tanulmányok

202/1987

MTA Számítástechnikai és Automatizálási Kutató Intézet Budapest

MAGYAR TUDOMANYOS AKADÊMIA SZÂMÎTÂSTECHNIKAI ÊS AUTOMATIZÂLÂSI KUTATÓ INTÊZETE
COMPUTER AND AUTOMATION INSTITUTE, HUNGARIAN ACADEMY OF SCIENCES
ИССЛЕДОВАТЕЛЬСКИЙ ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И АВТОМАТИЗАЦИИ
ВЕНГЕРСКОЙ АКАДЕМИИ НАУК

PROBLEMS OF COMPUTER SCIENCE

PROCEEDINGS OF THE JOINT WORKSHOP
OF COMPUTER AND AUTOMATION INSTITUTE
OF HAS AND COMPUTING CENTRE OF ARMENIAN
ACADEMY OF SCIENCES HELD IN

Budapest, September, 1987.

Edited by *G.B. MARANDŽJAN, B. UHRIN*

ПРОБЛЕМЫ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ

ТРУДЫ ОБЪЕДИНЕННОГО СЕМИНАРА
ИССЛЕДОВАТЕЛЬСКОГО ИНСТИТУТА
ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ И АВТОМАТИЗАЦИИ
ВАН И ВЫЧИСЛИТЕЛЬНОГО ЦЕНТРА АН АРМЯНСКОЙ ССР

Будапешт, сентябрь 1987 г.

Под редакцией *Г.Б. МАРАНДЖЯНА, Б. УХРИНА*

Tanulmányok 202/1987
Studies 202/1987

A kiadásért felelős:

KEVICZKY LÁSZLÓ

Főosztályvezető:

DEMETROVICS JÁNOS

ISBN 963 311 242 7
ISSN 0237-0131

C O N T E N T S

С О Д Е Р Ж А Н И Е

С.С. АГАРЯН, К.О. ЕГИАЗАРЯН: Новый класс преобразований в теории обработки дискретных сигналов 7

А.А. АРАКЕЛЯН: Методы оптимизации диалоговых систем экспериментальных исследований 25

J.DEMETROVICS, L. HANNÁK, L. RÓNYAI: On monotone clones 39

G. FREIMAN, A. HEPPES, B. UHRIN: A lower estimation for the cardinality of finite difference sets in R^n..... 63

A. KRÁMLI, P. LUKÁCS, D. SZÁSZ: A non-Wiener random walk in a 2-D Bernoulli environment 75

Г.Б. МАРАНДЖЯН: Минимальные номера в предполных нумерациях 99

L.L. MÁTÉ: CAD Tool Manager: an alternative system structure in electronic CAD 107

А.Е. МЕЛКОНЯН, В.А. БАРСЕГЯН, А.А. АЙВАЗЯН: Теоретико-модельный подход к оценке и управлению качеством продукции в условиях неопределенности параметров производственной системы 111

А.Е. МЕЛКОНЯН, С.М. МАНУЧАРЯН: Динамическая модель оптимального водораспределения в орошаемом земледелии с учетом ресурсных ограничений 131

KATALIN PÁSZTORNÉ VARGA: Logic programming and the algorithmic type problems 149

Э.М. ПОГОСЯН: К адаптивному синтезу стратегий управления .. 159

I. RATKÓ: A short note on measures of qualitative variables for disease studies 171

. RATKÓ, P. KERÉKFY, A. KRÁMLI, A. KISS, M. RUDA, J. SOLTÉSZ, J. DUBA, M. CSUKÁS, E. FARKAS, G. MARÓTI: Realization of large-sized pharmacological effect examination with microcomputer 175

.Г. ШУКУРЯН: О некоторых проблемах автоматизированного программирования микропрограммных вычислительных структур 181

. SZIRAY, Zs. NAGY, L.L. MÁTÉ: An expert system for test design of digital circuits 195

. UHRIN: The measure of covering the space by a set A and the projection successive minima of A-A 209

. RÓNYAI: A note on factoring polynomials modulo special primes 227

F O R E W O R D

The first joint workshop on problems of computer science, organized - in the framework of scientific cooperation between the Hungarian and Armenian Academies of Sciences - by Computer and Automation Institute of HAS and Computing Centre of AAS, was held in September, 1987, in Budapest.

The leading scientists from both institutes took part in the workshop. The themes of the workshop were those laid down in scientific plans of the cooperation and concerned some problems of computer science and computer technics.

The present volume reports on research continued in the course of cooperation and presented at the workshop. The papers are intended to researchers in the fields of mathematical problems of computer science and automatization, software for automatization systems and mathematical modelling of practical problems.

We hope that the proposed next workshop to be held in Erevan /Armenia, 1988/ will contribute to success of Armenian and Hungarian scientists of both Institutes in promoting the research of actual problems of computer science and applications.

Budapest, September 4-14, 1987

Ju. SHUKURIAN
Professor

J. DEMETROVICS
Corresponding Member of HAS

Предисловие

В сентябре 1987 г. в г. Будапеште в рамках двухстороннего научного сотрудничества между Академией наук Армянской ССР и Венгерской Академией наук состоялся первый совместный семинар по информатике, организованный Исследовательским институтом вычислительной техники и автоматизации ВАН /СТАКИ/ и Вычислительным центром АН Арм. ССР. В работе семинара приняли участие ведущие сотрудники и научные руководители по темам сотрудничества из обеих стран. В период работы семинара были заслушаны и обсуждены актуальные вопросы современных проблем информатики и вычислительной техники. Настоящий сборник трудов подитоживает работу семинара и научные результаты по двухстороннему сотрудничеству. Он может быть полезен научным работникам и специалистам в области математических задач информатики и автоматизации, математического обеспечения систем автоматизации и математического моделирования прикладных задач.

Намечаемый совместный семинар по информатике в г. Ереване /СССР, май-июнь 1988 г./ несомненно будет способствовать дальнейшему расширению сферы современных усилий армянских и венгерских ученых по расширению актуальных задач информатики и вычислительной техники.

Будапешт, 4-14 сентября 1987 г.

Ю. Шукурян
Профессор

Я. Деметрович
Член-корр. ВАН

MTA SZTAKI Tanulmányok 202/1987 pp. 7-23

НОВЫЙ КЛАСС ПРЕОБРАЗОВАНИЙ В ТЕОРИИ ОБРАБОТКИ ДИСКРЕТНЫХ СИГНАЛОВ

Агаян С.С., Егиазарян К.О.

I. Введение

В последние десятилетия, благодаря увеличению быстродействия ЦВМ, большое применение во многих задачах цифровой обработки сигналов на различных ее этапах (кодировании, восстановлении, сокращении избыточности) находят дискретные ортогональные преобразования. Классическими преобразованиями такого типа являются преобразования Фурье, Хаара, Слэнт и др. Первые из них, образующие класс дискретных преобразований Фурье(ДПФ), основанный на различных системах базисных функций - дискретных экспоненциальных функций (ДЭФ), функций Виленкина-Крестенсона (ВКФ), Виленкина-Понтрягина (ВПФ) и т.д., являются наиболее изученными и применимыми [4,14].

Известно ([3,16]), что ортогональные базисы являются оптимальными решениями перечисленных выше задач для определенного узкого класса процессов. С целью расширения этого класса создаются все новые базисные системы, в особенности, системы с заранее заданными свойствами. Вместе с этим намечается тенденция к нахождению единого метода их построения. Так, гармонический анализ на конечных абелевых группах позволил объединить существующие (Фурье-подобные) ДОП (Уолша, Адамара, теоретико-числовые преобразования и др.) в один класс, получить общий подход к алгоритмам быстрых ортогональных преобразований [6,8-10,12,18,19]. Относительно ДОП, определенных на конечных неабелевых группах, отметим работу [7].

В указанных практических задачах (в связи с обработкой больших массивов данных) существенное значение имеют вопросы эффективной реализации ДОП - построение асимптотически быстрых алгоритмов вычисления ДОП со сложностью, близкой к нижней границе.

Дискретным ортогональным преобразованием (входного) вектора $f=[f_0, f_1, \dots, f_{N-1}]^T$ по базисной системе, определяемой ортогональной матрицей $V=\|V_{jk}\|, (j,k=\overline{0,N-1})$ с элементами из некоторого коммутативного кольца с единицей называется (выходной) вектор $F=Vf$. ДОП, у которых элементы матрицы V состоят из корней ρ-ой степени из I в R, будем называть Фурьеподобными ДОП.

2. Гармонический анализ в базисах обобщенных функций Виленкина

Пусть G - произвольная конечная абелева группа, R -коммутативное кольцо с единицей, содержащее примитивный корень степени $exp\,G$ из I ($exp\,G$- наименьшее натуральное число, такое, что $(exp\,G)g=0$ для всех $g\in G$). Хорошо известно [8,II, I8], что характеры группы G, принимающие значения в кольце R, образуют полную мультипликативную ортонормированную систему функций, образующую, в свою очередь, изоморфную G двойственную к ней группу G'. Итак, группа G (G' определяется по G), и кольцо R задают мультипликативную ортогональную систему и структуру базиса, по которому определяется ДПФ на конечной абелевой группе G некоторой функции $f: G\to R$:

$$(F_G f)(\chi)=\sum_{g\in G} f(g)\chi(g)$$

где $\chi(g)\in G; g\in G$. Преобразования такого типа рассмотрены в работах [6,8-I0,I2,I8,I9].

С целью синтеза на G новых ортогональных базисов определим следующую тройку $(G, G^{\times}, R)$, где $G^{\times}$ - произвольная изоморфная G группа с элементами $H(g)\in G^{\times}(g\in G)$, образующими в R ортогональную систему функций, т.е.

$$H(g'+g'')=H(g')\odot H(g'') \quad \text{(условие изоморфизма)} \qquad (I)$$

$$\langle H_a(g), H_b(g)\rangle_G=[G]^{-1}\sum_{g\in G} H_a(g)\overline{H_b(g)}=\delta_{ab} \qquad (2)$$

(условие ортогональности)

где δ_{ab} - символ Кронекера [I4], $a, b, g, g', g''\in G$, $+$ и $\odot$ - операции в группах G и $G^{\times}$ соответственно.

Задав ограничения на кольцо R, характеризующие изоморфизм G и $G^{\times}$, и найдя аналитический вид элементов группы

G^x, определим Фурье-подобное ДОП на конечной абелевой группе G некоторой функции $f: G \to R$ по

$$(F_G f)(H) = \sum_{g \in G} f(g) H(g) \qquad (3)$$

(с симмированием в групповом кольце $R[G]$).

Введем теперь некоторые обозначения. Пусть $T = \exp G$, L = Н.О.К. (p, T), ζ_p – примитивный корень из I степени p в R, где p – произвольное натуральное число, Н.О.К. – наименьшее общее кратное. Обозначим через $\psi(c)$ ($c \in Z_{q_i}$) целочисленную функцию, удовлетворяющую условиям:

$$\begin{cases} \zeta_{q_i}^{\psi(c)} = \zeta_{q_i}^{c} \\ |\psi(c)| \le \left[\frac{q_i}{2}\right] \end{cases}$$

и определенную по

$$\psi(n) = n \quad \text{для } n = 0, 1, \ldots, \left[\frac{q_i}{2}\right]$$

$$\psi\left(\left[\frac{q_i}{2}\right] + K\right) = -\left]\frac{q_i}{2}\right[+ K \quad \text{для } K = 1, 2, \ldots, \left]\frac{q_i}{2}\right[- 1$$

<u>Утверждение I</u>. Пусть G – конечная абелева группа, R – коммутативное кольцо с единицей, содержащее примитивный корень степени L из I, и группа G^x определена следующим образом:

I) если $G = Z_N$, то

$$H_a(g) \circledcirc H_B(g) = H_a(g) H_B(g) \qquad (4)$$

2) если $G = \sum_{i=1}^{n} G_i = \sum_{i=1}^{n} Z_{q_i^{K_i}}$, где q_i – простые, K_i – натуральные числа, $n \ge 2$, $i = \overline{1.n}$, то

$$H_a(g) \circledcirc H_B(g) = \begin{cases} H_a(g) H_B(g) (= H_c(g)), & \text{когда } q_i \neq 2 \\ & \text{при } a_i \text{ или } B_i = 0 \pmod{q_i} \\ \overline{H_c(g)} & \text{в противном случае} \end{cases} \qquad (5)$$

где $a + b + c = 0$; $a, b, c, g \in G$

Тогда $\{H_a(g)\}$ образует ортогональные системы функций, причем случаю I) соответствует вид

$$H_a(g) = \zeta_r^{ag} \qquad (6)$$

а случаю 2):

$$H_a(g) = \xi_T^{\sum_{i=0}^{n-1} a_{n+1-j} g_j} \cdot \xi_4^{t\psi(g_m)\psi(a_m)} \qquad (7)$$

где $m = \overline{0, K_i - 1}$; $t = 0, 1$, которые мы назовем системами обобщенных функций Виленкина (ОВФ).

Доказательство. Известно [II], что произвольную конечную абелеву группу можно разложить в прямую сумму примарных циклических подгрупп: $G = \sum_{i=1}^{n} G_i$ (где $[G_i] = N_i = q_i^{K_i}$; $[G] = N = \prod_{i=1}^{n} N_i$, q_i - простые, K_i - натуральные числа, $i = \overline{1,n}$), каждая из которых изоморфна группе целых чисел Z_N с операцией сложения по модулю N_i. Зафиксируем этот изоморфизм и под будем подразумевать группу Z_{N_i}. Нетрудно проверить, что функции, определенные по (7), образуют в R дискретный ортогональный базис (выполняется условие (2)) и образуют группу G^x, изоморфную G и определенную по (5). Функции, удовлетворяющие (6), являются известными характерами группы G и образуют двойственную группу $G^x = G'$ в случае, когда G является циклической группой.

Замечание I. В случае I) утверждения I функции $H(g)$, определенные по (4), образуют мультипликативную группу G^x относительно обычного умножения функций - характеров группы G, и задают известный ортогональный базис (6) типа ДЭФ [6,8,I2, I4,I8].

Замечание 2. В случае 2) (при $q_i^{K_i} = q^K$, $i = \overline{1,n}$ и $4 = T$) утверждения I, каждому элементу $a \in G$, записываемому в виде $a = (a_1, a_2, \ldots, a_n)$, $(a_i \in G_i, i = \overline{1,n})$ однозначно соответствует базисная функция $H_a(g) \in G^x$ $(g \in G)$ типа ВКФ [6,8,I0,I2], имеющая вид (7) при $t = 0$:

$$H_a(g) = \chi_a(g) = \xi_T^{\sum_{j=1}^{n} a_{n+1-j} g_j} \qquad (8)$$

где $g = (g_1, g_2, \ldots, g_n) \in G$; $g_i \in G_i$, $i = \overline{1,n}$. Указанные базисные функции, записанные здесь в обратном лексикографическом порядке, можно задать также и другими способами [I0,I4], как, например, в лексикографическом порядке:

$$H_a(g) = \chi_a(g) = \xi_T^{\sum_{j=1}^{n} a_j g_j}$$

Замечание 3. Если в утверждении I имеет место случай 2) при различных $N_i = q_i^{k_i}$ и $L = T$, то аналитический вид базисных функций-характеров $H_a(g) \in G^x$ $(a, g \in G)$ типа ВПФ [6,8, I2] аналогичен (8) с той лишь разницей, что ξ_T является здесь Н.О.К. $(\exp G_i ; i = \overline{1,n})$.

Группа G^x, изоморфная G, определена в утверждении I так, чтобы системы образующих ее ортогональных базисов, содержащие в себе (см.замечания I,2,3) все известные Фурье-базисы (характерны), имели в то же время такие практически хорошие свойства этих базисов, как симметричность, периодичность и т.д. Более общее условие, накладываемое на кольцо R (т.е. $\xi_L \in R$), позволило синтезировать новые дискретные ортогональные базисы на группе G. В таблице (приведенной в конце работы) дана классификация Фурье-подобных ДОП, определенных на конечной абелевой группе. Рассмотрены преобразования в кольцах $R \in \{C, Z_M (M = F_d, M_p), Z_M[i], Z_M[\theta], Z_M[i\theta], Q(x)/(m(x))\}$, где C - поле комплексных чисел, Z_M - кольцо вычетов по модулю M, F_d, M_p - числа Ферма и Мерсенна, $Z_M[i]$ - кольцо целых комплексных чисел, θ - число Эйзенштейна, $Q(x)/(m(x))$ - полиномиальное кольцо классов вычетов кольца полиномов $Q(x)$ по модулю главного идеала, образуемого полиномом $m(x)$ [6,I3,I9]. Выделенные знаком * ДОП приведены для сравнения из работы [I9], а ** - получены впервые. Знаки + и - под графами G' и G^x означают, что указанные преобразования могут или не могут соответственно быть синтезированы заданием изоморфных групп G' и G^x, где G' - двойственная к G группа, а G^x - введенная по (4) и (5) изоморфная G группа. Как видно из таблицы, все указанные ДОП могут быть получены заданием (G, G^x, R). Список сокращений и обозначений приведен после таблицы.

Рассмотрим теперь пример получения базисных функций типа ОВФ.

Пусть $G = Z_3^2$ и в R существует примитивный корень ξ 6-ой степени из I. Базисные функции $H(g) \in G^x$, полученные по (7) при $m = 0$, $t = 1$, образуют приведенную в (9) ортогональную симметричную матрицу H. Каждому элементу $b \in G$, предста-

вимому в виде $b=(b_1,b_0)$, $b_1,b_0\in Z_3$ однозначно сопоставимо целое $b=3b_0+b_1$; базисные функции $H_b(g)$ с номерами b, являющиеся строками матрицы H, образуют конечную абелеву группу G^x с операцией $\odot$, заданной по (5), изоморфную группе G.

$$H=\left\|\begin{array}{ccccccccc}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1\\
1 & \xi & \xi^5 & \xi^2 & \xi^3 & \xi & \xi^4 & \xi^5 & \xi^3\\
1 & \xi^5 & \xi & \xi^4 & \xi^3 & \xi^5 & \xi^2 & \xi & \xi^3\\
1 & \xi^2 & \xi^4 & 1 & \xi^2 & \xi^4 & 1 & \xi^2 & \xi^4\\
1 & \xi^3 & \xi^3 & \xi^2 & \xi^5 & \xi^5 & \xi^4 & \xi & \xi\\
1 & \xi & \xi^5 & \xi^4 & \xi^5 & \xi^3 & \xi^2 & \xi^3 & \xi\\
1 & \xi^4 & \xi^2 & 1 & \xi^4 & \xi^2 & 1 & \xi^4 & \xi^2\\
1 & \xi^5 & \xi & \xi^2 & \xi & \xi^3 & \xi^4 & \xi^3 & \xi^5\\
1 & \xi^3 & \xi^3 & \xi^4 & \xi & \xi & \xi^2 & \xi^5 & \xi^5
\end{array}\right\| \qquad (9)$$

Перестановка строк и столбцов матрицы H по закону q - ичной ($q=3$) инверсии их номеров даст матрицу H', которую можно получить по (7) при $m=1$, $t=1$. В случае $R=C$ матрицы H и H' будут являться $H(6,9)$-обобщенными матрицами Адамара [I] -квадратными порядка 9 ортогональными матрицами с элементами-корнями 6-ой степени из I.

Замечание 4. Группа G^x, изоморфная группе G, может быть, вообще говоря, определена иным в утверждении I, способом. Множество всевозможных заданий троек (G,G^x,R) (с условиями изоморфизма групп G и G^x, налагаемыми на R), будет определять класс Фурье-подобных систем базисных функций.

3. Сложность и быстрые алгоритмы преобразований

Быстрые дискретные ортогональные преобразования (БДОП) - эффективные по быстродействию, значительно снижающие число операций, алгоритмы вычисления ДОП, в то время как их непосредственное вычисление требует порядка N^2 операций, где

N - длина обрабатываемого массива данных. Задача быстрого вычисления произвольного ДОП является частной задачей быстрого вычисления произведения двух матриц (известно [5], что две матрицы порядка $N=2^n$ можно умножить за $N^{\log_2 7}$ операций), когда одна из матриц является ортогональной и имеет определенную структуру. Первые работы, посвященные этой проблеме, восходят к началу XX века, однако существенным толчком явилась работы Кули и Таки [15] о быстром вычислении ДПФ длины N за $O(N \ln N)$ операций.

В работе [17] Моргенштерном получена оценка нижней границы для числа сложений, необходимых в вычислении семейства линейных функций по линейному алгоритму в поле комплексных чисел, а также аналогичная оценка в более частном случае быстрого преобразования Фурье (БПФ): $\frac{N}{2}\log N$, где N - порядок преобразования. Очевидно, что эта оценка является асимптотической, поскольку вычисление БПФ требует $N\log N$ операций (верхняя граница). Алгоритм Моргенштерна, дающий оценку лишь для числа сложений, невозможно прямо применить для ДОП, заданных не в поле комплексных чисел и чьи порядки отличны от $N=2^n$. Вследствие этого возникает задача модификации этого алгоритма применительно ко всем классам ДОП, в частности, и к Фурье-подобным преобразованиям в базисах типа ОВФ на конечной абелевой группе.

Использование группового аппарата в построении БДОП показало, что причина существования быстрых алгоритмов кроется в самой структуре группы, на которой они определяются. В основе известных алгоритмов БПФ на конечной абелевой группе G лежит идея разложения этих преобразований в преобразования на подгруппах группы G : чем больше факторизация G, тем меньше требуется операций для выполнения быстрых преобразований [8,12,18].

Итак, пусть $f=(f_1, f_2, \dots, f_N)$ - входная последовательность ($N=\prod_{i=1}^{S} p_i^{\gamma_i}$, где p_i - простые, S, γ_i - натуральные числа, $i=\overline{1,S}$), над которой производится некоторое линейное преобразование

$$y_k = \sum_{j=1}^{N} \beta_{kj} f_j \quad (k=\overline{1,L}) \qquad (10)$$

где суммирование производится в некотором кольце, $\beta_{kj} \in R$ $k = \overline{1,L}$; $j = \overline{1,N}$, что на матричном языке эквивалентно

$$y^T = A f^T \tag{II}$$

где $A = \|\beta_{kj}\|$ $(k = \overline{1,L}; j = \overline{1,N})$; $y = (y_1, y_2, \dots, y_L)$

Линейный алгоритм вычисления (I0) строим следующим образом: пусть $\mathcal{S} = (\tilde{\mathcal{F}}_0^{(1)}, \dots, \tilde{\mathcal{F}}_{m^{(1)}}^{(1)}, \dots, \tilde{\mathcal{F}}_0^{(S)}, \dots, \tilde{\mathcal{F}}_{m^{(S)}}^{(S)})$ - последовательность линейных афинных функций из R^N в R, таких, что $\tilde{\mathcal{F}}_0^{(1)}$ состоит из констант и переменных, $\tilde{\mathcal{F}}_0^{(i+1)} = \tilde{\mathcal{F}}_{m^{(i)}}^{(i)}$, и

$$\tilde{\mathcal{F}}_{j+1}^{(i+1)} = \tilde{\mathcal{F}}_j^{(i+1)} \cup \left\{ \sum_{k=1}^{\rho_{i+1}} \alpha_{jk}^{(i+1)} f_{jk}^{(i+1)} \right\} \tag{I2}$$

где $\alpha_{jk}^{(i+1)} \in R$; $f_{jk}^{(i+1)} \in \tilde{\mathcal{F}}_j^{(i+1)}$

$$j = 0, \dots, m^{(i+1)} - 1; \quad i = 0, \dots, S-1; \quad k = 1, \dots, \rho_{i+1} \tag{I3}$$

Пусть $\tilde{\mathcal{F}}$ - семейство L функций по N переменных над R, чья матрица коэффициентов есть A из (II) и $\mathcal{S}$ -линейный алгоритм, вычисляющий $\tilde{\mathcal{F}}$ ($\tilde{\mathcal{F}} \leqslant \tilde{\mathcal{F}}^{(S)}$) с наименьшим числом шагов m ($m = \sum_{\ell=1}^{S} m^{(\ell)}$), а следовательно, и операций.

Множество функций $\tilde{\mathcal{F}}_{j+1}^{(i+1)}$ образует матрицу $A_{j+1}^{(i+1)}$ размерности $\left(N + \sum_{\ell=0}^{i} m^{(\ell)} + j + 1\right) \times N$), где $m^{(0)} = 0$:

$$A_{j+1}^{(i+1)} = \prod_{\tau=0}^{i+1} \prod_{\ell=0}^{j+1} B_\ell^{(\tau)} = B_{j+1}^{(i+1)} A_j^{(i+1)} \tag{I4}$$

причем

$$B_{j+1}^{(i+1)} = \begin{bmatrix} I_{N + \sum_{\ell=0}^{i} m^{(\ell)} + j} \\ \hline b_j \end{bmatrix}; \quad B_0^{(i+1)} = I_{N + \sum_{\ell=0}^{i} m^{(\ell)}} \tag{I5}$$

$$A_0^{(i+1)} = B_0^{(i+1)} A_{m^{(i)}}^{(i)}; \quad i = \overline{1,S} \tag{I6}$$

где $I_{N + \sum_{\ell=0}^{i} m^{(\ell)} + j}$ - единичная матрица указанного в индексе порядка, b_j -вектор-строка той же длины, имеющая в ρ_{i+1} позициях элементы из множества $\{\alpha_{jk}^{(i+1)}\}$, а в остальных - нули. Таким образом, (I2) означает, что матрица $A_{j+1}^{(i+1)}$ получается из $A_j^{(i+1)}$ путем добавления к ней одной строки

$\varphi_{j+1}^{(i+1)} = \sum_{\kappa=1}^{\rho_{i+1}} \alpha_{j\kappa}' f_{j\kappa}^{(i+1)}$, где $f_{j\kappa}^{(i+1)}$ - строки из $A_j^{(i+1)}$, $\alpha_{j\kappa}' \in \{\alpha_{j\kappa}^{(i+1)}\}$; i,j,κ из (I3).

