в других областях цифровой обработки. Однако в ряде случаев объем обрабатываемых данных (N – до 10000 точек и более), размерность пространства данных ($n = 2 \div 4$, иногда до 10), длина ряда разложения (M – до 60 членов) /5,6,II/ требуют значительного вычислительного ресурса ЭВМ. При этом, в известных ранее алгоритмах большая часть времени затрачивается в главном, зависящем от N, цикле вычислений (как правило, $N \gg M$). Предложенный алгоритм имеет существенно меньшее количество операций в главном цикле – $C_1 \cdot N \cdot M'$ $(M' < 2^n \cdot M)$, что делает его асимптотически предельно быстрым в зависимости от M, а при оптимальном программировании этого алгоритма (что влияет на значение константы C_1) – предположительно абсолютно быстрым в зависимости от M при фиксированных n и N $(N \gg M)$.

Сравнение с алгоритмами /II,I3-I6/ и /5,6/, имеющими формулы быстродействия соответственно

$$A \cdot N \cdot M^{3-\frac{1}{n}},\quad A \gtrsim 1 \quad \text{и}$$

$$B_1 \cdot N \cdot M^2 + B_2 \cdot M^3 \qquad B_1 \gtrsim 1,\quad B_2 \approx \frac{1}{4},$$

---

* С программной реализацией операций с плавающей запятой

а также результат счета по описанному выше алгоритму и по алгоритму [6] при $n = 2$, $N = 225$, $M = 21$ показывают возможность многократного выигрыша времени при использовании предложенного алгоритма, имеющего формулу быстродействия

$$c_1 \cdot 2^n \cdot N \cdot M + c_2 \cdot M^3 , \quad c_1 \gtrsim 1, \quad c_2 \approx \frac{1}{4} .$$

Особый интерес представляет применение данного алгоритма на малых ЭВМ. Автор выражает благодарность О.Е.Горчакову, Н.Б.Богдановой и Г.А.Ососкову за полезные обсуждения опубликованных результатов.

Литература

1. Сеге С. Ортогональные многочлены. Физматгиз, М., 1962.
2. Beckmann P. Orthogonal Polynomials for Engineers and Physicists. The Golem Press, Boulder (Colorado, USA), 1973.
3. Суетин П.К. Классические ортогональные многочлены. "Наука", М., 1979.
4. Крылов В.И. Приближенное вычисление интегралов. "Наука", М., 1967.
5. Brun R. et al. CERN DATA HANDLING DIVISION, DD/75/23, Geneva, June, 1977.
6. Горчаков О.Е. ОИЯИ, Р11-84-188, Дубна,
7. Forsythe G.E. J. Soc. Industr. Appl. Math., 1957, vol. 5, No.2, p. 74 - 88 .
8. Weisfeld M. Numerische Mathematik, 1959, vol.1, p.38-40.
9. Hudson D. LSQFIT CERN Program Library E 202.
10. Pomentale T. PSI1 CERN Program Library E 204.
11. Pomentale T. PSI2 CERN Program Library E 205.
12. Гаджоков В., Богданова Н. ОИЯИ, Р11-12-860, Дубна, 1979.
13. Гаджоков В., Богданова Н. ОИЯИ, Р11-80-122, Дубна, 1980.
14. Гаджоков В., Богданова Н. ОИЯИ, Р11-80-781, Дубна, 1980.
15. Гаджоков В. ОИЯИ, Е11-80-782, Дубна, 1980.
16. Bogdanova N.,Gadjokov V.,Comp. Phys. Comm., 1981, vol. 24, No.2, p. 225-229.
17. Байла И., Ососков Г.А. ОИЯИ, Р10-11834, Дубна, 1978.
18. Байла И., Ососков Г.А., Хэрн А.К. ОИЯИ, Р10-11944, Дубна,1978.
19. Форсайт Дж., Малькольм М., Моулер К. Машинные методы математических вычислений, "Мир", М., 1980.
20. Воеводин В.В. Вычислительные основы линейной алгебры, "Наука", М., 1977.

Рукопись поступила в издательский отдел
5 октября 1985 года.

Сенченко В.А. Р5-85-724

Предельно быстрый алгоритм построения ортонормированной системы полиномов от многих переменных на произвольном дискретном множестве

Предложен предельно быстрый алгоритм построения ортонормированной системы полиномов от многих переменных на произвольном конечном дискретном множестве. Сложность алгоритма не превышает $O(2^n \cdot N \cdot M) + O(M^3)$, где n - количество переменных, N - количество элементов дискретного множества, M - количество ортогональных полиномов. В сравнении с известными алгоритмами сложности $O(N \cdot M^{3-1/n})$ и $O(N \cdot M^2) + O(M^3)$ соответственно показана возможность многократного выигрыша времени при использовании предложенного алгоритма на ЭВМ.

Работа выполнена в Лаборатории вычислительной техники и автоматизации ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна 1985

Перевод Н.С.Панковой

Senchenko V.A. P5-85-724

The Extremely Fast Algorithm Generating Orthonormal Multivariable Polynomials at Any Discrete Point Set

The extremely fast algorithm generating orthonormal multivariable polynomials at any finite discrete point set is presented. The complexity /the number of operations/ of the algorithm does not exceed $O(2^n \cdot N \cdot M) + O(M^3)$ were n is the number of variables, N-the number of points of the discrete set, M-the number of orthonormal polynomials. The possibility of repeated saving of time by using the presented algorithm at computer is shown in comparison with known alootithms, that have complexity $O(N \cdot M^{3-1/n})$ and $O(N \cdot M^2) + O(M^3)$, respectively.

The investigation has been performed at the Laboratory of Computing Techniques and Automation, JINR.

Preprint of the Joint Institute for Nuclear Research. Dubna 1985 .