Линейный алгоритм на матричном языке означает выполнение следующего преобразования

$$y'^T = A_{m^{(s)}}^{(s)} f^T$$

где $A_{m^{(s)}}^{(s)}$ факторизуется по (I4): $A_{m^{(s)}}^{(s)} = \prod_{\tau=0}^{s} \prod_{\ell=0}^{m^{(s)}} B_\ell^{(\tau+1)}$ используя далее (I5) и (I6); y' - расширенный вектор y (см. (I2)) длины $T = (N + \sum_{i=1}^{s} m^{(i)})$; f - начальный вектор $f = (f_1, f_2, \dots, f_N)$; $A_{m^{(s)}}^{(s)}$ есть $(T \times N)$-матрица, содержащая в себе матрицу A из (II).

Пусть $\Delta(A) = \max|\mathcal{D}|$, где $\mathcal{D}$ - детерминант любой квадратной подматрицы A. Имеет место

Утверждение 2. Если на каждом шаге алгоритма $\mathcal{S}$ имеет место $|\alpha_{j\kappa}^{(i)}| \le c_i$, $c_i > \frac{1}{\rho_i}$, $\rho = \max(\rho_i)$, $c > \frac{1}{\rho}$, $c\rho \ge c_i \rho_i$, для каждого $i = \overline{1,s}$; $j = \overline{0,m^{(i)}}$, $\kappa = \overline{1,\rho_i}$, то общее число операций t для вычисления $\mathcal{F}$:

$$t \ge (2\rho_s - 1) \log_{\rho c} |\Delta(A)| \qquad (I7)$$

Доказательство. Идея доказательства состоит в следующем:

I) показывается для каждого $i = \overline{1,s}$, что число шагов $m^{(i)}$ при вычислении $\mathcal{F}^{(i)}$ по (I2) оценивается

$$m^{(i)} \ge \log_{\rho_i c_i} |\Delta(A_{m^{(i)}}^{(i)})|$$

2) оценивается общее число операций для вычисления $\mathcal{F}_i$. Так как на каждом шаге выполняются по ρ_i и $\rho_i - 1$ умножений и сложений соответственно, то

$$t^{(i)} \ge (2\rho_i - 1) \log_{\rho_i c_i} |\Delta(A_{m^{(i)}}^{(i)})|, \qquad (I8)$$

откуда и получаем (I7).

Замечание 5. Полученная в утверждении 2 нижняя оценка сложности вычисления является асимптотической, поскольку в вычислениях алгоритма по (I2) учитываются всевозможные сложения и умножения, т.е. эта оценка не зависит от вида элементов $\{\alpha_{j\kappa}^{(i+1)}\}$ с i,j,κ из (I3).

Замечание 6. В случае $N = 2^z$ $(\rho_i = 2; i = 1)$ оценка общего числа шагов (совпадающая с нижней границей для числа опера-

ций сложения) совпадает с результатом Моргенштерна

$$m^{+} \geq \frac{\log_2 \Delta(A)}{\log_2(2c)} \qquad \text{(см. [I7])}$$

Пусть теперь $f(g): G \to R$, где G – конечная абелева группа, R – коммутативное кольцо с единицей. Поскольку G разложима в прямую сумму примарных подгрупп [II], то есть $G = \sum_{i=1}^{S} G_i$, где $|G_i| = p_i^{z_i}$, то возможно представление $f(g)$ в виде $f(g_1, g_2, \dots, g_S)$, $g_i \in G_i \ (i = \overline{1,S})$. В определении (3) дано понятие Фурье-подобного ДОП вектора-функции f на G. Поскольку $G^{\times}$ изоморфна G, то ее элементы – базис $H(g'+g'') = H(g')H(g'')$, где $g' = (g_1, \dots, g_i, 0, \dots, 0)$; $g'' = (0, \dots, 0, g_{i+1}, \dots, g_S)$, $i = \overline{1,S}$.

Таким образом

$$F_G(H) = \sum_{g \in G} f(g)H(g) = \sum_{g_1 \in G_1} \cdots \sum_{g_S \in G_S} f(g_1, \dots, g_S) H(g_1, \dots, g_S) =$$

$$= \sum_{g_1} \cdots \sum_{g_{S-1}} H(g_1, \dots, g_{S-1}, 0) \sum_{g_S} H(0, \dots, 0, g_S) f(g_1, \dots, g_S) \qquad (19)$$

$$= \sum_{g_1} \cdots \sum_{g_{S-1}} H(g_1, \dots, g_{S-2}, 0, 0) H(0, \dots, 0, g_{S-1}, 0) F_{G_S}(H_1, \dots, H_{S-1}) =$$

$$= F_{G_1}(\Lambda_1(F_{G_2}(\Lambda_2 \dots (\Lambda_{S-1}(F_{G_S}(H_S)H_{S-1})) \dots (H_1)),$$

где $\Lambda_K \ (K = \overline{1,S-1})$ определяют некоторые диагональные матрицы, и преобразование на группе G сводится к последовательному выполнению преобразований на подгруппах $G_i \ (i = \overline{1,S})$

Если теперь F_{G_i} образует матрицу $A_{G_i}^{(i)}$ для фиксированного $i = \overline{1,S}$, то для нахождения нижней границы числа операций, необходимых для вычисления (I9), можно использовать приведенный выше линейный алгоритм $\mathcal{S}$ (модифицированный алгоритм Моргенштерна).

Утверждение 3. Для вычисления (I9) необходимо

$$t \geq \sum_{i=1}^{S} \frac{N}{p_i^{z_i}} \log_{p_i c_i} |\Delta(A_{G_i})|^{2p_i - 1} \qquad (20)$$

операций, если выполняются условия утверждения 2.

Доказательство аналогично доказательству утверждения 2 до получения оценки (I9), откуда и имеем (20).

Теорема. Существует асимптотически быстрый алгоритм Фу-

рье-подобного ДОП по базисным функциям ОВФ на конечной абелевой группе, требующий

$$N\sum_{i=1}^{n}\frac{2p_i-1}{2}m_i \le K(G) \le N\sum_{i=1}^{n}[(p_i+1)m_i-1]+ \quad (12)$$
$$+O\left(\sum_{i=1}^{n}p_i^{m_i}\right)$$

операций.

Доказательство. Оценку верхней границы сложности получаем по (19), а нижней – по модифицированному алгоритму Моргенштерна (утверждение 2) и поскольку $\Delta(A_{G_i})=(p_i^{m_i})p_i^{m_i/2}$ Доказательство (21) означает, что алгоритм (19) является асимптотически быстрым.

Таблица

	№№	G	R	Фурье-подобные ДОП	Базис функций	G'	G^x
	1	2	3	4	5	6	7
ж	1	G_1	R_1	ДПФ	ДЭФ	+	+
ж	2	G_2	R_1	Уолша (У)	УФ	+	+
ж	3	G_3	R_1	Виленкина-Крестенсона (ВК)	ВКФ	+	+
	4	G_4	R_1	Виленкина-Понтрягина (ВП)	ВПФ	+	+
жж	5	G_2	R_1	Обобщенные Уолша (ОУ)	ОУФ	–	+
жж	6	G_3	R_1	Обобщенные ВК (ОВК)	ОВКФ	–	+
жж	7	G_4	R_1	Обобщенные ВП (ОВП)	ОВПФ	–	+
жж	8-9	G_5, G_6	R_2	ТЧП Ферма(ТЧП Ф)	ФФ	+	+
ж	10-11	G_7, G_8	R_3	ТЧП Мерсенна (ТЧП М)	МФ	+	+
ж	12-14	$G_9, G_{10}; G_5$	R_4, R_5, R_6	Псевдо ТЧП Ф (Пс ТЧП Ф)	ПсФФ	+	+
ж	15	G_{11}	R_7	Псевдо ТЧП М (Пс ТЧП М)	ПсМФ	+	+

	1	2	3	4	5	6	7
	16-17	$C_5^к, C_6^к$	R_2	ТЧП ВК Ф	ВКФФ	+	+
	18-19	$C_7^к, C_8^к$	R_3	ТЧП ВК М	ВКМФ	+	+
	20-22	$C_9^к, C_{10}^к, C_5^к$	R_4, R_5, R_6	Пс ТЧП ВК Ф	ПсВКФФ	+	+
	23	$C_{11}^к$	R_7	Пс ТЧП ВК М	ПсВКМФ	+	+
	24-25	C_{12}, C_{13}	R_2	ТЧП ВП Ф	ВПФФ	+	+
	26-27	C_{14}, C_{15}	R_3	ТЧП ВП М	ВПМФ	+	+
	28-30	C_{16}, C_{17}, C_{12}	R_4, R_5, R_6	Пс ТЧП ВП Ф	ПсВПФФ	+	+
	31	C_{18}	R_7	Пс ТЧП ВП М	ПсВПМФ	+	+
**	32-33	$C_5^к, C_6^к$	R_2	ТЧП ОВК Ф	ОВКФФ	–	+
**	34-35	$C_7^к, C_8^к$	R_3	ТЧП ОВК М	ОВКМФ	–	+
**	36-38	$C_9^к, C_{10}^к, C_5^к$	R_4, R_5, R_6	Пс ТЧП ОВК Ф	ПсОВКФФ	–	+
**	39	$C_{11}^к$	R_7	Пс ТЧП ОВК М	ПсОВКМФ	–	+
**	40-41	C_{12}, C_{13}	R_2	ТЧП ОВП Ф	ОВПФФ	–	+
**	42-43	C_{14}, C_{15}	R_3	ТЧП ОВП М	ОВПМФ	–	+
**	44-46	C_{16}, C_{17}, C_{12}	R_4, R_5, R_6	Пс ТЧП ОВП Ф	ПсОВПФФ	–	+
**	47	C_{18}	R_7	Пс ТЧП ОВП М	ПсОВПМФ	–	+
*	48-49	C_{19}, C_6	R_8	Комплексные ТЧП Ф(КТЧП Ф)	КФФ	+	+
*	50-51	C_{20}, C_{21}	R_9	КТЧП М	КМФ	+	+
*	52-53	C_{22}, C_{23}	R_{10}, R_{11}	К Пс ТЧП Ф	КПсФФ	+	+
*	54	C_{24}	R_{12}	К Пс ТЧП М	КПсМФ	+	+
*	55	C_{25}	R_{13}	Эйзенштейна ТЧП Ф(Э ТЧП Ф)	ЭФФ	+	+
*	56	C_{26}	R_{14}	К Э ТЧП Ф	КЭФФ	+	+
	57-58	$C_{19}^к, C_6^к$	R_8	КТЧП ВК Ф	КВКФФ	+	+
	59-60	$C_{20}^к, C_{21}^к$	R_9	КТЧП ВК М	КВКМФ	+	+
	61-62	$C_{22}^к, C_{23}^к$	R_{10}, R_{11}	К Пс ТЧП ВК Ф	КПсВКФФ	+	+

	I	2	3	4	5	6	7
	63	G_{24}^{K}	R_{12}	К Пс ТЧП ВК М	КПсВКМФ	+	+
	64	G_{25}^{K}	R_{13}	Э ТЧП ВК Ф	ЭВКФФ	+	+
	65	G_{26}^{K}	R_{14}	К Э ТЧП ВК Ф	КЭВКФФ	+	+
	66-67	G_{27}, G_{13}	R_8	КТЧП ВП Ф	КВПФФ	+	+
	68-69	G_{28}, G_{29}	R_9	КТЧП ВП М	КВПМФ	+	+
	70-7I	G_{30}, G_{31}	R_{10}, R_{11}	К Пс ТЧП ВП Ф	КПсВПФФ	+	+
	72	G_{32}	R_{12}	К Пс ТЧП ВП М	КПсВПМФ	+	+
	73	G_{33}	R_{13}	Э ТЧП ВП Ф	ЭВПФФ	+	+
	74	G_{34}	R_{14}	К Э ТЧП ВП Ф	КЭВПФФ	+	+
**	75-76	G_{19}^{K}, G_{6}^{K}	R_8	КТЧП ОВК Ф	КОВКФФ	-	+
**	77-78	G_{20}^{K}, G_{21}^{K}	R_9	КТЧП ОВК М	КОВКМФ	-	+
**	79-80	G_{22}^{K}, G_{23}^{K}	R_{10}, R_{11}	К Пс ТЧП ОВК Ф	КПсОВКФФ	-	+
**	8I	G_{24}^{K}	R_{12}	К Пс ТЧП ОВК М	КПсОВКМФ	-	+
**	82	G_{25}^{K}	R_{13}	Э ТЧП ОВК Ф	ЭОВКФФ	-	+
**	83	G_{26}^{K}	R_{14}	К Э ТЧП ОВК Ф	КЭОВКФФ	-	+
**	84-85	G_{27}, G_{13}	R_8	КТЧП ОВП Ф	КОВП ФФ	-	+
**	86-87	G_{28}, G_{29}	R_9	КТЧП ОВП М	КОВПМФ	-	+
**	88-89	G_{30}, G_{31}	R_{10}, R_{11}	К Пс ТЧП ОВП Ф	КПсОВПФФ	-	+
**	90	G_{32}	R_{12}	К Пс ТЧП ОВП М	КПсОВПМФ	-	+
**	9I	G_{33}	R_{13}	Э ТЧП ОВП Ф	ЭОВПФФ	-	+
**	92	G_{34}	R_{14}	К Э ТЧП ОВП Ф	КЭОВПФФ	-	+
*	93-95	G_7	R_{15}	Полиномиальные (П)	ПФ	+	+
	96-98	G_8, G_{11}	R_{15}, R_{16}	ВК П	ВК ПФ	+	+
	99-I0I	G_{14}, G_{15}, G_{18}	R_{15}, R_{16}	ВП П	ВП ПФ	+	+
**	I02-I04	$G_7^{K}, G_8^{K}, G_{11}^{K}$	R_{15}, R_{16}	ОВК П	ОВК ПФ	-	+
**	I05-I07	G_{14}, G_{15}, G_{18}	R_{15}, R_{16}	ОВП П	ОВП ПФ	-	+

Список сокращений и обозначений

ДПФ – дискретное преобразование Фурье,
ТЧП – теоретико-числовое преобразование,
(К) ((Пс) или (Э)) ТЧП (О)ВК(или П) Ф (или М) – (комплексные) (псевдо)или(Эйзенштейна)) теоретико-числовые преобразования (обобщенные) Виленкина-Крестенсона (или Понтрягина) Ферма (или Мерсенна),
(К) ((Пс)или (Э))(О)ВК(или П)Ф(или М)Ф-(комплексные)((псевдо) или (Эйзенштейна))(обобщенные)Виленкина-Крестенсона (или Понтрягина)Ферма(или Мерсенна) функции,
(О)ВК(или П) П(Ф)- обобщенные Виленкина-Крестенсона (или Понтрягина) полиномиальные (функции).

$$G_1 = Z_N\ (N>1);\ G_2 = Z_2^n;\ G_3 = Z_q^n;\ G_4 = \sum_{i=1}^{K} Z_{q_i}^{n_i}\ (q_i \neq q_j,\ i \neq j);$$

$$G_5 = Z_N\ (N = 2^{d+1},\ d \geq 0);\ G_6 = Z_N\ (N = 2^{d+2},\ d \geq 2);\ G_7 = Z_p;$$

$$G_8 = Z_{2p}\ (p>2);\ G_9 = Z_{2p}\ (p>3);\ G_{10} = Z_{2p^2}\ (p>3);\ G_{11} = Z_{p^2}$$

$$G_{12} = \sum_{j=1}^{K} Z_{N_j}^{l_j}\ (N_j = 2^{d_j+1});\ G_{13} = \sum_{j=1}^{K} Z_{N_j}^{l_j}\ (N_j = 2^{d_j+2});\ G_{14} = \sum_{j=1}^{K} Z_{p_j}^{l_j};$$

$$G_{15} = \sum_{j=1}^{K} Z_{2p_j}^{l_j}\ (p_j>2);\ G_{16} = \sum_{j=1}^{K} Z_{2p_j}^{l_j}\ (p_j>3);\ G_{17} = \sum_{j=1}^{K} Z_{2p_j^2}^{l_j}\ (p_j>3);$$

$$G_{18} = \sum_{j=1}^{K} Z_{p_j^2}^{l_j};\ G_{19} = Z_N\ (N = 2^{d+1}\ d>1);\ G_{20} = Z_{4p}\ (p>2);$$

$$G_{21} = Z_{8p}\ (p>2);\ G_{22} = Z_{8p}\ (p>3),\ G_{23} = Z_{8p^2}\ (p>3);\ G_{24} = Z_{8p^2}\ (p>2);$$

$$G_{25} = Z_N\ (N = 3\cdot 2^{d+1},\ d>1);\ G_{26} = Z_N\ (N = 3\cdot 2^{d+2},\ d>1);$$

$$G_{27} = \sum_{j=1}^{K} Z_{N_j}^{l_j}\ (N_j = 2^{d_j+1},\ d_j>1);\ G_{28} = \sum_{j=1}^{K} Z_{4p_j}^{l_j}\ (p_j>2);$$

$$G_{29} = \sum_{j=1}^{K} Z_{8p_j}^{l_j}\ (p_i>2);\ G_{30} = \sum_{j=1}^{K} Z_{8p_j}^{l_j}\ (p_j>3);\ G_{31} = \sum_{j=1}^{K} Z_{8p_j^2}^{l_j}\ (p_j>3);$$

$$G_{32} = \sum_{j=1}^{K} Z_{8p_j^2}^{l_j}\ (p_j>2);\ G_{33} = \sum_{j=1}^{K} Z_{N_j}^{l_j}\ (N_j = 3\cdot 2^{d_j+1},\ d_j>1);$$

$$G_{34} = \sum_{j=1}^{K} Z_{N_j}^{l_j}\ (N_j = 3\cdot 2^{d_j+2},\ d_j>1);\quad j = \overline{1,K}$$

$R_1 = C$; $R_2 = Z_{F_d}$ $(F_d = 2^{2^d} + 1)$; $R_3 = Z_{M_p}$ $(M_p = 2^p - 1)$;

$R_4 = Z_{m_1}$ $(m_1 = (2^p + 1)/3)$; $R_5 = Z_{m_2}$ $(m_2 = (2^{p^2} + 1)/(2^p + 1))$;

$R_6 = Z_{m_3}$ $(m_3 = 2^{s2^d} + 1,\ s \geq 2)$; $R_7 = Z_{m_4}$ $(m_4 = (2^{p^2} - 1)/(2^p - 1))$;

$R_8 = R_2[i]$; $R_9 = R_3[i]$; $R_{10} = R_4[i]$; $R_{11} = R_5[i]$; $R_{12} = R_7[i]$

$R_{13} = R_2[\Theta]$; $R_{14} = R_2[i\Theta]$; $R_{15} = Q(x)/(m_5)$ $\left(m_5(x) = \frac{x^p - 1}{x - 1}\right)$;

$R_{16} = Q(x)/(m_6)$ $\left(m_6(x) = \frac{x^{p^2} - 1}{x^p - 1}\right)$.

Литература

I. Агаян С.С., Егиазарян К.О. Обобщенные матрицы Адамара.- В сб.: Математические вопросы кибернетики и вычислительной техники. XII, Ереван, 1984, с.51-88.

2. Агаян С.С., Матевосян А.К. Быстрое преобразование Адамара. - В сб.: Математические вопросы кибернетики и вычислительной техники. XI, Ереван, 1982, с.73-90.

3. Айзенберг Н.Н., Поливко В.П., Семирот М.С. Об обратной задаче Карунена-Лоэва. - Радиотехника и электроника, 1981,№4, с.777-782.

4. Ахмед Н., Рао К. Ортогональные преобразования при обработке цифровых сигналов. М.: Связь,1980.

5. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М.: Мир, 1979.

6. Белоглазова О.В., Лабунец В.Г. Теория и применение преобразований Гаусса. - В сб.: Синтез управляющих и вычислительных систем. Свердловск, 1980, с.25-40.

7. Берман С.Д., Грушко И.И. К теории обработки дискретных сигналов. - Проблемы передачи информации, 1983, т.XIX, вып.4, с.43-49.

8. Бойко Л.Л. Обобщенные преобразования Фурье-Хаара на конечной абелевой группе. - В сб.: Цифровая обработка сигналов и ее применение. М.: Наука, 1981, с.12-22.

9. Карповский М.Г., Москалев Э.С. Спектральные методы анализа и синтеза дискретных устройств.Л.: Энергия. 1973.

I0. Кренкель Т.Э. Спектральный анализ на конечных коммутативных группах. - Радиотехника, I975, № 6, с.I9-23.

II. Курош А.Г. Теория групп. М., Л.: I944.

I2. Лабунец В.Г., Ситников О.П. Обобщенные и быстрые преобразования Фурье на произвольной конечной абелевой группе.- В сб.: Гармонический анализ на группах в абстрактной теории систем. Свердловск, I976, с.24-43.

I3. Нуссбаумер Г. Быстрое преобразование Фурье и алгоритмы вычисления сверток. М.: Радио и связь, I985.

I4. Трахтман А.М., Трахтман В.А. Основы теории дискретных сигналов на конечных интервалах. М.: Сов.радио,I975.

I5. Cooley J.M., Tukey J.W. An algorithm for the machine calculation of complex Fourier series. - Math. Comput., 1965, vol. 19, p. 297 - 301.

I6. Matevosyan A.K. The inverse problem of Karhunen-Loeve. Proc. of the Seventh European Meeting on Cybernetics and System Research. Ed. by R. Trappl, 1984.

I7. Morgenstern J. Note on a lower bound of the linear complexity of the fast Fourier transform. - J. of the ACM, 1973, vol. 20, No. 2, p. 305 - 306.

I8. Nicholson P.J. Algebraic theory of finite Fourier transforms. - J. of Computer and System Sciences, 1971, vol. 5, p. 524 - 547.

I9. Creutzburg R., Tasche M. F - Transformation und Faltung in kommutativen Ringen. - Elektronische Informationsverarbeitung und Kibernetik, 1985, vol. 21, No. 3, p.129 - 150.

20. Агаян С.С., Егиазарян К.О. Дискретные ортогональные преобразования на конечной коммутативной группе. - Рукопись деп. в АрмНИИНТИ от 2I.II.84 № I7 Ар-Д84,Ереван,I984.

A new class of transformations in the theory of processing discrete signals

S.S. Agajan, K.O. Egiazarjan

Summary

A new class of Fourier-like discrete orthogonal transformations is introduced. The transformations work on the set of functions having domain in a finite Abelian group and range in a commutative ring with unity. They are defined using the system of basic functions - the generalized Vilenkin-functions - containing some well known Fourier -type functions (like: discrete exponential functions, the functions of Vilenkin-Krestenson, the functions of Vilenkin-Pontrjagin). Some estimations of the complexity of computing discrete orthogonal transformations as well as some asymptotically fast algorithms for computing the transformations are given.

MTA SZTAKI Tanulmányok 202/1987 pp. 25-38

МЕТОДЫ ОПТИМИЗАЦИИ ДИАЛОГОВЫХ СИСТЕМ ЭКСПЕРИМЕНТАЛЬНЫХ ИССЛЕДОВАНИЙ

А.А.Аракелян

В программном докладе [I] , посвященном анализу перспектив развития систем автоматизации научных исследований(САНИ) отмечается, что успешная реализация основных направлений работ связана с проблемой создания "комплексной программы оптимизации блоков, узлов и целых систем оптимизации". Однако, [2-7] известно, что решение проблемы оптимального синтеза диалоговых систем экспериментальных исследований (ДСЭИ) в общем виде представляет собой практически неразрешимую задачу, так как она связана с необходимостью рассмотрения разнообразных групп частных критериев.

Данный доклад является развитием работ [8-I9]. Он посвящен формулированию и изучению принципов оптимальности и их реализации для некоторых задач оптимизации ДСЭИ, связанных с

а) оптимальным выбором варианта ДСЭИ [2-6,8];

б) организацией систем сбора статистики о работе САНИ [I,9,I0];

в) созданием трансляторов с алгоритмических языков высокого уровня [I,II];

г) созданием математического обеспечения ДСЭИ [I,I2,I3];

д) развитием методов математического обеспечения параллельного решения задач в многопроцессорных вычислительных системах [II-I3];

е) разработкой высокоуровневых языков описания специализированных систем [I] ;

ж) разработкой методов диагностирования ДСЭИ, объектов контроля (ОК) и методов и средств оптимизации их диагностического обеспечения [I,I3-I5].

I. Формализованная алгоритмическая модель ДСЭИ. Под формализованной алгоритмической моделью (ФАМ) ДСЭИ будем пони-

мать совокупность схем алгоритмов, программ и правил, позволяющих описать последовательность действий, направленных на реализацию процесса измерения, контроля и восстановления ОК.

ДСЭИ должна обеспечить возможность выполнения следующих обобщенных функций:

а) осуществления допускового количественного контроля и проведения на его основе классифицирования ОК по принципу "Функционирует - не функционирует" или принадлежности одному из возможных состояний [II];

б) диагностирования неисправностей аппаратурных и программных средств [I,I4,I5];

в) прогнозирования на основе сбора информации об отказах, возможных сбоев и отказов в оборудовании и системе программно-математического обеспечения (СПМО([I,I0];

г) восстановления состояния ОК посредством уменьшения разброса контролируемых параметров относительно их номинальных значений.

Опыт реализации, внедрения и эксплуатации ДСЭИ обуславливает необходимость выполнения свойств, связанных с системологическими особенностями [I,8-I5], а именно:

а) обучаемость, совершенствование и развитие при помощи разработки и программной реализации новых алгоритмов функционирования, измерения и контроля, проблемно-ориентированных языков (ПОЯ) программирования, трансляторов с этих языков и аппаратурных средств, реализующих разработанные программные модели [I,II-I3,I6,I7];

б) эффективность и оптимальность функционирования посредством выбора оптимальных диалоговых процедур [I,I6];

в) "выживаемость", т.е. простоту адаптации к изменяющимся аппаратурным и программным средствам [I,I3];

г) принцип модульности, т.е. относительной независимости компонентов, составляющих аппаратурные и программные средства [I,9,II-I3];

д) проведения самоконтроля, т.е. аппаратно-программного контроля правильности функционирования ДСЭИ [I2,I3] и диагностики ее неисправностей.

ДСЭИ, которая обеспечивает возможность выполнения перечисленных требований, относится к сложным человеко-машинным системам (ЧМС), позволяющим осуществление диалога с пользователем, развитие и обучение посредством создания новых аппаратурно-программных средств и выбор оптимальных диалоговых процедур.

Следуя [1], такие ДСЭИ можно отнести к классу программно-кибернетических систем, имеющих возможность самообучения и приспособления посредством создания новых аппаратурно-программных средств к изменяющейся внешней среде. При этом, изменение внешней среды может быть обусловлено как уровнем знаний и умений пользователей, так и техническими данными ОК.

В соответствии с [5,6] , разделим ДСЭИ на две независимые части: подсистему выдачи стимулирующих воздействий и подсистему преобразования и анализа реакции.

Через $x_1, x_2, \ldots, x_n$ обозначим совокупность параметров ОК, а через $X_1, X_2, \ldots, X_n$ множества принимаемых ими значений. Совокупность множеств сигналов на входах и выходах преобразователей ДСЭИ на различных этапах преобразования в процессе осуществления проверки и управления ОК обозначим через $B_0, B_1, \ldots, B_m$. При этом, B_0 - множество сигналов преобразователей, выполняющих непосредственное преобразование информации от ОК, а остальные $B_j, 1 \le j \le m$, сигналы соответствуют преобразователям на последующих этапах проведения контроля и управления. Обозначим через f_{ij} отображения множества сигналов B_i во множество сигналов B_j. Положим $B_{m+1} = \bigcup_{i=1}^{n} X_i$, определим на множествах B_i, $i=1,2,\ldots,m+1$ семейства отношений R_i. Обозначим через Q СПМО ДСЭИ. Тогда системой будем называть семейство отношений $\{R_i\}_{i=1,2,\ldots,m+1}$, которые согласовывают значения переменных из B_i со значениями переменных из B_j для выполнения множества целей C .Следовательно, ДСЭИ есть

$$S = \langle X, \{B_i\}, \{R_i\}, \{f_{ij}\}, Q, C \rangle \qquad (I)$$

Для расширения и уточнения определения ДСЭИ (I) рассмотрим дополнительные параметры, обозначив через τ_u количественные характеристики элементов системы преобразователей. В

качестве количественных характеристик можно рассматривать такие величины, как достоверность и время преобразования сигналов из множества B_i во множество B_j. ДСЭИ, будучи сложной технической человеко-машинной системой, представляющей собой совокупность взаимосвязанных и взаимодействующих элементов, функционирование которых направлено на выполнение определенных целей, обусловливает введение параметра, характеризующего время достижения этих целей. Обозначим через t - время контроля, а через $T = \bigcup_{i,j} t_{ij}$ множество, состоящее из временных характеристик, определяющих время t_{ij} преобразования сигналов из множества B_i во множество B_j. Пусть M - множество объектов контроля, считая и перспективные, которые необходимо обслужить, применяя ДСЭИ. Обозначим через $\{\theta_i\}_{i=1,2,\dots,m}$ множество требований, которым должна отвечать ДСЭИ, обеспечивающая возможность контроля объектов множества M. Тогда ДСЭИ можно представить в виде

$$S = \langle X, \{B_i\}, \{R_i\}, \{t_{ij}\}, Q, f, \{\tau_u\}, T, C, \{\theta_i\} \rangle \qquad (2)$$

2. Моделирование и оптимизация ДСЭИ в условиях неопределенности. В [5,6] рассматривается следующая задача определения состава модулей ДСЭИ. Пусть имеется m типов ОК, которые необходимо проверить, применяя ДСЭИ. Для каждого типа i ОК известны требования θ_i, которые предъявляются к ДСЭИ и являются n_i -мерными вещественными векторами, $i=1,2,\dots,m$. Так как требования, предъявляемые к ДСЭИ перспективными ОК не всегда известны и зависят от неопределенных факторов, то последовательность $\theta=\{\theta_i\}$ представляет собой последовательность случайных чисел. Предположим, что функция распределения $x(\theta)$ этой последовательности не вполне известна, но известно множество $\tilde{X}$ функций распределения, которому оно принадлежит.

Процесс выбора рационального варианта ДСЭИ состоит в следующем. На первом шаге рассматривается система требований $\theta^1 \subset \theta$, для которой выбирается вариант ДСЭИ S^1 и проверяется ее удовлетворение системе требований θ. Если вариант S^1 не удовлетворяет, то выбирается вариант S^2 и т.д. Если выбор ДСЭИ заканчивается на K -ом шаге, то мы будем говорить, что

выбор осуществляется за k шагов.

Обозначим через $d(\theta;\theta^1,\dots,\theta^k)$ и $y(\theta;\theta^1,\dots,\theta^k)$ нерандомизированную и рандомизированную решающие функции соответственно. Пусть $H(x,y)$ будет вектор-функцией, определенной на $X\times Y$, где Y - множество всех рандомизированных решающих функций.

Будем предполагать, что $H(x,y)$ выражает потери, связанные с удовлетворением требований по объему, безотказности функционирования, точности преобразования и передачи информации [2,5,6] и т.д.

Тогда степень предпочтения выбора $x\in X$ при известном может быть выражена следующим образом:

$$H_j(x,y)\le H_j(x',y') \quad \text{для любого } y'\in Y$$

Положим

$$H(\xi,\eta)=\int_X\int_Y H(x,y)\,d\xi\,d\eta,$$

где ξ, η - вероятностные распределения на X,Y соответственно и положим

$$\begin{aligned}
&V(\xi,\eta)=\{(\xi',\eta')\mid H(\xi',\eta')\le H(\xi,\eta)\},\ V^1(\xi)=\bigcap_\eta V(\xi,\eta),\\
&V^1=\bigcup_\xi V^1(\xi),\ V^2(\eta)=\bigcap_\xi V^2(\xi,\eta),\ V^2=\bigcup_\eta V^2(\eta),\\
&U^2(\eta)=\bigcup_\xi V(\xi,\eta),\ U^2=\bigcap_\eta U^2(\eta),\\
&U^1(\xi)=\bigcup_\eta V(\xi,\eta),\ U^1=\bigcap_\xi U^1(\xi).
\end{aligned}\tag{3}$$

Определим отношения

$$(\xi,\xi')\in\rho \leftrightarrow V^1(\xi')\subseteq V^1(\xi),\ (\eta,\eta')\in\sigma \leftrightarrow V^2(\eta')\subseteq V^2(\eta) \tag{4}$$

Стратегия ξ^* называется максимальной, если не существует такой $\xi\in X$, что $(\xi,\xi^*)\in\rho$.

Стратегия ξ^* называется оптимальной, если

$$\xi^*\in\{\xi'\mid V^1(\xi')=V^1\}. \tag{5}$$

В докладе получены условия существования стратегий, удовлетворяющих (5).

3. Оптимальный выбор структуры диалога ДСЭИ в условиях

неопределенности. Одна из задач построения диалогового обеспечения в системах "человек-машина" состоит в выборе оптимальной структуры диалога [I,I6]. В настоящее время достаточно хорошо изучены методы выбора оптимальной структуры диалога в предположении, что характеристики ДСЭИ, влияющие на эффективность диалога, известны и не изменяются во времени. Однако, как показывает опыт, эксплуатация сложных человеко-машинных систем тесно связана с вопросами адаптации структуры диалога к изменяющимся характеристикам системы [I,I6]. Такими характеристиками, например, могут быть квалификация специалистов, степень обеспеченности машинным временем, алгоритмическая сложность решаемых задач и т.д.

Перечисленные обстоятельства приводят к необходимости создания диалоговых систем, адаптирующихся к возможным изменениям характеристик системы "человек-машина" в области их неопределенности.

Сформулируем проблему математически. Пусть человеко-машинная система допускает функционирование в m диалоговых режимах. Обозначим их множество через X , а через Y множество пользователей, для которых известны характеристики функционирования системы "человек-ЭВМ", а именно: для каждой пары $(x,y)\in X\times Y$ известна функция эффективности $H(x,y)$. При этом $H(x,y)$ может быть как скалярной функцией, так и вектор-функцией размерности n . Последнее обусловлено тем, что на эффективность функционирования диалоговой системы одновременно оказывают влияние два и более фактора.

Таким образом, получаем следующую игру с векторной функцией выигрыша:

$$\Gamma = \langle X, Y, H(x,y) \rangle \qquad (6)$$

Для выбора оптимальной структуры диалога воспользуемся понятием обобщенного гарантированного выигрыша игры с векторной функцией выигрыша.

Определим для игры Γ (6) смешанную игру

$$\Gamma = \langle \widetilde{X}, \widetilde{Y}, H(\xi,\eta) \rangle \qquad (7)$$

множества (3) и отношения (4).

Тогда смешанные стратегии можно интерпретировать следующим образом. Вместо выбора конкретного диалога $x \in X$ выбирается вероятностная мера $\tilde{x}$, определенная на σ-алгебре $\mathcal{X}$ подмножеств множества X, и диалог x выбирается при помощи случайного устройства, сконструированного так, что для произвольного $X' \in \mathcal{X}$ вероятность того, что выбранный диалог $x \in X'$ равен $\xi(X')$.

В докладе приводятся условия существования, удовлетворяющей (5), смешанной стратегии для выбора диалога.

4. Оптимизация диагностического обеспечения ДСЭИ.

Вопросами построения тестов функционального контроля (ТФК) дискретных устройств (ДУ) посвящен ряд работ [14,15]. В этих работах предполагается, что между разработками ТФК существует согласованность в выборе тестов. Однако, в действительности может оказаться, что разработчики ТФК действуют независимо друг от друга, т.е. для каждого разработчика тесты, имеющиеся в распоряжении остальных, представляют собой неопределенные факторы.

В настоящем параграфе рассматривается вопрос построения полного множества ТФК микропроцессорных (МП) БИС при наличии неопределенных факторов, влияющих на процесс их построения. Целесообразность такого подхода объясняется возросшей сложностью и повышением степени интеграции МП БИС.

Предположим, что в МП БИС выделены n функционально законченных узлов (ФЗУ) и $I = \{1,2,\ldots,n\}$ множество разработчиков, проектирующих ТФК. При этом i-й разработчик занят проектированием ТФК i-го ФЗУ. Обозначим через X_i множество ТФК, находящееся в распоряжении i-го разработчика, $i \in I$. Процесс построения ТФК осуществляется следующим образом: i-й разработчик выбирает тест $x_i \in X_i$, $i \in I$ и таким образом получается ситуация $x = (x_1, x_2, \ldots, x_n)$. При этом выбор тестов производится разработчиками независимо друг от друга.

Обозначим [18] через $N_i(x_1, x_2, \ldots, x_n)$-множество неисправностей i-го ФЗУ МП БИС, обнаруживаемых тестами $x_1, x_2, \ldots, x_n$. Пусть N_i - множество неисправностей i-го ФЗУ МП БИС, $i \in I$.

Таким образом, если рассмотреть разработчиков ТФК в качестве игроков, множество ТФК X_i, находящиеся в распоряжении i -о разработчика в качестве стратегий i -о игрока, множество $X = \prod_{i \in I} X_i$ в качестве множества ситуаций и отношение предпочтения ρ_i, определенное следующим образом:

$(x, y) \in \rho_i$, тогда и только тогда, когда $x_i \neq y_i$, $x_j = y_j$, $j \neq i$, $j \in I$, $N_i(y) \subseteq N_i(x)$,

то получим общую бескоалиционную игру, где ρ_i -отношение частичного упорядочения.

Будем говорить, что множество ТФК $X' \subset X$ является полным [18], если

$$N_i = \bigcup_{x \in X'} N_i(x) \quad \text{для любого } i \in I \qquad (8)$$

Так как отношение предпочтения ρ_i является отношением частичного упорядочения, то для него существует кососимметричная функция сравнительной полезности $u_i(x, y)$, т.е. такая функция $u_i(x, y)$, что

$(x, y) \in \rho_i$ тогда и только тогда, когда $u_i(x, y) > 0$.

Определим отношение $\rho \subset X \times X$ следующим образом:

$(x, y) \in \rho$ тогда и только тогда, когда существует такое $i \in I$, что $(x, y) \in \rho_i$.

Подмножество $V_\rho(X)$ называется решением Неймана-Моргенштерна, или короче, НМ-решением, если

1) $(x, y) \notin \rho$ для любых $x, y \in V_\rho(X)$;

2) для любого $x \notin V_\rho(X)$ существует такой $y \in V_\rho(X)$, что $(y, x) \in \rho$.

Пусть множество тестов X_i, находящееся в распоряжении i -о разработчика, удовлетворяет условию:

$$\bigcup_{x_i \in X_i} N_i(x) = N_i \text{ для любых } i \in I \qquad (9)$$

Теорема. $N_i = \bigcup_{x \in V_\rho(X)} N_i(x)$, $i \in I$.

Для микро-ЭВМ из [14] приводится пример ТФК, для которых отношение ρ не является четным. Это обуславливает необходимость рассмотрения вероятностных распределений на множестве ТФК.

Обозначим через $\tilde{X}_i$ вероятностное распределение на Σ -алгебре множества стратегий X_i, $i \in I$. Положим

$$u_i(\tilde{x},\tilde{y}) = \sum_{x,y \in X} u_i(x,y)\tilde{x}(x)\tilde{y}(y).$$

Обозначим через $\tilde{\sigma}_i$ отношение, определенное следующим образом:

$(\tilde{x},\tilde{y}) \in \tilde{\rho}_i$, тогда и только тогда, когда $u_i(\tilde{x},\tilde{y}) \geq 0$.

Тогда игра

$$\tilde{\Gamma} = \langle \{\tilde{X}_i\}_{i\in I}, \{\tilde{\rho}_i\}_{i\in I}, I \rangle \qquad (10)$$

будет смешанным расширением бескоалиционной игры $\Gamma(7)$.

При этом смешанные стратегии можно интерпретировать следующим образом. Вместо выбора конкретного теста $x_i \in X_i$ игрок i выбирает вероятностную меру $\tilde{x}_i$, определенную на Σ - алгебре $\mathcal{X}_i$ подмножеств X_i и тест x_i выбирается при помощи случайного устройства, сконструированного так, что для произвольного $X_i' \in \mathcal{X}_i$ вероятность того, что выбранный тест $x_i \in X_i'$ равен $\tilde{x}_i(X_i')$, $i \in I$.

Теорема. Если множество I конечное, множества X_i компактные подмножества линейных выпуклых топологических пространств, отношения ρ_i ациклические, непрерывные, ρ_i выпуклые, то отношение ρ имеет Н-решение.

Следствие. Для любого такого множества ТФК, что $\tilde{\rho}_i$ ацикличны $i \in I$ существует Н-решение $V_{\tilde{\rho}}(\tilde{X})$.

5. Агрегирование в задачах оптимизации ТФК дискретных устройств [17]. Предположим, что X_0 является множеством тестов функционального контроля проверяемого устройства и обеспечивает возможность проверки некоторого множества неисправностей.

Предположим, что на X_0 задано отношение предпочтения ρ_0, определенное следующим образом:

$(x,y) \in \rho_0$ тогда и только тогда, когда $N(y) \subseteq N(x)$,

где $N(x)$ - множество неисправностей проверяемого устройства, которые обнаруживаются тестом x. Если через N_0 обозначим множество всех неисправностей, то [18]

$$\bigcup_{x \in V_{\rho_0}(X)} N(x) = N_0 \qquad (11)$$

Множество X_0 назовем множеством агрегированных тестов функционального контроля нулевого уровня.

Решить задачу (11) одним разработчиком, часто не пред-

ставляется возможным, так как множества неисправностей и тестов определяются коллективом разработчиков, каждый из которых занят проектированием ТДК отдельного функционально-законченного узла проверяемого устройства.

Одним из способов решения этой проблемы является построение иерархии задач проектирования ТДК дискретных устройств.

Определим агрегированное множество ТДК первого уровня:

$$X_1 \supseteq \{x(1) = \varphi_1(x(0)) \mid x(0) \in X_0\}.$$

Обозначим через Ψ_1 отображение из X_1 в X_0, т.е.

$$X_0 \supseteq \{x(0) = \Psi_1(x(1)) \mid x(1) \in X_1\}.$$

При этом полагаем, что X_1 является множеством ТДК, обеспечивающим возможность проверки множества неисправностей $N_1 \subseteq N_0$. Обозначим через ρ_1 отношение предпочтения, заданное на X_1.

Продолжив процесс агрегирования ТДК дальше, получим следующие выражения

$$X_2 \supseteq \{x(2) = \varphi_2(x(1)) \mid x(1) \in X_1\},$$
$$X_1 \supseteq \{x(1) = \Psi_2(x(2)) \mid x(2) \in X_2\},$$
ρ_2 отношение предпочтения на X_2, $N_2 \subseteq N_1$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$
$$X_{k+1} \supseteq \{x(k+1) = \varphi_{k+1}(x(k)) \mid x(k) \in X_k\},$$
$$X_k \supseteq \{x(k) = \Psi_{k+1}(x(k+1)) \mid x(k+1) \in X_{k+1}\},$$
ρ_{k+1} отношение предпочтения на X_{k+1}, $N_{k+1} \subseteq N_k$,

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$
$$X_m \supseteq \{x(m) = \varphi_m(x(m-1)) \mid x(m-1) \in X_{m-1}\},$$
$$X_{m-1} \supseteq \{x(m-1) = \Psi_m(x(m)) \mid x(m) \in X_m\},$$

ρ_m отношение предпочтения на X_m, $N_m \subseteq N_{m-1}$.

Для выбора шагов агрегирования m предполагаем, чтобы множество N_m было по мощности небольшим, а отношение предпочтения ρ_m несложным. Очевидно, что при рациональном агрегировании можно обеспечить выполнение включений $N_{k+1} \subseteq N_k$ и таким образом, обусловить возможность решения задачи на m-м шаге, т.е. нахождения НМ-решения $V_{\rho_m}(X_m)$.

5.I. Первое уравнение процесса проектирования ТДК.

Пусть выполнены все шаги агрегирования, а именно: определены X_k, ρ_k, $k = 0, 1, \ldots, m$. Тогда задачу проектирования ТДК

можно решить следующим образом.

Определим НМ-решение $V_{\rho_m}(X_m)$ и множество

$$\left\{ x(m-1) \in X_{m-1} \;\middle|\; \begin{matrix} \varphi_m(x(m-1)) = x(m), \; x(m) \in V_{\rho_m}(X_m) \\ \psi_m(x(m)) = x(m-1) \end{matrix} \right\}$$

Это множество есть

$$\bar{X}_{m-1} = \psi_m\left(V_{\rho_m}(X_m)\right).$$

Далее найдем НМ-решение $(\bar{X}_{m-1}, \rho_{m-1})$, т.е. множество

$$V_{\rho_{m-1}}(\bar{X}_{m-1}), \quad \bar{X}_{m-2} = \psi_{m-1}\left(V_{\rho_{m-1}}(\bar{X}_{m-1})\right).$$

Продолжив процесс построения НМ-решений для каждого шага агрегирования, получим НМ-решение $V_{\rho_0}(X_0)$.

Тогда описанный процесс агрегирования можно представить в виде следующего рекуррентного уравнения:

$$V_k = V_{\rho_k}\left(\psi_{k+1}(V_{k+1})\right), \; k = m-1, m-2, \ldots, 1, 0,$$
$$V_m = V_{\rho_m}(X_m). \qquad (12)$$

Под решением задачи построения оптимального множества ТФК будем понимать построение множества V_0.

В докладе приводится метод решения уравнения (12), основанный на результатах из [19].

6. Метод статистических решений в задачах контроля. Методы теории статистических решений нашли широкое применение в технической диагностике [20]. Однако, известные [20] методы решения проблемы определения работоспособности ОК характеризуются тем, что не учитывают наличие многокритериальности при оценке эффективности контроля. Рассмотрим следующую постановку задачи определения работоспособности ОК.

Пусть известно

а) множество состояний $\{S_i\}_{i \in \Delta}$ проверяемой микросхемы, при этом каждое S_i является n_i-мерным вектором, $i \in \Delta$;

б) множество X функций распределения $x(S)$ случайной величины $S = \{S_i\}_{i \in \Delta}$;

в) множество классов $\Omega = \{\omega_j\}$, которым принадлежат, в зависимости от состояния S_i, проверяемые ОК;

г) множество ТФК $\theta = \{\theta_k\}$ проверяемого ОК;

д) множество решающих функций $Y = \{y(s, \theta; \theta_{i1}, \ldots, \theta_{il})\}$, которые можно принять для выбора ТФК и вынесения решения об

окончании контроля или продолжении испытания;

е) функции $H_1(x,y)$, $H_2(x,y)$, выражающие в условиях выбора функции распределения $x(S) \in X$ и решающей функции $y \in Y$ представляют собой соответственно стоимость испытания и объем памяти, которую занимает испытательная программа ОК в оперативном запоминающем устройстве ЭВМ, управляющей ДСЭИ.

Тогда получим игру Г (6). В докладе приводятся алгоритм и условия существования решений (f^*, z^*), удовлетворяющих включениям

$$V^i \subseteq V^i(f^*, z^*) \subseteq U^i,$$

где $V^i, V^i(f^*, z^*), U^i$, определены согласно равенствам (3).

Литература

I. Петров Б.Н., Куклин Г.Н. Программа развития работ в области автоматизации научных исследований. Сб.Вопросы кибернетики. Автоматизация экспериментальных исследований, М., 1979.

2. Виттих В.А., Куклин Г.И. Информационный подход к решению задачи рационального проектирования систем автоматизации экспериментов. Сб. Вопросы кибернетики. Автоматизация экспериментальных исследований. М., 1979.

3. Египко В.М. Организация и проектирование систем автоматизации научно-технических экспериментов. "Наукова думка", Киев, 1978.

4. Виттих В.А., Куклин Г.Н., Томников Г.Н., Цыбатов В.А., Топологическая оптимизация систем сбора информации."Проблема управления и теория информации", т.7, № I, 1978.

5. Пономарев Н.Н., Фрумкин И.С., Гусинский И.С. и др. Автоматическая аппаратура контроля, М.,1975.

6. Долгов В.А., Кабаткин А.С., Сретенский В.Н. Радиоэлектронные автоматические системы контроля, М., 1978.

7. Калявин В.П., Хузин Р.З. Оптимизация элементной базы автоматизированных средств диагностирования. Известия ЛЭТИ, вып.278, 1980.

9. Аракелян А.А., Кочарян Р.А. А СОСД по результатам из-

мерений электрических параметров ИМС. Электронная техника, серия 3. Микроэлектроника, вып.2(I00), I978.

I0. **Arakelian A.H., Agaian S.S. On an algorithm of spectral analysis, Cybernetics and System Res.2, R.Trappl. North-Holland, 1984.**

II. Айказян Э.М., Аракелян А.А. Проблемно-ориентированный язык ДИСКИ. Тезисы докладов Ш Всесоюзной конференции "Диалог-83", Протвино, I983.

I2. Айказян Э.М., Аракелян А.А., АРМ инженера -разработчика РЭА. Сб.мат.вопросы кибернетики и вычислительной техники, Ереван, вып.ХШ, I984.

I3. Аракелян А.А., Папян С.С. Системное программное обеспечение комплекса непрерывного наблюдения за кардиологическими больными. Тез.доклада I Республиканской конференции по медтехнике и кибернетике, Ереван, I984.

I4. Аракелян А.А., Саядян Г.А., Оганджанян С.Р. Алгоритмы автоматического синтеза микропрограмм функционального контроля БИС. Автоматика и вычислительная техника, № I, I983.

I5. Амирбекян В.С., Аракелян А.А. и др. Пакет стандартных программ для решения задач автоматического синтеза тестов для микро-ЭВМ, Электронная техника, серия 3. Микроэлектроника, вып.4 (94),I98I.

I6. Аракелян А.А. Оптимизация систем автоматического контроля параметров АСУ ТП в условиях неопределенности. Ш Всесоюзное совещание "Надежность и эффективность АСУ ТП "АСУП", сб. тезисов. М., I984.

I7. Аракелян А.А., Кркеян А.М. и др. Построение специализированного процессора для генерации тестов функционального контроля БИС ЗУ, Электронная техника, серия 3. Микроэлектроника, вып.I (I07, I984.

I8. Аракелян А.А. Оптимизация диагностирования СОД на базе микропроцессоров. Тезисы докладов П Всесоюзного семинара по методам синтеза типовых модульных СОД, М., I985.

I9. Аракелян А.А. Существование решений разложимых отношений. Мат.проблемы кибернетики и вычислительной техники,№ХУ, I985.

The methods of optimization of dialogue-systems in experimental research

A.A. Arakeljan

Summary

In the paper the principles of optimization of dialogue-systems of experimental research are formulated and studied.

MTA SZTAKI Tanulmányok 202/1987 pp. 39-62

On monotone clones

J. Demetrovics - L. Hannák - L. Rónyai
Computer and Automation Institute
Hungarian Academy of Sciences
1502 Budapest, Pf.63, Hungary

In the present paper we deal with the connection between the structure of a finite bounded poset $P=\langle A;\leq\rangle$ and some algebraic properties of the clone $C(P)$ - the set of all monotone functions over P.

Let A be a finite set. The function $f:A^n \longrightarrow A$ is an n-variable function over A. A set of function over A is called clone if it is closed under arbitrary superpositions and both permutations and identifications of variables, and it contains all projections. If $P=\langle A;\leq\rangle$ and $Q=\langle B;\leq\rangle$ are posets, the map $f: A\mapsto B$ is monotone if $x\leq y$ in P implies $f(x)\leq f(y)$ in Q. The product $P\times Q$ is a poset $\langle A\times B;\leq\rangle$ where $(x,y)\leq(x',y')$ if and only if $x\leq x'$ in P and $y\leq y'$ in Q. If $P=\langle A;\leq\rangle$ then the n-variable function f is monotone if it is a monotone mapping from P^n into P. The poset $P=\langle A,\leq\rangle$ is bounded if there are elements $0,1\in A$ such that for all elements $x\in A$ $(0\leq x\leq 1)$.

The set of all monotone functions over $P=\langle A;\leq\rangle$, $C(P)$ obviously forms a clone over A. If P is finite and bounded then $C(P)$ is a maximal clone. /See Rosenberg [7] ./

The first part of the paper is strongly connected with the results in [3], where we studied the question:

Research partially supported by Hungarian National Foundation for Scientific Research, grant 1066

whether the clone $C(P)$ is finitely generated for every finite bounded poset P. /The clone C is finitely generated if there exists a finite set C' such that the least clone containing C' is exactly C./ We proved, that for a fairly large class of posets the answer is positive. Theorem 2 in [3] states that if L is a lattice, H a convex subset of L and $P=L\setminus H$, then $C(P)$ contains a near unanimity function /i.e. a function $d(x_1,\ldots,x_n)$ with the property

$$d(x,\ldots,x,y) = d(x,\ldots,y,x)=\ldots= d(y,x,\ldots,x) = x$$

and hence $C(P)$ is finitely generated.

In the first section we give a condition for a poset P to be obtained from a lattice L by removing a convex subset of L.

In the second section we investigate some congruence properties of varieties generated by the algebra $\langle A;C(P)\rangle$ where $P=\langle A;\leq\rangle$ is a bounded finite poset. We prove, that $\langle A;\ C(P)\rangle$ generates a congruence distributive variety provided $P= L\setminus H$ where L is a lattice and H a convex subset of L.

On the other hand the minimal poset for which $C(P)$ is not known to be finitely generated is R_8 . /See Fig. 1 /. We shall show that the algebra $\langle R_8;C(R_8)\rangle$ does not generate a congruence-modular variety.

In the third part we summarize some results on subclones of $C(R_8)$. We prove, that the clone of all monotone non-surjective functions over R_8 is finitely generated.

R_8

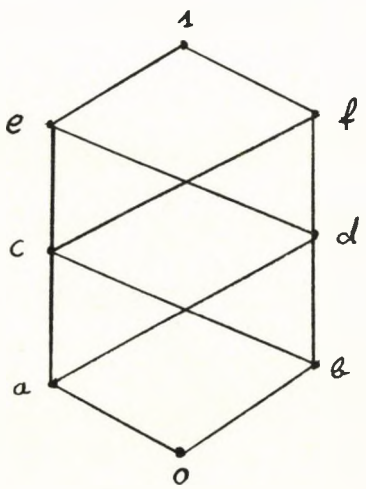


Figure 1.

1.§.

Let P be a finite bounded poset. $H \subseteq P$ is a hereditary subset if $H \neq \emptyset$ and $x \in H$, $y \in P$, $y \leq x$ imply that $y \in H$. A hereditary subset H of P is "good" if $a,b \in H$ implies $\sup(a,b) \in H$ whenever $\sup(a,b)$ exists in P.

Let

$$\mathcal{L} = \{H;\ H \text{ is a good subset of } P\}.$$

If $a \in P$ then $(a] = \{x \in P;\ x \leq a\}$ is obviously a good subset of P, so it is easy to check that $a \longrightarrow (a]$ is an order preserving injection of P into the poset $\langle \mathcal{L}; \subseteq \rangle$. If $a,b \in P$ are incomparable elements then $(a]$ and $(b]$ are also incomparable, hence P can be understood as a subposet of $\mathcal{L}$. A simple calculation shows that $\mathcal{L}$ is a lattice /if $H,K \subseteq P$ are good subset, then $H \cap K$ is a good subset, too/.

Proposition 1.1. For a finite bounded poset P the following are equivalent:

(i) P can be obtained as $P = L \setminus C$ where L is a lattice and C is a /possibly empty/ convex subset of L.

(ii) $\mathcal{L} \setminus P$ is a convex subset of $\mathcal{L}$.

Proof. (ii) $\longrightarrow$ (i) is trivial.

(i) $\longrightarrow$ (ii): Let us suppose that $C = L \setminus P$ is

not convex. Then there are $H,H' \in C$ and $a_o \in P$ for which $H \subseteq (a_o] \subsetneq H'$. As $H \neq (b]$ for every $b \in P$, H must contain /as a subset of P/ at least two maximal elements say a_1,a_2 which do not have least upper bound in P and $H \subseteq (a_o]$ implies that $a_1 \prec a_o$ and $a_2 \prec a_o$. Similarly we can choose $a_3,a_4 \in H'$ for which $a_o \leq a_3$ and sup (a_3,a_4) does not exist. Now suppose that P is a subset of a lattice L. Then let $b=\sup(a_1,a_2)$ in L and $c=\sup(a_3,a_4)$ in L. Then $a_1 \prec a_o$, $a_2 \prec a_o$ imply that $b \leq a_o$. On the other hand, from $a_3 \geq a_o$ we obtain that $c \geq a_o$. Now $b,c \in L \setminus P$, $b \leq a_o \leq c$ and $a_o \in P$ imply that $L \setminus P$ is not convex.

The following corollary is a simple reformilation of the statement just have been proved.

Corollary 1.2. A poset P cannot be obtained as a $P=L \setminus H$ where L is a lattice and H is a /possibly empty/ convex subset of L if and only if P contains elements, a_1,a_2,a_3,a_4 for which sup (a_1,a_2), sup (a_3,a_4) do not exist and $a_1 \prec a_3, a_2 < a_3$ hold.

Corollary 1.3. Let P be a bounded poset and suppose that $|P| \leq 7$. Then P can be obtained as $P=L \setminus H$ where L is a lattice and H is a /possibly empty/ convex subset of L.

Proof. Suppose that P has elements a_1,a_2,a_3,a_4 as

described in Corollary 1.2. It is obvious that a_1,a_2,a_3,a_4 are pairwise different elements of P. The poset P is bounded and sup (a_3,a_4) does not exist, hence a_3,a_4 have at least two minimal upper bounds, say b_1,b_2. Now P contains at least eight different elements $0,a_1,a_2,a_3,a_4$, $b_1,b_2,1$. This is a contradiction.

Corollary 1.4. /Lau [5]/. If the bounded poset P has at most seven elements then C(P) is finitely generated.

Proof. Using Corrolary 1.3 and [3], (Theorem 2) we get that P has a monotone near unanimity function /m.n.u.f./ and the statement follows.

Remark. The condition of Proposition 1.1 is not necessary for a poset to have a m.n.u.f.. Consider the poset RxR where R is the poset represented by Fig. 2.

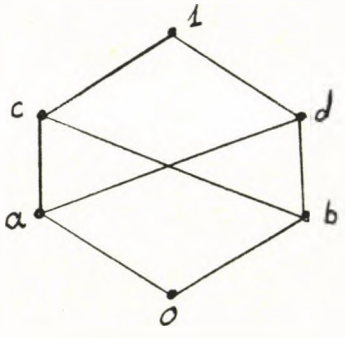


Figure 2.

Using [3], (Theorem 1), we know that RxR has a five variable m.n.u.f. . On the other hand let $a_1=(0,a)$, $a_2=(0,b)$, $a_3=(a,1)$

$a_4=(b,1)$. These elements satisfy the requirements of Corollary 1.2 hence RxR cannot be obtained by removing a convex subset from a lattice.

2.§.

In this section we shall investigate some congruence properties of varieties generated by an algebra of the form $\langle P;C(P)\rangle$ where P is a bounded finite poset.

Let us recall some definitions. A variety V is congruence distributive /modular/ if for every algebra $A \in V$ the congruence lattice of A is distributive /modular/.

In [4] Jonsson gave a Mal'cev condition for congruence distributive varieties: a variety V is congruence distributive iff it satisfies Δ_n for some $n \geqslant 2$ where Δ_n is the following property:

There are terms $p_0(x,y,z),\ldots,p_n(x,y,z)$ such that the following identities hold in V:

$$p_i(x,y,x)=x \quad (0 \leq i \leq n)$$
$$p_0(x,y,z)=x \quad p_n(x,y,z)=z$$
$$p_i(x,x,y)=p_{i+1}(x,x,y) \text{ for } i \text{ even}$$
$$p_i(x,y,y)=p_{i+1}(x,y,y) \text{ for } i \text{ odd.}$$

It is easy to see that Δ_n trivially implies Δ_m for all $m \geqslant n$.

The following Mal'cev condition for congruence modularity was obtained by Day([2]):

A variety V is congruence modular if for some $n \geqslant 2$ there are four variable terms $q_0(x,y,z,u),\ldots,q_n(x,y,z,u)$ such that V satisfies the following:

$$q_i(x,y,y,x) = x \qquad (0 \leq i \leq n)$$

$$q_0(x,y,z,u) = x \qquad q_n(x,y,z,u) = u$$

$$q_i(x,y,y,u) = q_{i+1}(x,y,y,u) \qquad \text{for } i \text{ odd}$$

$$q_i(x,x,u,u) = q_{i+1}(x,x,u,u) \qquad \text{for } i \text{ even.}$$

The property Δ_2 for a variety means that there is a ternary majority term. Thus, if for a bounded finite poset P the variety $\mathrm{Var}(\langle P, C(P)\rangle)$ satisfies Δ_2 then P is a lattice by [3] (Proposition 4).

Proposition 2.1. Let P be a bounded finite poset which is not a lattice. Then $\mathrm{Var}(\langle P, C(P)\rangle)$ does not satisfy Δ_3 .

Proof. In this case there are elements $a,b,c,d \in P$, $a \leq d$, $a \leq c$, $b \leq c$, $b \leq d$ such that there is no $x \in P$ for which $a \leq x$, $b \leq x$ $x \leq c$, $x \leq d$ hold. Now suppose that P satisfies Δ_3 and consider the element $x = p_1(a,c,b)$. Then $x \leq p_1(c,c,c) = c$ and $x \leq p_1(d,c,d) = d$. Now combining the facts

$$a = p_0(a,a,b) = p_1(a,a,b) \qquad \text{and}$$

$$p_1(a,a,b) \leq p_1(a,c,b)$$

we obtain the ineqality $a \leq x$.

Similarly, from

$$p_1(a,c,b) \geq p_1(a,b,b)$$

$$p_1(a,b,b) = p_2(a,b,b) \geq p_2(0,0,b)$$

$$p_2(0,0,b) = p_3(0,0,b) = b \qquad /0 \text{ is the least element of P/}$$

we conclude that $b \leq x$. This is a contradiction.

Mitschke ([6]) proved that if a variety V has a d-variable near unanimity term then it satisfies $\Delta_{2(d-2)}$ In particular, if P is a bounded poset and $P=L \setminus H$ where L is a lattice and H is a convex subset of L then $\mathrm{Var}(\langle P, C(P)\rangle)$ satisfies Δ_d for some $d \geq 2$. In Theorem 2.3 we shall improve this statement.

Let $\langle L, \vee, \wedge \rangle$ be a lattice. Following Baker we define the lattice polynomials r_1, r_2 and their duals s_1, s_2 as follows:

$$r_1(x_1,x_2,x_3) = x_1 \wedge (x_2 \vee x_3);$$
$$r_2(x_1,x_2,x_3) = x_1 \wedge x_3 ;$$
$$s_1(x_1,x_2,x_3) = x_1 \vee (x_2 \wedge x_3);$$
$$s_2(x_1,x_2,x_3) = x_1 \vee x_3 .$$

Next we make a few simple observations about these functions. The proofs are straightforward hence they are omitted.

<u>Proposition</u> 2.2. Let $\langle L, \vee, \wedge \rangle$ be a lattice and $x,y,z \in L$, $\underline{x}, \underline{y} \in L^3$, $\underline{x} \leq \underline{y}$. Then

(i) s_1, s_2, r_1, r_2 are monotone functions from L^3 to L.

(ii) $s_2(\underline{x}) \geq x_1$, $s_2(\underline{x}) \geq x_3$ and $r_2(\underline{x}) \leq x_1$, $r_2(\underline{x}) \leq x_3$.

(iii) $s_2(\underline{y}) \geq r_2(\underline{x})$.

(iv) $s_1(\underline{x}) \geq x_1$ and $r_1(\underline{x}) \leq x_1$.

(v) $s_1(\underline{y}) \geq r_1(\underline{x})$.

(vi) $r_i(x,y,x) = s_i(x,y,x) = x$ for $i=1,2$.

(vii) $r_1(x,x,y) = s_1(x,x,y) = x$.

(viii) $r_2(x,y,z) = r_2(z,y,x)$ and

$s_2(x,y,z) = s_2(z,y,x)$.

Theorem 2.3. Let P be a bounded poset such that $P=L\setminus H$ where L is a lattice, H is a convex subset of P. Then $V=\mathrm{Var}\left(\langle P, C(P)\rangle\right)$ satisfies Δ_4.

Proof. We may suppose that H is not empty. /Indeed, if $H=\emptyset$ then P is a lattice, hence Δ_2 holds in V./ Let $A=\{x\in P;\ \exists\ y\in H \quad x\not\leq y\}$, $B=P\setminus A$. A straightforward calculation shows that $x\in A$, $y\in L$, $y\not\leq x$ imply $y\in A$ and $x\in B$, $y\in L$ $y\geq x$ imply $y\in B$. Now let

$C=\{\underline{x}\in P^3;$ at least two components of x belong to A$\}$,

$D=\{\underline{x}\in P^3;$ at least two components of x belong to B$\}$.

It is easy to see that $C\cap D=\emptyset$ and $C\cup D=P^3$.

We define the functions $p_1,p_2\colon P^3\longrightarrow L$ as follows:

$$p_1(\underline{x}) = \begin{cases} s_1(\underline{x}) & \text{if } \underline{x}\in D \text{ and } s_1(\underline{x})\in B, \\ r_1(\underline{x}) & \text{if } \underline{x}\in C \text{ and } r_1(\underline{x})\in A, \\ x_1 & \text{otherwise;} \end{cases}$$

and

$$p_2(\underline{x}) = \begin{cases} s_2(\underline{x}) & \text{if } \underline{x} \in D, \\ r_2(\underline{x}) & \text{if } \underline{x} \in C. \end{cases}$$

First we show that $\mathrm{im}(p_1) \subseteq P$ and $\mathrm{im}(p_2) \subseteq P$ and p_1, p_2 are isotone functions from P^3 to P. $\mathrm{Im}(p_1) \subseteq P$ is obvious from the definition. If $\underline{x} \in D$ then at least one of x_1 and x_3 belongs to B. Now using the increasing property of B and by Proposition 2.2 (ii) $s_2(\underline{x}) \in B \subseteq P$. A similar argument works if $\underline{x} \in C$ so we conclude that $\mathrm{im}(p_2) \subseteq P$.

Now let $\underline{x}, \underline{y} \in P^3$, $\underline{x} \leq \underline{y}$. Then we can have the following three cases:

(a) $\underline{x}, \underline{y} \in D$;

(b) $\underline{x}, \underline{y} \in C$;

(c) $\underline{x} \in C$, $\underline{y} \in D$.

/We note that $\underline{x} \in D$, $\underline{y} \in C$ cannot occur./

The monotonity of p_2 follows immediately from Proposition 2.2 (i) and (iii). Now we prove the monotone property of p_1.

(a) If $s_1(\underline{x}) \in B$ then the monotone property of s_1 implies that $s_1(\underline{y}) \in B$ and $p_1(\underline{x}) = s_1(\underline{x}) \leq s_1(\underline{y}) = p_1(\underline{y})$. If $s_1(\underline{x}) \notin B$ then $p_1(\underline{x}) = x_1$ and $p_1(\underline{y}) \in \{y_1, s_1(\underline{y})\}$. In both cases $p_1(\underline{y}) \geq x_1$.

(b) A simple "dualization" of case (a)

(c) If $s_1(\underline{y}) \notin B$ and $r_1(\underline{x}) \notin A$ then $p_1(\underline{y}) = \underline{y}_1 \geq \underline{x}_1 = p_1(\underline{x})$.
If $s_1(\underline{y}) \in B$ and $r_1(\underline{x}) \in A$ then $p_1(\underline{y}) \geq p_1(\underline{x})$ follows from Proposition 2.2 (v). In the remaining cases we may use Proposition 2.2 (iv).

From Proposition 2.2 (vi) we get that for i=1,2

(1) $\quad p_i(x,y,x) = x$ for every $x,y \in P$.

From Proposition 2.2 (vii) we obtain that

(2) $\quad p_1(x,x,y) = x$ for every $x,y \in P$.

Now let $x,y \in P$. We shall compute $p_1(x,y,y)$.

If $(x,y,y) \in D$ then $y \in B$ and

$$s_1(x,y,y) = x \vee (y \wedge y) = x \vee y = s_2(x,y,y) = p_2(x,y,y) \in B.$$

If $(x,y,y) \in C$ then $y \in A$ and

$$r_1(x,y,y) = x \wedge (y \vee y) = x \wedge y = r_2(x,y,y) = p_2(x,y,y) \in A.$$

In both cases we conclude that

(3) $\quad p_1(x,y,y) = p_2(x,y,y)$ for $x,y \in P$.

From Proposition 2.2 (viii) we derive the following identity:

(4) $\quad p_2(x,y,z) = p_2(z,y,x)$ $\quad x,y,z \in P$.

Now we define the function $p_3: P^3 \to P$ as

(4') $\quad p_3(x,y,z) = p_1(z,y,x)$

p_3 is a monotone function and from (1) and (2) we obtain the identities

(5) $p_3(x,y,y) = y$ for $x,y \in P$ and

(6) $p_3(x,y,x) = x$ for $x,y \in P$.

Combining the identities (3) and (4) we obtain

(7) $p_2(x,x,y) = p_3(x,x,y)$, $x,y \in P$.

The identities (1)-(3) and (5)-(7) show that the functions $p_o = x_1$, p_1, p_2, p_3 and $p_4 = x_3$ satisfy the identities involved in Δ_4.

The proof is now complete.

Remark. Our Jónsson functions satisfy some additional identities as well, namely (4) and (4').

The property Δ_n for a fixed n is a property of order varieties.

Proposition 2.4. If P is a poset satisfying Δ_n and P' is a retract of P then P' satisfies Δ_n. If P_i $i \in I$ are posets and Δ_n holds for every P_i then $Q = \prod_{i \in I} P_i$ also satisfies Δ_n.

Proof. Let $d: P \longmapsto P'$ be a retraction and $p_o, \ldots, p_n$ the Jónsson functions for P. Then let $p_i'(x) = d(p_i(x))$ $0 \leq i \leq n$. Clearly, $\operatorname{im}(p_i') \subseteq P'$ and p_i is an isotone function $(0 \leq i \leq n)$. It is obvious that p_o' and p_n' are the first and the third projection on $(P')^3$ resp. If $x,y \in P'$ then

$$p_i'(x,y,x) = d(p_i(x,y,x)) = d(x) = x \qquad (0 \leq i \leq n)$$

as required. The proof of the remaining identities is immediate from the definition.

Now for $i \in I$ $p_0^i,\ldots,p_n^i$ will denote a collection of Jónsson functions for the poset P_i. For all j $(0 \leq j \leq n)$ we shall define the functions

$p_j : Q^3 \longmapsto Q$ as

$$(\ldots, \underline{x}^i, \ldots) \xrightarrow{p_j} (\ldots, p_j^i(\underline{x}^i), \ldots)$$

where $\underline{x}^i \in P_i^3$. The proof of the identities is a routine calculation and therefore it will be omitted.

Concerning R. Pöschel's question for the existence of a finite generating set of $C(P)$ the first unsettled case is the eight element poset R_8 /see Fig. 1 /.

We do not know whether $C(R_8)$ is finitely generated or not. Now we show that event the congruence modularity fails to hold for $\mathrm{Var}(\langle R_8, C(R_8)\rangle)$.

Proposition 2.5. $\mathrm{Var}(\langle R_8, C(R_8)\rangle)$ is not congruence modular.

Proof. Suppose that there are monotone functions $q_0,\ldots,q_n$ as described in Day's theorem. In [3] we proved that if $g: R_8 \to R_8$

is an idempotent isotone function then $g(\underline{x}) \in \{c,d\}$ whenever $\underline{x} \in (c,d)^n$. The functions q_i are idempotent, hence $q_i(\underline{x}) \in \{c,d\}$ if $\underline{x} \in \{c,d\}^4$ $(0 \leq i \leq n)$.

Next we remark that for every $0 \leq i \leq n$

(8) $q_i(c,d,d,d) = q_i(c,c,d,d)$.

Indeed if these values were different then from

$(c,d,d,d) \leq (c,e,e,d) = \underline{x}$ and

$(c,c,d,d) \leq (c,e,e,d)$

we would have $c \leq q_i(\underline{x})$ and $d \leq q_i(\underline{x})$. On the other hand from inequalities

$$\underline{x} \leq (e,e,e,e)$$

$$\underline{x} \leq (f,e,e,f)$$

we obtain $q_i(\underline{x}) \leq e$, $q_i(\underline{x}) \leq f$ and this contradicts with the fact there is no element in R_8 between c,d, and e,f.

Now using (8) we shall prove that

$$q_i(c,c,d,d) = c$$

for every $0 \leq i \leq n$. This is true for i=0. If we know this equality for $i \leq n$, then there are two cases:

a/ i is even. Then from Day's theorem

$$c = q_i(c,c,d,d) = q_{i+1}(c,c,d,d) .$$

b/ i is odd. Then from (8) and from Day's theorem we obtain

$$c = q_i(c,d,d,d) = q_{i+1}(c,d,d,d).$$

Using again (8) for i+1 we conclude that

$$c=q_{i+1}(c,c,d,d).$$

Finally, we obtain that

$$c=q_n(c,c,d,d)$$

and this is a contradiction.

3.§.

So far we have obtained only negative facts about the poset R_8. Our aim in this section is to prove something in the positive direction. The non-surjective functions from $C(R_8)$ and the projections form a clone D. We are going to show that D is finitely generated. This result will be obtained as a consequence of a more general theorem.

Let A,B, $A \supseteq B$ be nonempty finite sets and C is a clone of operations over A. Let $C[B]$ denote the clone consisting of the projections and all functions $f \in C$ for which $\operatorname{im} f \subseteq B$.

A k-ary function $(k \geq 3)$ $d: A^k \to B$ is B-near unanimity function if for every $x,y \in B$
$x = d(y,x,\ldots,x) = d(x,y,x,\ldots,x) = \ldots = d(x,x,\ldots,x,y)$ hold.

Theorem 3.1. If the clone C contains a B-near unanimity function d then $C[B]$ is finitely generated.

The proof will depend on the following lemma.

Lemma 3.2. Let $\emptyset \neq B \subseteq A$ finite sets and $\langle A,F \rangle$ an algebra over A. Suppose that $\operatorname{im} f \subseteq B$ for every $f \in F$ and that F contains a k+1-ary B near unanimity function d. Then an operation $g: A^k \to B$ is a polynomial function over $\langle A,F \rangle$

iff it preserves all subalgebras of $\langle A,F\rangle^k$.

This statement /and its proof/ is a slight modification of corrolary 5.1. from Baker-Pixley [1]. For the sake of completeness we include a proof here.

Proof of Lemma 3.2. The necessity of the condition is obvious. Now suppose that $g: A^n \longmapsto A$ preserves all subalgebras of $\langle R_g,F\rangle^k$. We shall use induction on $|D|$ to obtain interpolating polynomials for g on every subset D of A. Let us have a collection of vectors $\underline{x}^1,\ldots,\underline{x}^k$ from A^n. As g preserves the subalgebra of A^k generated by the vectors $x_i^1,x_i^2,\ldots,x_i^k$ where $1 \leq i \leq n$, the vector $g(\underline{x}^1),\ldots,g(\underline{x}^k)$ belongs to this subalgebra, hence there is a polynomial p for which $p(\underline{x}^1)= g(\underline{x}^1),\ldots,p(\underline{x}^k) =g(\underline{x}^k)$. So far we have obtained that on every $\min(k, |A|^n)$- element subset D of A there is an interpolating polynomial p_D for g. So we may suppose that $|A|^n > k$ and there is an interpolating function on every ℓ element subset of A^n, where $k \leq \ell \leq A^n$. Let D be a subset of $A^n, |D| = \ell + 1$. Let $x_1,\ldots,x_{k+1}$ be different elements of D and let $D_i = D \setminus \{x_i\}$ $1 \leq i \leq k+1$. By induction hypothesis, there are interpolating polynomials p_{D_i} for g on the sets D_i, $1 \leq i \leq k+1$. We remark that if an interpolating polynomial p_{D_i} is a projection, we may replace it with the polynomial $d(p_{D_1},\ldots,p_{D_i})$, so we may suppose that our polynomials take values only from B. Now using the B-near unanimity of d

we obtain that $d(p_{D_1},\dots,p_{D_{k+1}})$ interpolates g on D. The proof is complete.

Proof of theorem 3.1. Let us suppose that d is a $k+1$-ary function for some $k \geq 2$ and consider the subsets of A^k which are not subalgebras of the algebra $\mathcal{U}^k = \langle A, C[B] \rangle^k$. For every such subset D we choose a function $f_D \in C[B]$, which does not preserve D. By the definition of subalgebras such functions do exist. Now let us consider the following set:

$F = \{ f_D;\ D \subseteq A^k,\ D$ is not a subalgebra of $\langle A, C[B] \rangle^k \} \cup \{d\}$.

F is a finite set of operations and $F \subseteq C[B]$ because of $f_D \in C[B]$ and $d \in C[B]$. Now consider the algebra $\mathcal{A} = \langle A, F \rangle$. A subset $H \subseteq A^k$ is subalgebra of $\mathcal{U}^k$ iff it is a subalgebra of $\mathcal{A}^k$. Thus applying Lemma 3.2, we obtain that every $f \in C[B]$ is a polynomial function over $\langle A, F \rangle$, that is $C[B] \subseteq [F]$. /Here $[F]$ denotes the clone generated by F over A./ The inclusion $[F] \subseteq C[B]$ is obvious, hence F is a generating set of $C[B]$. The proof is complete.

Now we prove two preparatory propositions about the poset R_8.

Proposition 3.3. Let $C \subseteq R_8$ be a proper bounded subposet of R_8. Then there is a proper retract B for which $C \subseteq B$.

Proof. The poset R_8 has a unique order preserving automorphism

π for which $\pi^2=id$, $\pi(a)=b$, $\pi(c)=d$, $\pi(e)=f$ hold. First we notice that if $x \in \{a,b,c,d,e,f\}$ then $R_8 \setminus \{x\}$ is a retract of R_8. Indeed, let us define the map $r_x: R_8 \to R_8 \setminus \{x\}$ as

$$r_x(y) = \begin{cases} \pi(y) & \text{if } y = x, \\ y & \text{otherwise.} \end{cases}$$

It is obvious that r_x is a monotone map, $\mathrm{im}(r_x) = R_8 \setminus \{x\}$ and $r_x(r_x(y)) = r_x(y)$ for every $y \in R_8$, hence r_x is a retraction and $R_8 \setminus \{x\}$ is a proper retract of R_8. Now if B is a proper bounded subposet of R_8, then $B \supseteq \{a,b,c,d,e,f\}$ is impossible and the statement follows.

Proposition 3.4. If B is a proper retract of R_8, then there exists a monotone B-near unanimity function.

Proof. In this case B is a bounded poset and $|B| \leq 7$. By Corrollary 1.3 there is an isotone near unanimity function $d_B: B^k \to B$ for some $k \geq 3$. Let $r: R_8 \to B$ denote a retraction onto B. Now we define the function $d'_B : R_8^k \longrightarrow B$ as

$$d'_B(x_1,\ldots,x_k) = d_B(r(x_1),\ldots,r(x_k))$$

The function d'_B has the desired properties.

Now we return to our original subject, to the clone D.

Theorem 3.5. The clone D consisting of all non-surjective monotone functions of R_8 is finitely generated.

Proof. If $f \in D$ and f is not a projection then $\mathrm{im}(f)$ is a proper bounded subposet of R_8, hence $\mathrm{im}\, f \subseteq B$ for some proper retract B, by Proposition 3.3. So, it is enough to prove that $C(R_8)[B]$ is finitely generated for every proper retract B. By Proposition 3.4. $C(R_8)[B]$ contains a B-near unanimity function. Applying Theorem 3.1 we get that $C(R_8)[B]$ is finitely generated.

Finally, the authors wish to thank to I. Ágoston, B. Csákány, G. Czédli, R. Queckenbush, I.G. Rosenberg and Á. Szendrei for their valuable remarks and suggestions.

[1] K.A. Baker, A.F. Pixley; Polynomial interpolation and the chinese remainder theorem for algebraic systems. Math.Z., 143 /1975/ 165-174.

[2] A.Day; A characterisation of modularity of congruence lattice of algebras. Canad. Math. Bull., 12 /1969/ 167-173.

[3] J. Demetrovics, L. Hannák, L. Rónyai; Near unanimity functions and partial orderings. Proc. 14. ISMVL., Manitoba, /1984/ 52-56.

[4] B. Jónsson; Algebras whose congruence lattice are distributive. Math. Scand., 21 /1976/ 110-121.

[5] D. Lau; Bestimmung der Ordnung maximaler Klassen von Functionen der k-wertigen Logik. Zeitschrift für Math. Logik und Gundlagen der Mathematik,24 /1978/ 79-96.

[6] Mitschke; Near unanimity identities and congruence distributivity in equational classes. Alg. Univ., 8 /1978/ 1, 29-32.

[7] I.G. Rosenberg; Über die functional Vollständigkeit in den mehrwertigen Logiken. Rozpr. CSAV Rada Math. Priv. Ved., Praha 80, 4 /1980/ 3-93.

On monotone clones

J. Demetrovics, L. Hannák, L. Rónyai

Summary

The paper deals with the connection between the structure of a finite bounded poset $P = \langle A; \leq \rangle$ and some algebraic properties of the clone $C(P)$ - the set of all monotone functions over P. The first part of the paper deals with the question whether the clone $C(P)$ is finitely generated for every finite bounded poset P. It is proved that for a fairly large class of posets the answer is positive. In the second part some congruence properties of varieties generated by the algebra $\langle A; C(P) \rangle$ are investigated.

MTA SZTAKI Tanulmányok 202/1987 pp. 63-73

A lower estimation for the cardinality of finite difference sets in R^n

G. Freiman
Department of Mathematics, Tel-Aviv University
Ramat-Aviv, 69978 Tel-Aviv, Israel
A.Heppes and B. Uhrin
Computer and Automation Institute, Hungarian
Academy of Sciences
1502 Budapest, P.f.63, Hungary

1. Introduction

In what follows the sets A,B are finite subsets of R^n or more specifically of $\Lambda \subset R^n$, where Λ is a point lattice i.e. integer combination of n independent vectors (expecially $\Lambda = Z^n$, the integer vectors). The cardinality of A is denoted by $|A|$ and $-A$ means the set $\{-a: a \in A\}$. We denote by $A+B$ the algebraic sum of the sets. The set $A+A$ is called the sum-set of A and $A-A=A+(-A)$ is the difference set of A.

It is clear that

$$(1.1) \qquad \frac{|A|(|A|+1)}{2} \geq |A+A| \geq 2|A| - 1$$

and

$$(1.2) \qquad |A|^2 - |A| + 1 \geq |A-A| \geq 2|A| - 1$$

for any finite $A \subset R^n$.

The structure of sum-sets of $A \subset Z^n$ such that

$$(1.3) \qquad |A + A| < \text{const}\, |A|$$

(the sum-sets of small cardinality), has been investigated by Freiman [1] .

On the other hand, little is known about $|A-A|$. A plausible idea is to compare $|A-A|$ with $|A+A|$ and to apply the results for $|A+A|$ (say from [1]). This has been done only for $A \subset Z^1$. Namely P. Erdős asked the following question (see [2]):

Is it true that for every constant c there exists another constant c' such that for any finite $A \subset Z^1$

$$(1.4) \qquad \frac{|A+A|}{|A|} \leq c \quad \text{implies} \quad \frac{|A-A|}{|A|} \leq c' \quad ?$$

J. Conway conjectured that $|A+A| \leq |A-A|$ for every finite $A \subset Z^1$ but this proved to be false (see [3] for details). I.Z. Ruzsa [2] has raised the question converse to that of Erdős':

Is it true that for every constant c there exists another constant c" such that for any finite $A \subset Z^1$

$$(1.5) \qquad \frac{|A-A|}{|A|} \leq c \quad \text{implies} \quad \frac{|A+A|}{|A|} \leq c'' \quad ?$$

Ruzsa answered both questins in the affirmative showing that $c'=c^2$ is good in (1.4) $c''=c^3$ is good in (1.5). He also showed that, denoting $\alpha := \log c' / \log c$ and $\beta := \log c'' / \log c$, for these numbers we have asymptotically (i.e. when $c \to +\infty$)

$$(1.6) \qquad 1.2 < \alpha \leq 2 \ , \quad 1.02 < \beta \leq 3$$

(see [2]).

In the n-dimenisonal case the proof of statement analogous to (1.6) seems to require basicly new ideas although, using the lexicographic ordering, a set in R^n "behaves" like a 1-dimensional one. Freiman [1] worked aut a whole theory to study lower estimations of $|A+A|$ in terms of $|A|$, however, his theory is not directly applicable to $|A-A|$.

The need for sharp lower estimations for $|A-A|$ in terms of $|A|$, when $A \subset \Lambda \subset R^n$, has been raised by Uhrin who used the trivial inequality $|A-A| \geq 2|A| - 1$ in proving

theorems sharpening the classical theorem of Minkowski-Blichfeldt in geometry of numbers (see [4],[5]). The method can be used succesfuly also in structures more general than R^n,[6]. We can state that the sharper estimation for $|A-A|$ we have, the sharper results in geometry of numbers can be proved (see Section 3 for details). More exactly, the following problem has been formulated in [4],[5]: Denoting by φ the canonical map of R^n into R^n/Λ, characterize the L-measurable sets $H \subset R^n$ such that

$$|H \cap \omega - H \cap \omega| \geq r\,|H \cap \omega| - s \tag{1.7}$$

for all $\omega \in \varphi(H)$, where $r \geq 2$, $s \geq 0$ are constants.
A "prototype" of characterization we have in mind is the following result proved for sum-sets, [1]: The (affine) dimension of the set $A \subset R^n$ is defined as the minimum of dimensions of flats containing the set.
If the dimension of $A \subset R^n$ is m then

$$|A+A| \geq (m+1)\cdot|A| - \frac{m(m+1)}{2} . \tag{1.8}$$

The mai result of this paper is that (1.8) is true also for $|A-A|$. This is proved in Section 2.
Section 3 is devoted to some possible sharpenings of the result and also to investigations of the equality in the inequality proved. The application of the results to geometry of numbers is also described in Section 3.

2. A lower bound for $|A-A|$

In this section we prove

Theorem 2.1. If the affine dimension of $A \subset R^n$ is m, then

$$|A-A| \geq (m+1)\cdot|A| - \frac{m(m+1)}{2} . \tag{2.1}$$ □

To establish our inductive proof, first we prove a

Lemma 2.1. If A spans the n-space then A can always be reduced by two points such that for the reduced set A' the inequality

$$|A-A| \geq |A'-A'| + 2(n+1) \tag{2.2}$$

holds. □

Proof of the lemma: Let $a_1, a_2, \ldots, a_k$ be the points of A where a_1 and a_k are two vertices of the convex hull of A such that they are unique points of A being on two parallel supporting hyperplanes of the convex hull. We can assume that $a_1 = -a_k$ (this can be achieved after a suitable translation of A). We shall denote the negative of an arbitrary point x by x'. Consider the set $B = A \cup (-A)$ and let C be the convex hull of B. Consider the edges of the polytope C that me t at a_1. On each of these edges there is a point of B nearest to a_1. We denote these points by $b_1, b_2, \ldots, b_d$, where d is the degree of the vertex.

Since C is an n-dimensional polytope, $d \geq n$ holds and the edge-vectors $a_1 - b_i$, $i=1,\ldots,d$ are all different. Let S_i be a supporting hyperplane of C containing nothing of C but the edge e_i defined by a_1 and b_i and S_i', be its "negative" i.e. the supporting hyperplane parallel to S_i (and containing nothing but the edge e_i', defined by a_k and b_i').

If f_1 and f_2 are point of C such that $f_1 - f_2 = a_1 - b_i = c_i$ then f_1 must lie on S_i and f_2 on S_2', more precisely, $f_1 \in e_i$ and $f_2 \in e_i'$. Consequently, the rest of the proof of our lemma requires only a simple 2-dimensional analysis as follows:

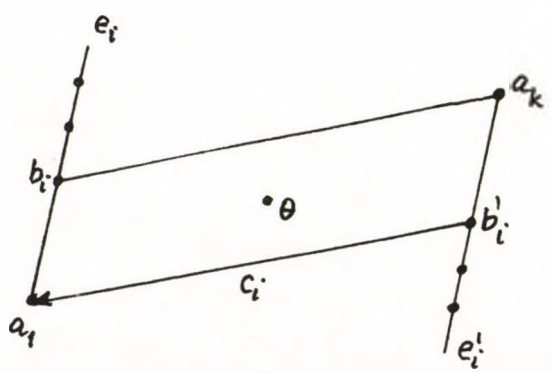


Figure 1.

It is clear (Fig.1) that f_1 and f_2 must lie on the closed segment $[a_1, b_i]$ and $[b_i', a_k]$, respectively. On the other hand we know that only the endpoints of these segments belong to B. Thus c_i has only two "representations" in B: a_1-b_i' and b_i-a_k. Consequently, as either b_i or b_i' belongs to A as well, the removal of a_1 and a_k from A reduces the cardinality of $|A-A|$ by at least 2d. That completes the proof of the lemma. ■

Before the proof of the theorem let us note that

$$|A-A| \geq 2|A|-1 \tag{2.4}$$

and equality holds in (2.4) if and only if $A=\{a, a+f, a+2f, \ldots, a+(k-1)f\}$, where f is a lexicographically positive element. (Both (2.4) and the statement about the equality in (2.4) can be proved easily taking the points of A into lexicographic order.)

Proof of the theorem. We can assume without the loss of generality that m=n. Let $k:=|A|$. It is clear that $k \geq n+1$. The proof goes by induction on n+k. For n=1 (2.1) is true for all k by (2.4). Assume that the theorem is true for all n and k such that $k \geq n+1$ and $n+k \leq N-1$ and let n,k be such that $n+k=N$, $k \geq n+1$.

We shall distinguish three special cases.

Case 1. If the convex hull of A is a pyramid, e.e. (k-1) points of A span a hyperplane and the remaining point, say a_k, is outside the hyperplane, then every vector a_k-a, $a-a_k$, $a \in A'=\{A \setminus a_k\}$ is unique, hence $|A-A| \geq |A'-A'| + 2(k-1)$ and using the inductive hypothesis we get

$$|A-A| \geq (n+1)k - \frac{n(n+1)}{2} + k-2. \tag{2.5}$$

We note that (2.5) is even sharper than (2.1).

Case 2. Let the set $A' = A \setminus \{a_1, a_k\}$, where a_1, a_k are the two points quaranteed by the lemma, be of the dimension $d < n$. If $d=n-2$ then for A we have the Case 1. Let $d=n-1$. The $a_1 - a_k$ and $-a_1 + a_k$ are unique, hence for the set $A' = A \setminus a_1$ we have $|A-A| \geq |A'-A'| + 2$ and applying to A' the Case 1 we get

$$|A-A| \geq (n+1)k - \frac{n(n+1)}{2} + k - (n+2). \tag{2.6}$$

We note that in this case $k \geq n+2$ must hold, hence (2.6) is again sharper than (2.1).

Case 3. Assume that the convex hull of A is neither pyramid nor Case 2 holds. Then the set $A' = A \setminus \{a_1, a_k\}$, where a_1, a_k are the two points guaranteed by the lemma, is necessarily n-dimensional, hence applying the lemma and the inductive hypothesis we get (2.1). ∎

3. Remarks, applications

In this section we first mention some possible sharpenings of (2.1). When looking for sharpenings, the emphasis is taken on improving the multiplying constant (n+1) in (2.1) rather than the additive one $n(n+1)/2$. Let us consider a d-dimensional set D of points v_i $i=1,\ldots,k$ in R^n and let v be a vector not parallel to the d-dimensional flat spanned by D. It is easy to show that the cardinality of the difference set of the union E of D and D+v is $3|D-D|$, i.e. to any finite set we can construct another one with twice as many points while increasing its dimension by one and tripling the cardinality of its difference set. To see this, notice that by projection along v the difference vectors of E are mapped onto the difference vectors of D. The first three

of the four difference vectors v_i-v_j, $(v_i+v)-v_j$, $v_i-(v_j+v)$ and $(v_i+v)-(v_j+v)$ are all different and all are mapped onto v_i-v_j. Thus we can say that to each 'old' difference vector there corresponds exactly 3 'new' ones.

Let us look more closely at the situation in R^3. If D is 1-dimensional then $|D-D| \geq 2k-1$. "Doubling" this set according to the above rule we get $|E-E| \geq 3(2k-1) = 3|E| - 3$ and "doubling" E we finally get $|F-F| \geq 3(3|E|-3) \simeq 9|F|/2 - 9 = 4.5|F| - 9$. We see that this configuration yields "better" estimation than (2.1), in the sense that instead of $4|F|$ we have $4.5|F|$.

We conjecture that for m-dimensional sets consisting of $2^m.t$ points the above construction supplies the "best" sets. The formula in (2.1) and the conjectured one coincide in the one and two-dimensional case and the second is sharper for every higher dimension. Our opinion that the proved lower estimation is not the 'real' one is supported by the fact that even for 3-dimensional case simple parity consideration show that the additive constant -6 can be replaced by -5.

In the remaining part of the section we briefly sketch the application of Theorem 2.1 to geometry of numbers mentioned in Sec.1.

Let $\Lambda \subset R^n$ be a point lattice of full dimension and let P be one of its unit cells, i.e. a subset being in a one-to-one correspondence with the quotient space R^n/Λ. Usually $P=\{x \in R^n : x = \sum_{i=1}^{n} \lambda_i b_i,\ 0 \leq \lambda_i < 1\}$, where $b_1, \ldots, b_n \in R^n$ is a basis of Λ. Then the canonical map $\varphi : R^n \to R^n/\Lambda \sim P$ is defined as

$$(3.1) \qquad \varphi(x) = \sum_{i=1}^{n} \{\xi_i\} b_i \quad , \quad x = \sum_{i=1}^{n} \xi_i b_i \in R^n,$$

where $\{\xi_i\}$ is the fractional part of ξ_i, i.e. the number such that $0 \leq \{\xi_i\} < 1$ and $\xi_i - \{\xi_i\}$ is integer. By definition, for any set $S \subset R^n$ $\varphi(S) := \bigcup_{x \in S} \varphi(x)$.

Now, for any set $S \subset R^n$ denote (see [4], [5], [6]).

(3.2) $\quad S_i := \{ x \in P : |(S-x) \cap \Lambda| = i \}, \quad i = 0,1,2,\ldots .$

One can see easily that $\varphi(S) = \bigcup_{i=1}^{\infty} S_i$.
Moreover, the following identities can be easily proved

(3.3) $\quad V(\varphi(S)) = \sum_{i=1}^{\infty} V(S_i),$

(3.4) $\quad V(S) = \sum_{i=1}^{\infty} i \cdot V(S_i) .$

(These identities hold in a much more general setting, see [6]).
The set S defines one more decomposition of $\varphi(S)$ as follows.

(3.5) $\quad S^{(j)} := \{ x \in P : \dim ((S-x) \cap \Lambda) = j \}, \quad j = 0,1,2,\ldots,n,$

where dim (·) is the affine dimension of the set.
It is clear that $\varphi(S) = \bigcup_{j=0}^{n} S^{(j)}$, consequently

(3.6) $\quad V(\varphi(S)) = \sum_{j=0}^{n} V(S^{(j)})$

Now, we have

Lemma 3.1. For any set $S \subset R^n$, if $S_i \cap S^{(j)} \neq \emptyset$, then

(3.7) $\quad |(S-S) \cap \Lambda| \geq (j+1)i - \frac{j(j+1)}{2} .$ □

Proof: The proof is an application of a simple idea used in [4] (see also [5], [6]) and (2.1). Namely, if $x \in S_i \cap S^{(j)}$, then $(S-x) \cap \Lambda \neq \emptyset$ and we can write

(3.8) $\quad (S-S) \cap \Lambda \supseteq (S-x) \cap \Lambda \; - (S-x) \cap \Lambda .$

hence

(3.9) $\quad |(S-S) \cap \Lambda| \geq |(S-x) \cap \Lambda \; - (S-x) \cap \Lambda| .$

Applying (2.1) we get (3.7). ∎

In spite of its simplicity, the lemma has an interesting consequence.

Theorem 3.1. Let $S \subset R^n$ be a bounded Lebesgue-measurable set of positive L-measure V(S). Then

$$(3.10) \qquad |(S-S)\cap\Lambda| \geq (m+1)\cdot\frac{V(S)}{V(\varphi(S))} - \frac{M(M+1)}{2}$$

where $m:=\min\{j: V(S^{(j)})>0\}$ and $M:=\max\{j:V(S^{(j)})>0\}$. □

Proof: The identities (3.3), (3.4) and (3.6) show that if $V(S_i)>0$ then there is j(i) such that $V(S^{(j(i))})>0$ and $S_i \cap S^{(j(i))} \neq \emptyset$. For such a pair of indices we have by (3.7)

$$(3.11) \qquad |(S-S)\cap\Lambda| \geq (j(i)+1)\, i - \frac{j(i).(j(i)+1)}{2}$$

and this implies

$$(3.12) \qquad |(S-S)\cap\Lambda| \geq (m+1)i - \frac{M((M+1)}{2}$$

Using (3.4) we get (3.10). ∎

The inequality (3.10) is a useful extension of Theorem 2, [6], p.936., where we had to assume the symmetry of the sets $(S-x)\cap\Lambda$, having had in hand the result (2.1) only for symmetric sets.

The inequality (3.10) can be compared also with the inequality

$$(3.13) \qquad |(S-S)\cap\Lambda| \geq 2\frac{V(S)}{V(\varphi(S))} - 1$$

proved earlier in [4],[5] , and in more general structures in [6].

Of course, the question, for what sets the right hand side

of (3.10) is greater than that of (3.13), seems to be worthy of study. In other words we have a problem to characterize those sets S for which

$$(3.14) \qquad (m-1)\,\frac{V(S)}{V(\varphi(S))} > \frac{M(M+1)}{2} - 1 .$$

Using (3.10) one can write statements of the following type (analogously to those in [6]):
"If $V(S)/V(\varphi(S)) > q$ then S-S contains at least p lattice points".
In any case, the decompositions (3.3), (3.4) and (3.6) seem to interesting also in themselves.

References

[1] G. Freiman, "Foundations of a structural theory of set addition", Am.Math.Soc., Providence, R.I., 1973.

[2] I.Z. Ruzsa, On the cardinality of A+A and A-A, In: A. Hajnal, V.T. Sós, Eds., "Combinatorics", Coll. Math. Soc. J. Bolyai, Vol 18, North-Holland, 1978, 933-938.

[3] I.Z. Ruzsa, Sets of sums and differences, Proc. of Seminaire de Theorie des nombres de Paris (1982-1983), Birkhäuser, Boston, 1984, 267-273.

[4] B. Uhrin, Some estimations useful in geometry of numbers, Period. Math. Hungar., 11 (1980), 95-103.

[5] B. Uhrin, On a generalization of Minkowski convex body theorem, J. of Number Th., 13 (1981), 192-209.

[6] B. Uhrin, Some remarks about the lattice points in difference sets, in: J. Szabados, Ed., "Proc. of A.Haar Memorial Conf.(Budapest, 1985)", Coll. Math. Soc. J. Bolyai, Vol 49, North-Holland, Amsterdam-New York, 1986, 929-937.

A lower estimation for the cardinality of finite difference sets in R^n

G. Freiman, A. Heppes, B. Uhrin

Summary

If $a_i \in R^n$, $i = 1,2,\ldots,N$, are different points such that the dimension of their affine hull is equal to m, then the first named author proved that among the points $a_i + a_j$, $1 \leq i,j \leq N$, there are at least $(m+1)N - m(m+1)/2$ different ones. In the paper it is shown that the same conclusion is true for the points $a_i - a_j$, $1 \leq i,j \leq N$. Using a method of the third named author, the result yields some refinements of the classical Minkowski-Blichfeldt theorem in the geometry of numbers /this is also discussed in the paper/.

MTA SZTAKI Tanulmányok 202/1987 pp. 75-98

A NON-WIENER RANDOM WALK IN A 2-D BERNOULLI ENVIRONMENT

A. Krámli[1] - P. Lukács[1] - D. Szász[2]

Summary For a degenerate random walk in a 2-D Bernoulli environment without local traps computer results show a non-Wiener behavior. For a better exploitation of the memory, the analysis is based on the statistics of the first exit time from a square.

Key words. Random walks in random environments, testing normality and independence, Erdős-Kac statistics, Monte-Carlo methods.

[1] Computer and Automation Institute of the Hungarian Academy of Sciences. H-1111, Budapest, Kende u. 13-17.

[2] Mathematical Insitute of the Hungarian Academy of Sciences. H-1364, Budapest, P. O. B. 127, Hungary.

Research partially supported by Hungarian National Foundation for Scientific Research, Grant No 1066

There are delicate problems where a conjectured anomaly differs from the regular behavior very little, e.g. a logarithmic correction vs. a power of time. In these cases even the use of powerful computers requires increased care because of limitations in memory and computing time.

An example is the asymptotic behavior of random walks in a random environment (RWiRE). Here we suppose that the environment is Bernoulli, i.e. it is translation-invariant and, for different sites, the random transition probabilities are independent. (In the Bernoulli case, for $d \geq 2$, the typical behavior is meant to be (strictly) diffusive, i.e. one with an asymptotically linear increase of the mean square displacement and with a Wiener limiting process for trajectories in the standard diffusion scaling. For d=1 Sinai (S-1982) showed that the displacement of a n.n. RWiRE $X_n : n \in \mathbb{Z}_+$ satisfying a suitable centralization condition increases like $(\log n)^2$, (i.e. the behavior is strongly subdiffusive.)

According to general belief there are two critical dimensions $d_{cr} \leq \bar{d}_{cr}$ determined as follows:
- if $d \geq \bar{d}_{cr}$, then any non-degenerate RWiRE on $\mathbb{Z}^d$ satisfying a suitable centralization condition is diffusive
("non-degenerate" means that the random transition probabilities are uniformly bounded away

from 0) and $\bar{d}_{cr}$ is the smallest integer with this property;

- if $d \geq d_{cr}$, then sufficiently small random perturbations of the simple symmetric random walk (SSRW) satisfying a suitable centralization condition behave diffusively and d_{cr} is the smallest integer with this property (the SSRW is determined by the non-random transition probabilities $p(x,y)=(2d)^{-1}$ if $x,y \in \mathbb{Z}^d$, $|x-y|=1$ while a perturbation is meant small if for a suitable $\varepsilon = \varepsilon(d) > 0$ Prob $(|p(x,y)-(2d)^{-1}| \leq \varepsilon)=1$ for any $x,y \in \mathbb{Z}^d$, $|x-y|=1$.)

We remark that even the formulation of the suitable centralization condition is a quite hard problem.

Relying upon formal perturbation theory and renorm-group methods several authors have agreed that $d_{cr} \sim 2$ (F (1984), L(1984), O(1984), P(1984)), i.e. in case of small disorder, for $d=2+\varepsilon$ the behavior is diffusive while for $d=2-\varepsilon$ it is subdiffusive (in fact, F (1984) showed that $\langle x^2(t) \rangle \sim t^{1-\varepsilon^2}$). As to the critical case $d=2$, the results also agree that in the increase of the mean square displacement there may be at most a logarithmic correction to the linear one. F(1984) obtains $\langle x^2(t) \rangle \asymp t(1+\frac{c}{\log t})$ and O(1984) even derives a strictly diffusive behavior. F(1984) also gives intuitive arguments for the extension of the aforementioned methods to the large disorder case. On the other hand, Bramson and Durett, B-R(1986) outline a rigorous construction for non-degenerate RWiRE on any $\mathbb{Z}^d$, $d \geq 1$ where the non-degenerate environment is translationally-invariant but not a Bernoulli one and the behavior is subdiffusive.

Since the environments of their example obey an exponential mixing condition, they stress that similar phenomenon may also occur for Bernoulli environments thus casting doubts on the folk believe that $d_{cr}<\infty$. The environment they construct is given by a random potential containing larger and larger traps visited by the random walker sufficiently often and this slowdown produces the subdiffusivity.

Earlier Marinari et al. M(1983) made numerical experiments in order to check whether the behaviour of a two dimensional random walk in a Bernoulli environment had a subdiffusive character.

The model studied by them looks as follows. To each site in $\mathbb{Z}^2$ there are assigned four numbers Q_i $(i=1,2,3,4)$ uniformly distributed in $(0,1)$. The transition probability to each of the four neighbouring sites is defined by $p_i(k)=Q_i^k \left(\sum_{j=1}^{4} Q_j^k\right)^{-1}$, where k is a parameter ranging between 0 and $+\infty$. For $k=0$ this model reduces to the SSRW while in the limit $k\to\infty$ only one edge is likely. The slowdown of the random walk with increasing k is shown in table 3 of [M]. It is clear that trapping configurations arise in the limit $k\to\infty$.

Obukhov, O (1984) interpreted the observations of Marinari et al. as follows:

he calculated the distribution function of the exit time from the simplest quasi-trap (two neighbouring sites) for finite k. He proved that above a critical k the

expected exit time is infinite. (In the two dimensional case the critical k value is equal to 6).

He concluded that the anomalous effects are caused by these local quasi-traps which, of course, do not occur in a non-degenerate model.

An intriguing question is really whether traps analoguous to the ones of B-D(1986) can be localized sufficiently often in a Bernoulli environment (it is worth stressing that we do not understand exactly any of the terms "trap", "analoguous", "localized", "sufficiently often" of the last clause but a clever simulation may help to imagine the geometry). Our aim, however, is more modest: we want to decide whether non-Wiener behavior can occur at all in a Bernoulli environment without local traps. To get a more characteristic answer we even drop the non-degeneracy condition.

Our simple model is motivated by the situation described above. We have studied the "complementary" case to the limit $k=\infty$: for each site there are three possible edges while the fourth one is forbidden.

More formally: let f be a random variable assigned independently to each site with $(P(f=e_i)=1/4 \quad (i=1,\ldots,4)$ where $e_i (i=1,\ldots,4)$ are the unit vectors in $\mathbb{Z}^2$ and the

transition probability to each of the four neighbouring sites is

$$p_i = \begin{cases} 0 \text{ if } e_i = f \\ 1/2 \text{ if } e_i = -f \\ 1/4 \text{ otherwise.} \end{cases}$$

An elementary combinatorial consideration shows that in our model there are no finite (local) trapping configurations. It is obvious (see figure 1.) that the boundary of the convex hull of any possible finite trap should contain at least one site $\underline{x}$ through which there exists a supporting line of the convex hull not parallel with the co-ordinate axes. Now there are two edges at $\underline{x}$ showing outside from the convex hull and both of them cannot be forbidden.

Figure 1.

On the other hand arbitrary large forbidden regions, sort of scatterers (see figure 2.) can exist.

Figure 2.

The configuration of figure 3. can be considered as an almost trapping one.

Figure 3.

The probability of such a configuration is proportional to exp(perimeter of the area log(1/4)). So the expected exit time is finite.

In point 2 we briefly describe the statistical method used in our evaluation of the computer results. The essential argument for using statistics based upon the first exit time from a square is that this ensures a maximal exploitation of the piece of the environment saved in the memory. Point 3 summarizes the numerical results while point 4 draws the conclusions.

2. Statistical Consideration

Our task is to test the hypothesis H_o that the normed trajectory of the moving particle $A^{-1/2}x(At)$ obeys the functional central limit theorem.
Our statistical considerations follow the concepts of well known textbooks (cf. e.g. L(1959)). There are different standard methods for testing the normality of $t^{-1/2}x(t)$: e.g. methods of semi-invariants, chi-square test, Kolmogorov-Smirnov test, Sarkadi-test, etc. The above methods require numerical experiments generating trajectories, which leave the given random

environment (region) in the course of the observation with a small but not negligible probability.

For a region of a given size and a fixed time necessary for the standard tests, the typical trajectories leave the region earlier than the admissible observation time for the standard statistical methods. In order to exploit the information supplied by the typical trajectories of the moving particle until leaving the region, we propose the statistical investigation of the distribution of the exit time.

Let $T_a = \min \{ k \mid k \geq 1, x(k) \in \mathbb{Z} \setminus [-a, a], x(0) = 0 \}$

It is well known that for a symmetric random walk with diffusion coefficient δ on the lattice $\mathbb{Z}$ the following result of Erdős and Kac (E-K(1946)) is true: the exit time of the scaled random walk is asymptotically equal to the exit time of the Wiener process, i.e.

$$\lim_{N \to \infty} P_0 \left[T_N > t \frac{N^2}{\delta^2} \right] = 1 - F(t), \quad t \geq 0, \qquad (*)$$

where

$$F(t) = 1 - \frac{4}{\pi} \sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1} e^{-\frac{\pi^2}{8}(2k+1)^2 t}$$

For a two dimensional SSRW the exit time $T_{a,b}$ from the region $[-a,a] \times [-b,b]$ under the condition that $\underline{z}(0) = \underline{0} \in \mathbb{Z}^2$ is by definition $T_{a,b} = \min \{ {}_x T_a, {}_y T_b \}$ where $x(t)$ and $y(t)$ are the two coordinates of $z(t)$.

In the SSRW case the ${}_x T_a$, ${}_y T_b$ are asymptotically independent.

So $z^{T}_{a,b}$ can be regarded as the minimum of two independent random variables with distributions $F(\frac{\sigma^2 t}{a^2})$ and $F(\frac{\sigma^2 t}{b^2})$, respectively where $\sigma^2 = \frac{1}{2}$. In order to test the non-diffusive character of our RWiRE, first we have to test the independence of the exit times x^{T}_{a}, y^{T}_{b}. If the independence is acceptable, then we can test whether the distribution function of the exit time (defined by (*)) is of Erdős-Kac type for some σ.

3. Numerical results

The hypothesis that $A^{-1/2}x(At)$ $(A\to\infty)$ is asymptotically a 2-D Wiener process and the invariance of its 2x2 covariance matrix under the rotation by $\pi/2$ involve that the x and y components are asymptotically independent Wiener processes with a common unknown variance σ^2. Thus we should test the hypothesis H_o: $\{ \underline{z}^{T}_{a,a} = \min\{ x^{T}_{a}, y^{T}_{a} \}$ has a distribution function of the form $G_{\sigma^2}(t) := 1-(1-F(\frac{\sigma^2 t}{a^2}))^2$ for some $\sigma^2\}$. Notice that H_o is a composite hypothesis, allowing that σ^2 varies in $(0,\infty)$.

Having a large sample (our simulation considers 50 independent random environments of size 401x401 with 200

random walks in each of them) there are several possibilities to estimate the parameter σ^2 of the distribution function $G_{\sigma^2}(t)$. One of them is the "minimum χ^2-estimate": for a given partition of the t axis say $(-\infty, t_1]$ $(t_1, t_2]$,..., , (t_k, ∞) $(k > 1)$ one looks for a value σ^2 for which the χ^2 distance between the empirical probabilities

$$P_i = P\{{}_zT_{200,200} \in (t_i, t_{i+1}]\} \quad (t_o = -\infty, \; t_{k+1} = +\infty)$$

and the theoretical probabilities $Q_i = G_{\sigma^2}(t_{i+1}) - G_{\sigma^2}(t_{i+1})$ —— is minimal. Another way is to minimize the J-divergence $(P_i - Q_i) \cdot \lg \frac{P_i}{Q_i}$ between the aforementioned two distributions.

First we wanted to see how strongly the estimate depended on the chosen distance and partition. For both distances and all reasonable partitions we obtained $\sigma^2 \in [0.39, 0.40]$. As an example Table 1 shows the dependence of the χ^2-distance on σ^2 for the partition consisting of 10 cells equiprobable with respedt to $G_{0.5}(t)$.

σ^2	χ^2	σ^2	χ^2
0.10	9780635	0.44	271.30
0.15	215139.5	0.45	389.75
0.20	29176.89	0.50	1290.22
0.25	7564.57	0.55	2695.83
0.30	2137.00	0.60	4652.31
0.35	399.96	0.65	7262.09
0.36	247.53	0.70	10680.94
0.37	139.03	0.75	15124.88
0.38	69.05	0.80	20883.84
0.39	33.35	0.85	28341.40
0.40	28.51	0.90	38003.29
0.41	51.84	0.95	50533.43
0.42	101.15	1.00	66804.00
0.43	174.76		

square: 401x401
df=9

Table 1.

For testing the hypothesis H_o we used χ^2 statistics with two partitions consisiting of 10 and 33 cells equiprobable with respect to $G_{o.4}(t)$. Our conclusion is that for no value σ^2 the shape of the empirical exit time distribution is acceptable as $G_{\sigma^2}(t)$. Tables 2 and 3 show the behavior of χ^2 around its minimum for 10 and 33 cell partitions, respectively.

σ^2	χ^2	σ^2	χ^2
o.385	52.856	0.395	41.955
0.386	50.342	0.396	42. 557
0.387	48.152	0.397	43.458
0.388	46.283	0.398	44.655
0.389	44.732	0.399	46.146
0.390	43.496	0.400	47.929
0.391	42.572	0.401	50.002
0.392	41.959	0.402	52.363
0.393	41.653	0.403	55.010
0.394	41.652	0.404	57.941

square: 401x401
df=9

Table 2.

σ^2	χ^2	σ^2	χ^2
0.385	93.720	0.395	86.677
0.386	91.426	0.396	87.867
0.387	89.492	0.397	89.391
0.388	87.921	0.398	91.242
0.389	86.700	0.399	93.428
0.390	85.827	0.400	95.940
0.391	85.313	0.401	98.775
0.392	85.137	0.402	101.932
0.393	85.311	0.403	105.418
0.394	85.823	0.404	109.216

square: 401x401
df=32

Table 3.

Though, of course, our tables only give a discretization of the function $\chi^2(\sigma^2)$ the smoothness of this function implies that the conclusion is true at any reasonable significance level (for 9 and 32 degrees of freedom the critical χ^2 values at level 99.95 % are 29.666 and 64.995, respectively).

The same statistics were computed for 201x201 and 301x301 squares (parts of the same environments as for the 401x401 case). Tables 4,5,6 and 7 show the behaviour of the χ^2 values around their minima for these sizes resulting the same conclusion as before.

σ^2	χ^2	σ^2	χ^2
0.390	105.314	0.395	102.802
0.391	104.151	0.396	103.277
0.392	103.321	0.397	104.076
0.393	102.820	0.398	105.191
0.394	102.649	0.399	106.623

square: 201x201
df=9

Table 4.

σ^2	χ^2	σ^2	χ^2
0.387	178.354	0.392	175.174
0.388	177.958	0.393	175.684
0.389	175.930	0.394	176.554
0.390	175.303	0.395	177.803
0.391	175.050	0.396	179.413

square: 201x201
df=32

Table 5.

σ^2	χ^2	σ^2	χ^2
0.388	50.296	0.393	48.447
0.389	49.294	0.394	49.016
0.390	48.611	0.395	49.891
0.391	48.243	0.396	51.071
0.392	48.188	0.397	52.555

square: 301x301
df=9

Table 6.

σ^2	χ^2	σ^2	χ^2
0.388	110.469	0.393	108.205
0.389	109.287	0.394	108.828
0.390	108.479	0.395	109.788
0.491	108.036	0.396	111.103
0.392	107.942	0.397	112.760

suare: 301x301
df=32

Table 7.

Remark. The invariance of the covariance matrix of $A^{-1/2}\, x(At)$ under the rotation by $\pi/2$ implies that even in the non-Wiener case the components are uncorrelated. Nevertheless it may happen that the two components are asymptotically dependent. For the sake of completeness we tested the independence of the exit times $_xT_a$ and $_yT_b$ for a given 101x101 environment based on 1000 trajectories - again using χ^2 test with 5x5 cells (4x4 degress of freedom). For equiprobable - with respect to $F(\frac{0.5}{50^2}\, t)$ - cells we have got χ^2 =11.64. This value is between the 10 % and 90 % quantiles of χ^2_{16} (9.312 and 23.542). Consequently this value is consistent with asymptotic independence of the components.

4. Conclusions

(i) For the RWiRE introduced in section 1, the axial components of the random walk are asymptotically independent:

(ii) Statistical evaluations based upon the first exit time of the random walk from a square show that asymptotically, this RWiRE is not a Wiener process.

(iii) The RWiRE considered is degenerate but we expect that, by applying our numerical and statistical methods, the diffusivity of non-degenerate RWiRE's can also be checked.

(iv) At present, we are unable to use our numerical method for suggesting a rigorous proof for the non-diffusivity of this or other Bernoulli RWiRE.

(v) Our method should also be useful to check the diffusivity of other processes where memory constraints may arise, e.g. to check whether, on R^1, the trajectory of a Brownian point particle of fixed mass M interacting with an ideal gas of identical point particles of mass 1 through elastic collisions is asymptotically Wiener or not (cf. O-R-D(1986) and Sz-T(1986)).

We remark that recent numerical results by Sinai's group suggest a non-diffusive behavior (personal communication).

5. Technical Remarks

We have used the RNDM2 random number generator from the CERN LIBRARY.

On the IBM 3330 under OSVS1 the required time for the simulation was 20^h CPU. The length of a trajectory was bounded by 200000 steps. For every sample the time and site of the exit from the 401x401 square are saved. So future proposed statistics can be computed.

The reliability of the numerical results obtained was checked carrying out all methods for the SSRW, too.

The results are in a good agreement with the theory e.g. the empirical distribution of the exit time fits to the theoretical one:the chi-square value for 9 degrees of freedom, was 8.456, which is inside the interval between the 40 % and 60 % quantiles.

6. Acknowledgement

D.S. expresses his warmest gratitude to Laurie Snell, John Kemeny, Denis Devlin and all colleagues at the Mathematics Department of Dartmouth College for teaching him the technique and art of computing and for providing him with sophisticated facilities. In fact, a preliminary version of the program was written on a Fat Mac and then transferred to an IBM XT through the main frame of Dartmouth Time Sharing System.

Thanks are due to the referees for their valuable remarks.

REFERENCES

B-D (1986) M. BRAMSON - R. DURRETT:
Random Walk in Random Environment: A Counter-example.Manuscript. pp.9., 1986.

E-K (1946) P. ERDŐS - M. KAC. On certain limit theorems of the theory of probability, Bull. Am. Math. Soc. 52, 292-302 (1946)

F (1984) D.S. FISHER. Random walks in random environments. Phys. Rev. A. 30, 960-964 (1984)

LU (1984) J.M. LUCK: A numerical study of diffusion and conduction in a 2D random medium, J. Phys. A. 17. 2069 (1984)

L (1959) E.L. LEHMAN: Testing Statistical Hypotheses, Wiley, New York (1959)

M (1983) E. MARINARI - G. PARISI - D. RUELLE-P. WINDEY. On the Interpretation of $\frac{1}{f}$ Noise. Communications in Mathematical Physics 89:1 (1983)

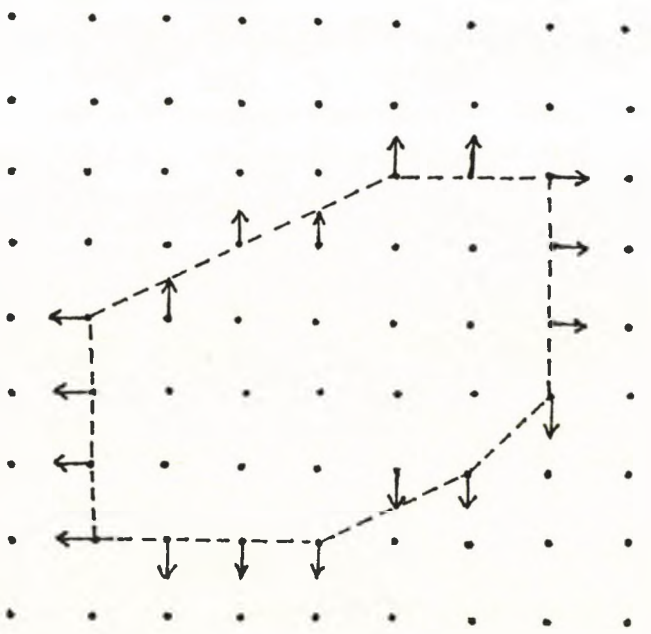
O (1984) S.P. OBUKHOV. Random Walk in Inhomogeneous Medium, Letters to J. of Emp. and Theor. Physics. 39:21 (1983)

O-R-D (1986) E. OMERTI, M. RONCHETTI, D. DÜRR.
Numerical evidence for mass dependence in diffusive behaviour of the "Heavy Particle" on the line.
J. Stat. Phys. 44:339 /1986/

P (1984) L. PELITI. Self avoiding walks. Phys. Rep. 103, 225-231 (1984).

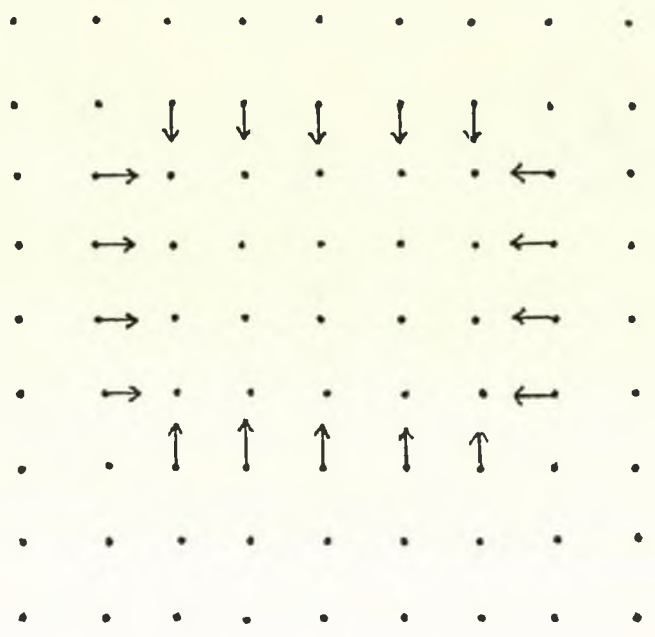
S (1982) Ya. G. SINAI. Limit behaviour of one-dimensional random walk in random environment. Theor. Prob. and Appl. (in Russian) 27 . 247-258 (1982)

Sz- T (1986) D SZÁSZ - B. TÓTH. Towards a Unified Dynamical Theory of the Brownian Particle in an Ideal Gas. Commun. in Math. Physics pp. 31 (submitted)



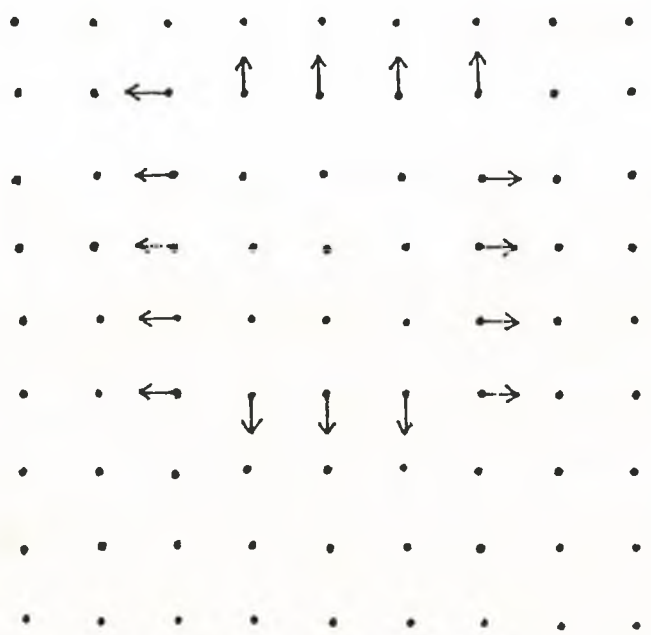
↑ forbidden direction

figure 1.



↑ forbidden direction

figure 2.



↑ forbidden direction

figure 3.

Caption of figures

Figure 1. Non-existence of finite traps

Figure 2. Example of a finite scatterer

Figure 3. Example of an almost trap

MTA SZTAKI Tanulmányok 202/1987 pp. 99-106

МИНИМАЛЬНЫЕ НОМЕРА В ПРЕДПОЛНЫХ НУМЕРАЦИЯХ

Маранджян Г.Б.

В настоящей работе мы исследуем множество минимальных номеров объектов в предполных нумерациях. Мы получим результаты достаточно общего характера, которые позволяют вывести в качестве следствий дескриптивные характеристики множеств минимальных (относительно некоторого упорядочения) номеров объектов, имеющих различную математическую природу. Среди полученных следствий будут как известные результаты, полученные как данным автором, так и другими авторами, а также новые результаты.

Через $\mathcal{N}$ будем обозначать множество натуральных чисел 0, 1,2,... . Будем предполагать, что задана некоторая геделевская нумерация частично рекурсивных функций (ЧРФ) $\{\varphi_i\}$. Через $\mathcal{T}_i$ будем обозначать область значений одноместной ЧРФ с номером i .

Определение 1. Полное рекурсивное отношение квазипорядка $\preccurlyeq$ на натуральных числах будем называть приемлемым упорядочением.

Напомним, что нумерацией произвольного множества S называется отображение ν множества $\mathcal{N}$ на множество S .

Понятия нумерованного множества и предполной нумерации будем понимать как в монографии Ю.Л.Ершова [1].

Определение 2. Множество A будем называть стабильным относительно приемлемого упорядочения $\preccurlyeq$, если существует такая общерекурсивная функция (ОРФ) f , что выполнено условие

$$\forall x \left[\mathcal{T}_x \subseteq A \Rightarrow \forall y \left[y \in \mathcal{T}_x \Rightarrow y \preccurlyeq f(x) \right] \right].$$

Существование бесконечного стабильного множества легко будет следовать из теоремы 1. Нетрудно заметить, что бесконечное множество, стабильное относительно естественного упорядочения натуральных чисел, есть сильно эффективно иммунное множество в смысле определения Т.Г.Мак-Лафлина [8].

Можно показать, что существует бесконечное стабильное множество, которое не только не является сильно эффективно иммунным, но даже не иммунно. Однако, это сопровождается приемлемым упорядочением, неестественными с точки зрения теории объемной сложности.

Введем обозначение. Пусть $\mathcal{G}=(S,\nu)$ есть нумерованное множество и $\preccurlyeq$ - приемлемое упорядочение. Тогда

$$M^{\mathcal{G}}_{\preccurlyeq}=\left\{a \mid \forall b\,[\nu b=\nu a \Rightarrow a\preccurlyeq b]\right\}$$

Смысл множества $M^{\mathcal{G}}_{\preccurlyeq}$ очевиден - это множество минимальных в смысле отношения $\preccurlyeq$ номеров равных объектов из S.

Теорема I. Если $\mathcal{G}=(S,\nu)$ - предполно нумерованное множество, то при любом приемлемом упорядочении $\preccurlyeq$ $M^{\mathcal{G}}_{\preccurlyeq}$ стабильно относительно $\preccurlyeq$.

Пусть, действительно, $\mathcal{G}=(S,\nu)$ - предполная нумерация и $\preccurlyeq$ - приемлемое упорядочение. Определим частично рекурсивную функцию α следующим образом. Вычисление ведем по шагам. Для вычисления $\alpha(x,y)$ в каждый момент времени t делаем попытку вычислить $\varphi_x^t(0),\varphi_x^t(1),\ldots,\varphi_x^t(t)$ и параллельно проверяем, выполнено ли отношение $y\prec\varphi_x^t(i)$ хотя бы для одного значения i . Как только это отношение выполнится, хотя бы для одного значения i, вычисления прекращаем и полагаем значение $\alpha(x,y)$ равным наименьшему из таких i . Определим теперь ЧРФ β следующим образом:

$$\beta(x,y)\simeq\varphi_x(\alpha(x,y)). \qquad (1)$$

Очевидно, что для любых x,y имеем:

$$\beta(x,y)\downarrow \Rightarrow \beta(x,y)\succ y. \qquad (2)$$

Как известно [I] , если $\mathcal{G}=(S,\nu)$ - предполная нумерация, то всякой ЧРФ γ найдется такая ОРФ δ , что если пара $(p,\delta(p))$ принадлежит области определения функции γ , то $\nu\gamma(p,\delta(p))=\nu\delta(p)$. Возьмем в роли функции γ функцию β , а соответствующую ОРФ обозначим через ε . Итак, если $(p,\varepsilon(p))$ принадлежит области определения функции β , то имеем:

$$\nu\beta(p,\varepsilon(p))=\nu\varepsilon(p). \qquad (3)$$

Докажем теперь, что множество $M^{\sigma}_{\preccurlyeq}$ стабильно. В самом деле, пусть $\mathcal{T}_a \in M^{\sigma}_{\preccurlyeq}$. Убедимся в том, что

$$\forall z \left[z \in \mathcal{T}_a \Rightarrow z \preccurlyeq \varepsilon(a) \right]. \qquad (4)$$

Если $\mathcal{T}_a$ пусто, то (4) очевидно. Итак, пусть $\mathcal{T}_a$ непусто. Допустим, что (4) не имеет места. Поскольку $\mathcal{T}_a$ есть область значений непустой функции с номером a, то существует такое натуральное число q, что $\varphi_a(q)$ определено и $\varepsilon(a) \prec q$. Тогда, как легко видеть из определения функций α и β, будем иметь $\alpha(a, \varepsilon(a))\downarrow$ и $\beta(a, \varepsilon(a))\downarrow$. Следовательно, используя (2), получим

$$\varepsilon(a) \prec \beta(a, \varepsilon(a)). \qquad (5)$$

Но $\beta(a,\varepsilon(a)) \in \mathcal{T}_a$, следовательно, $\beta(a, \varepsilon(a)) \in M^{\sigma}_{\preccurlyeq}$, а последнее соотношение противоречит тому, что $\nu\beta(a,\varepsilon(a)) = \nu\varepsilon(a)$ и одновременно имеет место (5). Из полученного противоречия следует, что (4) верно. Заметив, что a было выбрано произвольно, завершаем доказательство.

Таким образом, для любого приемлемого упорядочения $\preccurlyeq$ и для любого предполно нумерованного множества $\mathcal{G}$ рекурсивно перечислимые подмножества множества $M^{\sigma}_{\preccurlyeq}$ в некотором смысле конструктивно ограничены. Этот результат носит более общий характер, чем теорема 1 из [3].

Как отмечено в работе А. Лейера [9], А.Багчи в работе, недоступной автору данной работы, доказал, что множество минимальных (относительно функции объема $\mathcal{S}$, такой, что по всякому n канонический номер конечного множества $Q^{\mathcal{S}}_n = \{a \mid \mathcal{S}(a) \leqslant n\}$ может быть эффективно найден) номеров ЧРФ геделевой нумерации иммунно. Как легко видеть, из приведенной выше теоремы следует, что определив $a \preccurlyeq b$ как $\mathcal{S}(a) \leqslant \mathcal{S}(b)$ с функцией объема $\mathcal{S}$, удовлетворяющей лишь условию конечности $Q^{\mathcal{S}}_n$ и без требования, чтобы канонический номер конечного множества $Q^{\mathcal{S}}_n$ можно было находить эффективно, можно заключить, что соответствующее множество минимальных номеров иммунно.

Пусть в дальнейшем изложении $\mathcal{S}$ есть произвольная ОРФ такая, что множества $\{a \mid \mathcal{S}(a) \leqslant n\}$ конечны. Определим $M^{\sigma}_{\preccurlyeq,f} = \{a \mid \forall a = \forall b \Rightarrow \mathcal{S}(a) \preccurlyeq f(\mathcal{S}(b))\}$, где f произвольная ОРФ. Будем

говорить, что a находится в f-окрестности числа b, если $S(a) \leqslant f(S(b))$.

Следствие I. Для любого предполно нумерованного множества $\mathcal{G} = (S, \nu)$ и любой монотонно возрастающей ОРФ f всякий алгоритм, перечисляющий числа из f-окрестностей минимальных относительно $\preceq$ номеров объектов множества S в нумерации ν, конструктивно ограничен.

Доказательство - очевидно.

Таким образом, если даже пытаться алгоритмически порождать не минимальные номера, а числа из их "вычислимых окрестностей", то и в этом случае решение не выходит за рамки порождения конечного числа объектов, то есть задача не имеет эффективного решения ни для какого бесконечного фрагмента общей задачи.

Должным образом обпределяя объекты нумеруемого множества (в частности - равенство объектов), из доказанной выше теоремы мы получаем (и порой - усиливаем) ряд результатов, полученных различными способами и для различных систем объектов. Покажем это на ряде примеров.

I. Ю.Хартманис и Т.Бейкер [7] доказали иммунность множества минимальных номеров (среди эквивалентных) следующих классов машин Тьюринга:

а) $VPTM = \{M_i \mid FS \vdash \exists k [T_i(n) \leqslant k + n^k]\}$,

где FS - формальная система, непротиворечивая и достаточно богатая, возможности которой позволяют формулировать и доказывать элементарные факты о машинах Тьюринга и вычисления с их помощью.

Нетрудно проверить, что можно построить такую нумерацию этих машин, что будут удовлетворены условия теоремы I, более того, при естественных определениях меры сложности множество минимальных по объему машин окажется сильно эффективно иммунным.

б) $VPLM = \{M_i \mid FS \vdash [M_i$ всюду определена и $L(M_i) \in PTIME]\}$.

Аналогично тому, как это можно сделать в пункте а), доказывается сильная эффективная иммунность.

в) $PTM = \{M_i \mid \exists k [T_i(n) \leq k + n^k]\}$.

Поскольку этот случай является факторизацией случая а), то благодаря этому множество минимальных описаний строго эффективно-иммунно.

В [7] приведен ряд других классов, рассмотрение которых мы опускаем, так как и они подпадают под условия теоремы I, либо являются результатами факторизации случаев, уже подпавших под эти условия.

2. Приведем результат, полученный Б.А.Кушнером в работе [2]. Пусть h - произвольная общерекурсивная функция, $\{F_i\}$-геделева нумерация алгоритмов из N в множество рациональных чисел. Обозначим через D множество конструктивных действительных чисел. Если x - конструктивное действительное число, то F_i есть его вычисление, если при любом n имеет место

$$|x - F_i(n)| \leq 2^{-n}.$$

Пусть $D_{h,n}$ есть множество тех конструктивных действительных чисел, любое вычисление F_i которых таково, что $h(i) \geqslant n$. Б.А.Кушнер доказал, что для любых общерекурсивных h, f и алгоритма ψ типа $N \to D$ если $\forall n [\psi(n) \in D_{h,f(n)}]$, то можно построить число n_f такое, что для всех i имеет место

$$f(i) \leq n_f$$

Нетрудно убедиться, что этот результат можно получить из следствия I.

3. В работе настоящего автора [4] показано, что множество минимальных номеров из классов доказуемо эквивалентных частично рекурсивных функций $(\varphi_i \sim \varphi_j \equiv_{Df} [PA \vdash [\varphi_i = \varphi_j]])$ сильно эффективно иммунно.

Из теоремы I этот результат следует с усилением: можно ослабить требования к мере объемной сложности.

4. Кроме приведенных примеров условиям теоремы I удовлетворяет ряд множеств минимальных в том или ином смысле описаний объектов различной природы и, естественно, при различных конкретизациях отношения $\lesssim$. Во всех случаях, когда отношение определяется с помощью функций, принимающих каждое из своих значений лишь конечное число раз, притом не требуется, чтобы это проверялось эффективно, соответствующее множество оказывается эффективно иммунным. Вот перечень ряда таких сис-

тем.

а) Постовские нумерации рекурсивно перечислимых множеств;

б) Частично рекурсивные функции конечного суппорта;

в) Функционалы конечных типов;

г) Монотонные рекурсивные алгебры на ω Я.Московакиса [10];

д) Диофановы уравнения, эквивалентные по представляемым ими рекурсивно перечислимыми множествами по Ю.В.Матиясевичу;

е) Формулы интуионистской, а также классической формальной арифметики и ее аксиоматизируемых расширений;

ж) Термы λ -исчисления;

з) R -позитивные Σ-формулы теории допустимых множеств Й.Барвайза [6].

Приведенный выше список может быть легко продолжен.

Литература

1. Ершов Ю.Л. Теория нумераций. Наука, М., 1977.
2. Кушнер Б.А. Сложно-вычислимые действительные числа. Zeitschr. Math. Logik und Grundl. d. Math.,19,447-452,1973.
3. Маранджян Г.Б. Об алгорифмах минимальной сложности.ДАН СССР, 213 №4, 787-788, 1973.
4. Маранджян Г.Б. Минимальные по объему алгоритмы среди доказуемо эквивалентных. Прикладная математика, вып.2,75-81, Изд-во ЕГУ, Ереван,1983.
5. Матиясевич Ю.В. Диофантовы множества. Успехи математических наук, т.ХХУП, вып.5, 185-222, 1972.
6. Barwise J. Admissible Sets and Structures. Springer-Verlag, Berlin, Heidelberg, New York, 1975.
7. Hartmanis J., Baker T.P., Relative succinctness of languages and separation of complexity classes. Math. Foundation of Computer Sci., Lecture Notes in Computer Sci., v. 74, Springer-Verlag, Berlin, 1979, 70 - 88.
8. McLaughlin T.G. On a class of complete simple sets. Canadian Math. Bull., ', No. I, 1965, 33 - 37.
9. Meyer A., Program size in restricted programming languages. Information and Control, 21, No. 4, 1965, 382-394.

IO. Moschovakis Y. On the basic notions in the theory of induction. Logic, Foundations of Mathematics, and Computability Theory. D.Reidel Publ. Comp., Dodrecht-Holland, I977, 207 - 236.

Minimal numbers in pre-computer numberings

G.B. Marandžjan

Summary

In the paper the set of minimal numbers in pre-complete numbering of objects is studied. The results are of a sufficiently general nature, so that descriptive characteristics of sets of minimal numbers can be derived as corollaries from them, for objects of different mathematical nature.
Among corollaries there are known results of the author or other authors, as well as somw new ones.

MTA SZTAKI Tanulmányok 202/1987 pp. 107-109

CAD Tool Manager: an Alternative System Structure in Electronic CAD

by Levente L. Máté
Computer and Automation Institute, Hungarian Academy of Sciences
(H-1502 Pf. 63, Budapest, HUNGARY)

CAD systems are decomposed to CAD tasks. A CAD task solves a functionally well defined design problem (e.g. placement, routing etc.). CAD tasks are working on instances. An instance is a given input for a given CAD task (e.g. a placement of a logic circuit is an instance for a router task when its routing on an uncommitted logic array is to be found).

The following statements hold for almost all tasks of Computer Aided Design in Electonics:

- the formalized CAD task fits weakly to the engineering problem to be solved;
- the goal function of the CAD task is an oversipmlified model of the real aim of the engineering problem;
- the formalized CAD task is likely to have super-polynomial complexity;
- various heuristic procedures have been developed and implemented to solve CAD tasks;
- individual heuristic procedures have different efficiency when working on different instances;
- individual heuristic procedures have different efficiency when working on different stages of the same instance;
- some heuristic procedures can continue the work of others on the same instance.

Engineers using CAD tasks have a certain knowledge which they apply when dealing with given instances.
This knowledge is used

- to choose the best fit heuristic procedure to begin the work on the given instance;
- to supply the appropriate input parameter values for the chosen heuristic procedure;
- to evaluate the results of the applied heuristic procedure;
- to decide upon the means of continuing work on the given instance;
- to collect experience on the solution achieved.

CAD Tool Manager (CTM) is an expert system which takes over most of the above mentioned duties of the engineer using CAD. It is not intended to substitute existing heuristic procedures, neither to develop new ones. On the contrary, it applies them as tools but above that it organizes and controls their work too.

CTM itself is organized as a free market. Each heuristic procedure has an expert system above itself which acts on this market as an Enterprise. The Inviter is an actor in the market too. It makes a thorough analysis of the stage of the solution having been achieved and invites tenders for continuing work on the instance. By then, each Enterprise gives an offer characterizing the chances of its heuristics if taking over the next step. The Inviter analyses the offers and finally chooses the best one. In the evaluating process the Inviter gets advice from its Advisor. The Advisor is also an expert dealing with case stories which were collected in the life of the given application of CTM. Case stories are compressed histories about instances solved earlier by CTM and collected by the Librarian which is also an expert system.

Knowledge bases of Enterprises are based upon a-priori engineering knowledge about the heuristic procedure represented. They contain knowledge which is evaluated on parameter values collected about the instance under work and the stage of the solution achieved. These parameters are extracted by the heuristic procedures themselves even in today solutions.

The knowledge base of the Inviter develops by the use of the system. The most important part of it is the credibility measure of the individual Enterprises in similar situations. Obviously initial credibility is given to the system by human expert when starting its usage. It is very important to keep the sum of credibility in a constant level, thus not allowing the whole system to go on strike.

The Librarian is a very dumb expert. It mantains a log on each instance taking notes at heuristic procedure changes. This contains a brief description of the stage previously achieved and the offer of the chosen Enterprise.

The Advisor is given questions by the Inviter like the following: What is the credibility of Enterprise i in situation j? Then Advisor makes a query to Librarian for case story notes on Enterprise i in situations similar to j. Evaluating the available notes Advisor gives an answer to the Inviter.

An implementation of CTM is under development for test design purposes.

CAD Tool-Manager: an alternative system structure in electronic CAD

L.L. Máté

Summary

The paper summarize the tasks and duties of CAD in electronics. CAD Tool Manager (CTM) is an expert system which takes over most of the duties mentioned. In the paper a short discussion of basic features of CTM can be found.

N – объем партии;

n – количество брака;

$n_л$ – количество ложного брака;

[illegible] – количество необнаруженного брака;

[illegible] – общее количество технологических операций;

[illegible] – общее количество плановых статистических наблюдений за ходом i-ой операции технологического процесса производства генераторов;

[illegible] – общее количество исполнителей, выполняющих i-ую технологическую или контрольную операцию на [illegible] наблюдении;

[illegible] – стоимость накладных расходов на единицу производимого генератора на i-ой операции [illegible] ($i = 1, 2, \ldots,$ [illegible]);

[illegible]

MTA SZTAKI Tanulmányok 202/1987 pp. 111-130

ТЕОРЕТИКО-МОДЕЛЬНЫЙ ПОДХОД К ОЦЕНКЕ И УПРАВЛЕНИЮ КАЧЕСТВОМ ПРОДУКЦИИ В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ ПАРАМЕТРОВ ПРОИЗВОДСТВЕННОЙ СИСТЕМЫ

Мелконян А.Е., Барсегян В.А.,Айвазян А.А.

В настоящее время все более возрастает значение гарантирования некоторого высокого уровня качества продукции. Это, в первую очередь, как показала отечественная [I-6] и зарубежная [7-9] практика, можно обеспечить посредством автоматизации процесса управления качеством продукции, а также оперативным контролем этого процесса.

Воздействие случайных факторов (формальное количественное описание которых или невозможно, или настолько сложно, что не представляет практического интереса) на ход производственного процесса приводит к нарушению нормального хода процесса производства, и тем самым, снижает уровень качества производственного изделия. Поэтому при проектировании систем управления реальными объектами в ряде случаев представляется целесообразным отказаться от применения строго количественного анализа, так как не всегда известны функциональные связи между различными параметрами процесса для построения точной математической модели функционирования объекта управления. Поэтому им не всегда может быть поставлена в соответствие адекватная, и в то же время простая модель, позволяющая получить имеющее практическую ценность решение обычными методами теории управления. Естественно также, что если процесс не может быть смоделирован в рамках теории, то и не может быть достигнуто удовлетворительное управление этим процессом.

В связи с этим возникает необходимость в проведении комплексных исследований с целью общесистемной оценки уровня качества производимых изделий в условиях недостатка знаний об особенностях протекания процессов.

В теории систем фигурируют два понятия: понятие случай-

ности, восходящее к теории вероятностей и понятие "нечеткой формулировки", относящееся к теории нечетких множеств. В свою очередь недостоверность можно подразделить на "случайность" и "нечеткость формулировок". Однако в некотором смысле эти два понятия похожи друг на друга. Например, функцию принадлежности нечеткого множества, которая характеризует нечеткость высказывания о принадлежности элемента множеству, иногда рассматривают, как обобщение функции распределения вероятности, выражающей понятие случайности. Такая аналогия позволяет сформулировать задачу управления в условиях неопределенности в полном соответствии с постановкой задач обычной теории статистического управления [10].

Рассмотрим понятие вероятности нечеткого события. Нечеткое событие A определяют как борелевски измеримое нечеткое множество A с функцией принадлежности μ_A. Вероятность нечеткого события A определяют интегралом Лебега-Стильтьеса [17]

$$\tilde{P}(A) = \int_{R^n} \mu_A(x)\,dP = E(\mu_A) \qquad (1)$$

Таким образом вероятность нечеткого события A определяют с помощью функции $\mu_A(x)$, отражающей смысл события A и с помощью вероятности $P(x)$, которая определяет частоту появления X.

При описании и моделировании производственных процессов обычно используются три абстрактные операции (обработка, сборка и управление). Однако не все процессы можно подвести под эти операции. Поэтому возникает необходимость математического описания производственных операций с учетом их специфики. При математическом описании производственных операций над изделиями необходимо учитывать комплекс характеристик, описывающих свойства заготовок до поступления на данную операцию, во время операции и послеоперационного контроля их состояния.

В качестве объекта исследования в статье рассматривается технологический процесс производства генераторов мощностью до 100 кВт. Для оценки выхода годных генераторов, а также

учета влияния факторов, не поддающихся формализации, и которые в то же время являются причиной снижения процента годных генераторов в их общем количестве, предлагается нижеследующее математическое описание процесса производства генераторов. На каждом этапе производства целесообразно различать два несовместимых события $(A, \bar{A})$ производственно-технологического характера и одновременно учитывать четыре состояния выпускаемых генераторов: H_1, H_2, H_3, H_4. Здесь приняты следующие обозначения:

A – на i-ом этапе производственного процесса качество изготовления генератора[1] хорошее;

$\bar{A}$ – на i-ом этапе производственного процесса качество изготовления генератора плохое;

H_1 – качество генератора хорошее и он не забракован;

H_2 – качество генератора плохое, но он не забракован из-за необнаружения неисправностей;

H_3 – качество генератора плохое и он забракован;

H_4 – качество генератора хорошее, но он забракован ложно.

Утверждение I. Если на предыдущем этапе процесса производства генератор находился в состоянии H_2, H_3, H_4, то он не может находиться в состоянии H_1 на последующем этапе. Это объясняется тем, что при изготовлении некачественного генератора или его бракования на предыдущем этапе, исключается возможность его нахождения в состоянии H_1.

Утверждение 2. Изменение состояния выпускаемых генераторов на рассматриваемом этапе зависит от его состояния на предыдущих этапах процесса производства.

Основываясь на вышеизложенном, процесс изменения состояния генераторов по этапам производства можно описать неоднородным марковским процессом. При этом вероятность $\tilde{P}(H_1, i)$ нечеткого события C, определяемого фразой "качество генераторов на i-ом этапе процесса производства приблизительно хо-

---

I) Здесь и далее на всех этапах технологического процесса (кроме сборочных операций) под "генератором" следует понимать соответствующий "задел генератора".

рошее" можно получить, имея выражение для оценки "четкой" вероятности выхода качественных генераторов на каждом последовательном этапе процесса производства

$$P(H_1,i) = P_i(H_1,0)\,P_i(k)\,P_i(A)$$

а также значения функции принадлежности $\mu_c(H_1,i)$ нечеткого события C .

Здесь приняты следующие обозначения:

$P_i(H_1,0)$ - вероятность того, что на i -ом этапе поступающие материалы качественны;

$P_i(k)$ - вероятность того, что на i -ом этапе контрольная операция произведена качественно;

$P_i(A)$ - вероятность того, что на i -ом этапе генератор изготавливается качественно.

Причем, вероятность $P(H_1,i)$ и функцию принадлежности $\mu_c(H_1,i)$ нечеткого события C можно задать в виде следующей таблицы.

Лингвистическая оценка качества	Плохое	Среднее (нехорошее - неплохое)	Хорошее
$P(H_1,i)$	P_1	P_2	P_3
$\mu_c(H_1,i)$	μ_1	μ_2	μ_3

Поскольку выпускаемые генераторы на каждом этапе процесса производства будут находиться в одном из состояний H_1, H_2, H_3, H_4, то в силу полноты групп событий [II] можно записать:

$$\tilde{P}_1(i) + \tilde{P}_2(i) + \tilde{P}_3(i) + \tilde{P}_4(i) = 1 \qquad i = 1,2,\ldots,n$$

где $\tilde{P}_1(i)$, $\tilde{P}_2(i)$, $\tilde{P}_3(i)$, $\tilde{P}_4(i)$ - соответственно являются вероятностями нахождения выпускаемого генератора в состояниях H_1, H_2, H_3, H_4 к концу i -го этапа.

Следовательно вероятность выхода генераторов хорошего качества может быть рассчитана на основании вероятностей $\tilde{P}_1(i)$, $\tilde{P}_2(i)$ $\tilde{P}_3(i)$, $\tilde{P}_4(i)$, которые на i -ом этапе процесса производства оп-

ределяются соответствующими вероятностями сохранения или перехода состояний, заложенных на $(i-1)$-ом этапе, т.е.

$$\tilde{P}(i) = E \cdot \tilde{T}(i)$$

где $\tilde{P}(i) = |\tilde{P}_1(i) \quad \tilde{P}_2(i) \quad \tilde{P}_3(i) \quad \tilde{P}_4(i)|$;

$E = |1.1\ 1\ 1|$

$$\tilde{T}(i) \quad \begin{vmatrix} \tilde{P}_{11}(i) & \tilde{P}_{12}(i) & \tilde{P}_{13}(i) & \tilde{P}_{14}(i) \\ \tilde{P}_{21}(i) & \tilde{P}_{22}(i) & \tilde{P}_{23}(i) & \tilde{P}_{24}(i) \\ \tilde{P}_{31}(i) & \tilde{P}_{32}(i) & \tilde{P}_{33}(i) & \tilde{P}_{34}(i) \\ \tilde{P}_{41}(i) & \tilde{P}_{42}(i) & \tilde{P}_{43}(i) & \tilde{P}_{44}(i) \end{vmatrix}$$

$\tilde{P}_{11}(i)$, $\tilde{P}_{22}(i)$, $\tilde{P}_{33}(i)$, $\tilde{P}_{44}(i)$ – вероятности сохранения генераторов на i-ом этапе в состояниях H_1, H_2, H_3, H_4, заложенных на $(i-1)$-ом этапе соответственно.

Остальные элементы матрицы $\tilde{P}_{ij}(i)$ – это вероятности перехода состояний генераторов на i-ом этапе из i-го состояния, заложенного на $(i-1)$-ом этапе процесса производства в j-й.

Учитывая реальные условия процесса производства генераторов и утверждение I, имеем:

$$\tilde{P}_{21}(i) = \tilde{P}_{31}(i) = \tilde{P}_{41}(i) = 0$$

$$\tilde{P}_{32}(i) = \tilde{P}_{31}(i) = \tilde{P}_{34}(i) = 0$$

$$\tilde{P}_{42}(i) = \tilde{P}_{43}(i) = \tilde{P}_{44}(i) = 0$$

$$\tilde{P}_{24}(i) = 0$$

Соответственно:

$$\tilde{T}(i) \quad \begin{vmatrix} \tilde{P}_{11}(i) & \tilde{P}_{12}(i) & \tilde{P}_{13}(i) & \tilde{P}_{14}(i) \\ 0 & \tilde{P}_{22}(i) & \tilde{P}_{23}(i) & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

Таким образом для нечеткой, в смысле Заде [I2], статистико-вероятностной оценки технологического процесса производства генераторов нам необходимо рассчитать вероятности

$$\tilde{P}_{11}(i),\ \tilde{P}_{12}(i),\ \tilde{P}_{13}(i),\ \tilde{P}_{14}(i),\ \tilde{P}_{22}(i),\ \tilde{P}_{23}(i).$$

Более того, следует также различать события B_1 и B_2, которые могут появиться после i контрольных операций, где

B_1 - выпускаемый генератор принят (состояния H_1, H_2);

B_2 - выпускаемый генератор забракован (состояния H_3, H_4).

Составим схему сохранения или перехода в i-ое состояние выпускаемых генераторов. Для того, чтобы сохранить качество выпускаемого генератора, заложенное на $(i-1)$-ом этапе, необходимо сохранить его и в интервале времени i-го этапа, т.е. необходимо, чтобы в течение i-го этапа производства произошло нечеткое событие A, вероятность $\tilde{P}_i(A)$ которого рассчитывается с помощью функции принадлежности μ_A и следующей формулы:

$$P_i(A) = P_{01}(i)\,P_0(i)$$

где $P_{01}(i)$ - вероятность, определяющая надежность i-го технологического этапа; $P_0(i)$ - вероятность, определяющая на i-ом этапе правильность выбора рабочих по своим квалификациям и рабочим местам.

Помимо этого, в процессе контроля должно появиться событие B_1 со следующими вероятностями:

$$P(B_1/A, i),\quad P(B_1/\bar{A}, i)$$

При этом генератор будет находиться в H_1-ом состоянии в конце i-го этапа с вероятностью

$$P_{11}(i) = P_i(A)\,P(B_1/A, i)\,P(H_1/H_1, i)$$

где $P(H_1/H_1, i)$ - вероятность нахождения генератора в H_1-ом состоянии на i-ом этапе, когда он на $(i-1)$-ом этапе находился в H_1-ом состоянии.

Аналогичным образом получаем вероятности:

$$P_{12}(i) = P_i(\bar{A})\,P(B_1/\bar{A}, i)\,P(H_2/H_1, i);$$

$$P_{13}(i) = P_i(\bar{A})\,P(B_2/\bar{A}, i)\,P(H_3/H_1, i);$$

$$P_{14}(i) = P_i(A)\,P(B_2/A,i)\,P(H_4/H_1,i);$$
$$P_{22}(i) = P_i(\bar{A})\,P(B_1/\bar{A},i)\,P(H_2/H_2,i);$$
$$P_{23}(i) = P_i(\bar{A})\,P(B_2/\bar{A},i)\,P(H_3/H_2,i)$$

Если нарушения качества выпускаемых генераторов характеризуется простейшим потоком, то можно записать:

$$P_{o1}(i) = 1 - \lambda_i t_i ,$$

где λ_i – интенсивность нарушения качества выпускаемых генераторов на i-ом этапе процесса производства;

t_i – время i-го этапа процесса производства.

Для расчета вероятностей $P(B_1/A,i), P(B_1/\bar{A},i), P(B_2/A,i), P(B_2/\bar{A},i)$ необходимо оценить величины условных вероятностей на отсутствие или наличие нарушений качества выпускаемых генераторов на i-ом последовательном этапе процесса производства. В силу полноты групп событий будем иметь:

$$P(B_1/A,i) = 1 - P(B_2/A,i);$$
$$P(B_2/\bar{A},i) = 1 - P(B_1/\bar{A},i).$$

Значения вероятностей $P(B_2/A,i)$ и $P(B_1/\bar{A},i)$ являются соответственно вероятностями ложных и необнаруженных нарушений качества выпускаемых генераторов. Поэтому можем принять:

$$P(B_2/A,i) \equiv P(B_2/B_1,i)$$
$$P(B_1/A,i) \equiv P(B_1/B_2,i)$$

Здесь же отметим, что необнаруженные нарушения качества выпускаемых генераторов могут быть вызваны следующими причинами: часть выпускаемых генераторов на i-ом этапе не контролируется вообще; в контролируемой части i-го этапа возникают необнаруженные нарушения выпускаемых генераторов из-за недостаточной точности средств контроля; часть параметров выпускаемых генераторов на i-ом этапе контролируется визуально.

Примем вероятность наступления события, заключающегося в том, что в контролируемых, неконтролируемых и визуально контролируемых частях i-го этапа одновременно существуют необнаруженные нарушения качества, близкой к нулю. Тогда в силу

полноты групп событий можно записать:

$$P_{но\,i}(B) = P_{но\,к\,i}(B) + P_{но\,\bar{к}\,i}(B) + P_{но\,вк\,i}(B),$$

где $P_{но\,к\,i}(B)$, $P_{но\,\bar{к}\,i}(B)$, $P_{но\,вк\,i}(B)$ – являются вероятностями существования необнаруженного нарушения качества выпускаемых генераторов, соответственно в контролируемых, неконтролируемых и визуально контролируемых частях.

В контролируемой части i-го этапа существование необнаруженного нарушения качества равносильно существованию необнаруженного нарушения качества при отказе или отсутствии средств контроля. В визуально контролируемой части i-го этапа существование необнаруженного нарушения качества выпускаемых генераторов вызвано недостоверностью средств контроля. Одновременно допустим маловероятным существование необнаруженного нарушения качества генераторов одновременно по нескольким контрольным парам.

Если известны плотности распределений контролируемого параметра выпускаемых генераторов и погрешности средств контроля по контролируемому параметру, то вероятность $P_{но\,i}(B)$ можно определить из нижеследующего выражения, как вероятность совместного появления двух независимых событий:

$$P_{но\,i}(B) = K_1 \int_{-\infty}^{\alpha_{1н}} f_1(i)\left[\int_{-\infty}^{\alpha_{1в}} f_2(i)\,db\right]da + K_2 \int_{-\infty}^{\alpha_{гн}} f_1(i)\left[\int_{-\infty}^{\alpha_{гв}} f_2(i)\,db\right]da, \quad (2)$$

где a – значение контролируемого параметра выпускаемых генераторов в соответствии с техническим условием (ТУ);

b – результат измерения контролируемого параметра генераторов;

$f_1(i)$ – плотность распределения контролируемого параметра генератора;

$f_2(i)$ – плотность распределения погрешности средств контроля по контролируемому параметру генератора;

$\alpha_{1н}, \alpha_{1в}$ – соответственно нижние и верхние значения допусков контролируемого параметра генераторов;

$\alpha_{2н}, \alpha_{2в}$ – соответственно нижние и верхние контрольные допуски по контролируемому параметру генераторов;

K_1, K_2 – коэффициенты для поля допуска контролируемого параметра;

$K_1 = K_2 = 1$ – коэффициент для двустороннего поля допуска контролируемого параметра;

$K_1 = 1, K_2 = 0$ – коэффициент для одностороннего поля допуска, если все значения контролируемого параметра находятся выше допустимой границы;

$K_1 = 0, K_2 = 1$ – коэффициент для одностороннего поля допуска, если все значения контролируемого параметра находятся ниже допустимой границы.

Ложные указания нарушения качества выпускаемых генераторов обусловлены, как правило, недостаточной точностью средств контроля качества в контролируемой части i-го этапа и ошибками визуального контроля части параметров выпускаемых генераторов на i-ом этапе. Будем считать, что вероятность наступления события, заключающегося в том, что в контролируемых и визуально контролируемых частях одновременно существуют ложные указания на нарушения качества выпускаемых генераторов, близка к нулю. Тогда в силу полноты групп событий имеем:

$$P_{ло\,i}(В) = P_{ло\,к\,i}(В) + P_{ло\,вк\,i}(В)$$

где $P_{ло\,к\,i}(В)$, $P_{ло\,вк\,i}(В)$ – вероятность ложного указания на нарушение качества выпускаемых генераторов соответственно в контролируемых и визуально контролируемых частях i-го этапа; $P_{ло\,i}(В)$ – безусловная вероятность ложного указания на нарушение качества выпускаемых генераторов на i-ом этапе.

Если плотности распределения значений контролируемого параметра выпускаемых генераторов, а также погрешности средств контроля известны наперед, то вероятность $P_{ло\,i}(В)$ можно рассчитать согласно (2):

$$P_{\text{ло}\,i}(B_2) = \int_{\alpha_{1H}}^{\alpha_{1B}} f_1(i)\left[K_1\int_{-\infty}^{\alpha_{2H}} f_2(i)\,db + K_2\int_{\alpha_{2B}}^{+\infty} f_2(i)\,db\right]da$$

Таким образом, на каждом i-ом этапе процесса производства можно оценить вероятности появления ложных и необнаруженных нарушений качества выпускаемых генераторов следующим образом:

$$P(B_1/\bar{A},i) = 1 - P(B_2/\bar{A},i) = \frac{1}{P(B_2,i)}\left\{K_1\int_{-\infty}^{\alpha_{1H}} f_1(i)\left[\int_{\alpha_{2H}}^{\alpha_{2B}} f_2(i)\,db\right]da + K_2\int_{\alpha_{1B}}^{+\infty} f_1(i)\left[\int_{\alpha_{2H}}^{\alpha_{2B}} f_2(i)\,db\right]da\right\} + \frac{P_{\text{но}\bar{A}\,i}(B_1) + P_{\text{но вк}\,i}(B_1)}{P(B_2,i)};$$

$$P(B_2/A,i) = 1 - P(B_1/A,i) = \frac{1}{P(B_1,i)}\left\{\int_{\alpha_{1H}}^{\alpha_{1B}} f_1(i)\left[K_1\int_{-\infty}^{\alpha_{2H}} f_2(i)\,db + K_2\int_{\alpha_{2B}}^{+\infty} f_2(i)\,db\right]da\right\} + \frac{P_{\text{ло вк}\,i}(B_2)}{P(B_1,i)};$$

где $P(B_2,i)$, $P(B_1,i)$ – вероятности неработоспособности и работоспособности выпускаемых генераторов на i-ом этапе технологического процесса производства, соответственно.

Получение точной численно-количественной оценки качества выпускаемых генераторов в непрерывно изменяющихся производственных условиях чрезвычайно сложно. В этих условиях применение аппарата теории нечетких множеств в сочетании с "обычной" теорией статистического управления представляется довольно целесообразным. Это связано с тем, что состояние процесса производства генераторов трудно определить по функциональным связям отдельных операций. Поэтому более целесообразно определение этого состояния через уровень динамического показателя качества продукции (ДПК).

Анализ результатов обработки данных пассивного статистического наблюдения в ходе производства генераторов, а также приведенное выше математическое описание процесса их производства, позволяют установить следующую иерархию факторов, влияющих на уровень ДПК:

- случайные воздействующие факторы;
- уровень достоверности контроля;
- возможности технологического процесса производства для изготовления качественных генераторов по всему технологическому циклу (по всем операциям);
- распределение исполнителей по своим рабочим местам.

Достижение высокого уровня обеспечивается в основном путем повышения качества и надежности каждой технологической операции процесса производства, т.е. минимизацией вероятности выхода брака ($P_{\delta i}$) при ограниченности затрат на его достижение.

Таким образом ДПК в процессе производства характеризуется тремя частными показателями: стоимостным, численно-количественным и вероятностным.

Рассмотрим упрощенную модель контроля производственных операций для двух случаев: сплошной 100%-й контроль и выборочный контроль. При 100-м контроле полная стоимость затрат на контроль и замену бракованных генераторов равна:

$$C_{\Pi} = \sum_{i=1}^{n} \left(N_i C_{Ki} + K_i \tilde{P}_i(\bar{A}) N_i C_{3i} \right), \qquad (3)$$

где N_i – общее количество генераторов, поступающих на i-ую операцию; C_{Ki} – стоимость контроля генераторов после i-ой операции; K_i – доля брака генераторов, пропущенных при контроле; C_{3i} – затраты на ремонт и замену бракованных генераторов; n – число операций.

Аналогичным образом для выборочного контроля будем иметь:

$$C_{\Pi} = \sum_{i=1}^{n} \left\{ m_i C_{Ki} (1 - P_{mi}) / P_{mi} + P_{mi} \left[m_i C_{Ki} + \right.\right.$$
$$\left.\left. + (N_i - m_i) \tilde{P}_i(\bar{A}) C_{3i} + m_i K_i \tilde{P}_i(A) C_{3i} \right] \right\} \qquad (4)$$

где m_i – число генераторов, отобранных для контроля после i-ой операции;

P_{mi} – вероятность приемки партии генераторов.

В выражении (4) первое слагаемое представляет собой ожидаемую стоимость отбракованных партий, а второе – стоимость приемки партии после i-ой операции, которая принята годной после проверки m_i-го количества генераторов. С другой стороны,

стоимость годных генераторов после i -ой операции равна

$$C_{ri} = C_i N_1 \prod_{i=1}^{i-1} \tilde{P}(H_1, i), \qquad (5)$$

где C_i - стоимость единицы генератора на i -ой операции;

N_1 - количество заготовок, поступающих на i -ую операцию.

Тогда стоимостной ограничивающий показатель ДПК можно определить как относительный ущерб после i -ой технологической операции, т.е.

$$\theta_i = 1 - \frac{C_{ni}}{C_{ri}} \qquad (6)$$

Численно-количественной оценочной величиной показателя ДПК является среднее число годных генераторов, приходящихся на одно забракованное изделие (K_i) на выходе i -ой операции. Причем, поскольку в составе выходных, признанных годными, генераторов существуют также необнаруженные, т.е. фактически некачественные генераторы, то естественно, что действительное K_i меньше соответствующего расчетного значения. Однако, для более точной и реальной численно-количественной оценки значения ДПК следует учитывать также и вероятностный характер ДПК. В связи с этим введем в рассмотрение величину T_i, учитывающую численно-количественный и вероятностный характер показателя ДПК:

$$T_i = \frac{K_i \tilde{P}(H_1, i)}{0{,}5 + K_i \tilde{P}(H_1, i)}$$

При этом в качестве критерия качества для всех технологических операций процесса производства целесообразно использовать средневзвешенный арифметический показатель

$$W = \sum_{i=1}^{n} R_i T_i$$

где R_i - параметр весомости ДПК i -го этапа.

Здесь средневзвешенный показатель W и параметры весомости R_i $(i = 1, 2, \ldots, n)$ подчинены единой цели управления качеством продукции.

Существуют различные методы определения R_i, а именно: метод стоимостных регрессионных зависимостей: метод предельных

и нормальных значений; метод эквивалентных соотношений и экспертный метод. Для производственных процессов такого типа, как производство генераторов, наиболее целесообразными являются метод предельных и нормальных соотношений и экспертный метод. Причем экспертный метод является более универсальным и целесообразным, поскольку его можно применять в любых производственных ситуациях. Метод же предельных и нормальных значений можно применять только там, где достоверно известны предельные допустимые значения показателей качества продукции. В данном случае мы применим метод предельных и нормальных значений. Несколько модифицируя его, параметр весомости R_i можно получить из нижеследующего соотношения:

$$R_i = \frac{\frac{1}{P_{iн} - P_{iпр}}}{\sum \frac{1}{P_{iн} - P_{iпр}}}$$

Поскольку W включает в себя компоненты, определяющие затраты на производство и расходы, обусловленные появлением брака в процессе производства при различных способах контроля, то исходную задачу целесообразно свести к максимизации W при соответствующем ограничении на величину θ, т.е.

$$\theta \leq \theta_{доп}$$

Это означает, что в процессе производства генераторов необходимо одновременно стремиться к минимизации значений $C_п$, $P_{бi}$ и максимизации величин K_i, $\tilde{P}(H_1, i)$. Таким образом, исходную задачу можно сформулировать следующим образом: смоделировать изучаемый технологический процесс производства генераторов с целью анализа уровня ДПК на всех стадиях процесса производства и максимизировать его в тех $i = \{l\}$ технологических операциях, где имеет место неравенство

$$W_i < W_{i\,зад}$$
$$i = \{l\}$$

а по значению достигнутого уровня $max\, W$ оперативно управлять качеством генераторов в процессе их производства.

Рассмотрим математическую модель анализа и повышения уровня ДПК с нижеследующими исходными данными:

N – объем партии;

n – количество брака;

$n_{ло}$ – количество ложного брака;

$n_{но}$ – количество необнаруженного брака;

i^* – общее количество технологических операций;

j^* – общее количество пассивных статистических наблюдений за ходом i -ой операции технологического процесса производства генераторов;

γ^* – общее количество исполнителей, выполняющих i -ую технологическую или контрольную операцию за j -ое наблюдение;

C_i – стоимость заводских расходов на единицу производимого генератора по i -ом операциям процесса производства (i =I,2,..., i^*);

$W_{i\text{зад}}$ – заданный уровень ДПК после i -ой операции процесса производства;

ρ – количество параметров технологического режима на i -ой операции;

t – количество разрядов, по которым разбивается допустимый предел значения исследуемого параметра при статистической обработке данных;

S – значения контролируемого параметра при параметрическом контроле за j -ое наблюдение;

b_1^*, b_2^* – соответственно нижние и верхние допустимые граничные значения измеряемых параметров генераторов;

h^* – количество разрядов, которые используются при статистической оценке закона распределения значения исследуемого параметра.

Пусть нам известны исходные данные процесса производства генераторов за определенный период времени t, т.е. произведена регистрация исходных данных за определенное j^* количество наблюдений.

В этом случае коэффициент k_i оценивается следующим образом:

$$k_i = \frac{P_{ri}}{P_{бi}} \; ; \qquad P_{ri} = 1 - P_{бi} \; ;$$

$$P_{\delta i} = \frac{q_i}{1 - \Sigma q_{i-1}} ; \qquad q_i = \frac{n_i}{N}, \quad i = 1, 2, \ldots, i^*$$

Одновременно на каждой i -ой операции процесса производства оценивается стоимостный признак ДПК согласно(3-6).Далее определяется значение $\tilde{P}(H_1, i)$ и для каждой i -ой операции технологического процесса производства оценивается показатель уровня ДПК:

$$W_i = T_i R_i \qquad i = 1, 2, \ldots, i^*$$

Затем, по определенным заданным уровням ДПК, проверяется условие

$$W_i < W_{i\,зад}, \quad i = 1, 2, \ldots, i^* \qquad (7)$$

Причем, на технологических операциях производства генераторов, где не выполняется условие (7), формируется новый фиксированный уровень для ДПК, а именно:

$$W_{i^*\,зад} = W_i$$

Там же, где выполняется условие (7), просто фиксируются номера этих операций $i = \{l\}$.

Таким образом, совокупность математических соотношений, заданных выражениями (8), составляет математическую модель анализа уровня ДПК на каждой i -ой операции процесса производства генераторов в условиях неопределенности:

$$k_i = \frac{P_{ri}}{P_{\delta i}} ; \qquad \theta = 1 - \frac{C_{ni}}{C_{ri}} ; \qquad \theta_i \le \theta_{i\,зад};$$

$$T_i = \frac{k_i \tilde{P}(H_1, i)}{0{,}5 + k_i \tilde{P}(H_1, i)}, \qquad R_i = \frac{\frac{1}{P_{iн} - P_{iпр}}}{\sum_{i=1}^{n} \frac{1}{P_{iн} - P_{iпр}}}, \qquad (8)$$

$$W_i = T_i R_i ;$$

$$W_i < W_{i\,зад}$$

$$\begin{cases} W_{i\,зад} = W_i & \text{при отрицательном ответе} \\ \text{фиксируются } i = \{e\} & \text{операций при полож ответе} \end{cases}$$

С целью повышения W_i на $i = \{e\}$ технологических операциях, рассмотрим задачу повышения уровня ДПК в условиях неопределенности. Математическую модель максимизации W_i $i = \{l\}$ представим в виде трех подмоделей: повышения W_i , когда на

i -ой операции производится параметрический контроль; повышения W_i, когда на i -ой операции производится визуальный контроль, или эта операция вообще не контролируется; подмодель оптимального размещения исполнителей по своим рабочим местам.

Рассмотрим эти подмодели в отдельности.

I. Подмодель повышения W_i при параметрическом контроле.

Согласно [14], на практике, отклонение измеряемого параметра выпускаемой продукции подчинено нормальному закону с математическим ожиданием m и среднеквадратическим отклонением σ. Исходя из этого определяются экстремальные значения измеряемого параметра S из всей совокупности j -наблюдений. Далее, разбивая весь диапазон измеряемого параметра на h^* равных элементарных частей, определяем граничное значение параметра элементарных участков ΔS_h, т.е.

$$S_h = \min S + (h-1)\Delta S; \quad S_{h+1} = \min S + h\Delta S,$$

где S_h -нижняя граница участка; S_{h+1} -верхняя граница участка.

Из совокупности исходных данных выбирается то количество генераторов m_h, которые имеют $S_{h\,STAT}$ значение измеряемого параметра, т.е.

$$S_h < S_{h\,STAT} \leq S_{h+1} \qquad h = 1, 2, \ldots, h^* \qquad (9)$$

На основе значений j^* и m_h определяется частота P_{h^*} выполнения условия (9) по формуле:

$$P_{h^*} = \frac{m_{h^*}}{j^*}, \qquad h = 1, 2, \ldots, h^*$$

Далее, определяется среднее значение каждого элементарного участка

$$\bar{S}_h = \frac{S_h + S_{h+1}}{2}$$

На основе значений $\bar{S}_h$, P_{h^*} определяется математическое ожидание m^* и дисперсия σ случайной величины S по общеизвестным формулам [11].

$$m^* = \sum_{h=1}^{h^*} \bar{S}_h P_{h^*}; \qquad \sigma = \sqrt{\frac{\Sigma(\bar{S}_h - m^*)^2}{h^*}}$$

Далее, по критерию λ^2 оценивается степень совпадения экспериментального закона распределения случайной величины с теоретическим законом распределения. Процедура такой оценки дана в [11]. Далее, определяем нормированные значения нижеследующих величин:

$$m = \frac{m^*}{\sigma}; \quad b_1 = \frac{b_1^*}{\sigma}; \quad b_2 = \frac{b_2^*}{\sigma}; \quad S_{ном} = \frac{S_{ном}^*}{\sigma}$$

В конечном итоге определяем значение S_0 -смещения центра распределения случайной величины S , максимизирующей W_i .

2. Подмодель повышения W_i , когда на i -ой операции производится визуальный контроль или процесс вообще не контролируется.

Рассмотрим задачу оптимального выбора параметров технологического режима производственного процесса. Представим эту задачу в виде математической подмодели повышения W_i . Целевой функцией в данном случае является функция ДПК - Ω , а решением задачи - максимизация Ω . Одновременно заметим,что в реальных условиях процесса производства решение этой задачи фактически равносильно решению задачи повышения W_i , когда на i -ой операции производится визуальный контроль или процесс вообще не контролируется. Причем, как правило, функция ДПК является параболой второго порядка [18]

$$\Omega = A_1 y^2 + A_2 y + A_3 \qquad (10)$$

где y - значение управляемого параметра;

Ω, A_1, A_2, A_3 - матрицы, состоящие из $\beta = t^{\rho}$ соответствующих элементов;

t - количество пределов изменения параметров технологического режима;

ρ - количество параметров технологического режима.

Тогда математическую подмодель **повышения уровня** W_i посредством оптимального выбора параметров производственного процесса можно представить в следующем виде:

$$\begin{aligned} &\Omega = A_1 y^2 + A_2 y + A_3 \\ &\max y = -\frac{A_2}{2A_1} \\ &\max \Omega = -\frac{A_2^2}{4A_1} + A_3 \end{aligned} \qquad (11)$$

Далее проверяется условие:

$$\max W_i' < W_{i\,зад}$$

$$W_i' = \begin{cases} W_i \text{ при } \omega = 1, \; i = \{l\} \; l \in i \\ W_{i\,зад} \text{ при } \omega = 0 \quad i = 1, 2, \ldots, i^* \end{cases}$$

где W_i' -уровень ДПК после соответствующей оптимизации.

3. Подмодель оптимального распределения исполнителей по технологическим операциям.

На основе исходных данных, на каждой i -ой операции технологического процесса производства генераторов определяется среднее число нижеследующих параметров

$$N_{\gamma i} = \frac{\sum_{j=1}^{j^*} N_{\gamma i j}}{j^*};$$

$$n_{\gamma i} = \frac{\sum_{j=1}^{j^*} n_{\gamma i j}}{j^*};$$

$$n_{(\text{ло})\gamma i} = \frac{\sum_{j=1}^{j^*} n_{(\text{ло})\gamma i j}}{j^*};$$

$$n_{(\text{но})\gamma i} = \frac{\sum_{j=1}^{j^*} n_{(\text{но})\gamma i j}}{j^*}$$

Далее, определяется вероятность $\tilde{P}_{\gamma i}(A)$ по формуле (I) с помощью функции принадлежности μ_A и вероятности $P_{\gamma i}(A)$, получаемой из следующих выражений:

$$P_{\gamma i}(A) = P_{\gamma}(B,i) = P_{(\text{ло})\gamma i}(B) + \frac{N_{\gamma i} - n_{\gamma i} - n_{(\text{но})\gamma i}}{N_{\gamma i}};$$

$$P_{(\text{ло})\gamma i}(B_2) = \frac{n_{(\text{ло})\gamma i}}{N_{\gamma i}}$$

При этом из множества значений вероятности $\tilde{P}_{\gamma i}(A)$ выбирается максимальное, а номер γ указывает соответствующего исполнителя, который должен обслуживать i -ую операцию с целью получения максимального значения W_i. Аналогичным образом, определяя вероятности $\tilde{P}_{\gamma i}(K)$, устанавливается целесообразность контроля i -ой операции и выбирается соответствующий контролер с максимальным значением W_i.

Предлагаемые имитационные подмодели технологического процесса производства генераторов позволяют оперативно контролировать и управлять качеством генераторов в ходе их производства. При соответствующей корректировке они универсальны в прикладном значении и для других отраслей промышленности.

Литература

I. Эффективность систем управления качеством. М.,Изд-во стандартов,1970г.

2. Организация процесса управления. Под редакцией Г.Х.Попова, М., Экономика, 1975г.

3. Абрамов В.А. Математические методы в организации производства и управлении. МИЭТ, 1974г.

4. Шор Я.Б. Основные понятия и термины системы управления качеством продукции. - В.Ж.: Стандарты и качество, 1970г. №2.

5. Гличев А.В. Экономическая наука и качество продукции.- В.Ж.: Стандарты и качество, 1970г., № 9.

6. Трапезников В.А. Вопросы управления экономическими системами. - В.Ж.: Наука и жизнь, 1969г., № 1.

7. Х.Урио. Задачи службы контроля качества. Сборник материалов по зарубежным источникам. М., Наука, 1972.

8. А.Баар. Эффективность систем управления качеством. Сборник материалов по зарубежным источникам. М.,Наука,1972.

9. Л.А.Конарева. Управление качеством продукции в промышленности США. М., Наука, 1977.

10. К.Асаи, Х.Танака. Приложение теории нечетких множеств к задаче принятия решений и задаче управления. Перевод из журнала "Сисутэму то сэй гё", 1975г.

11. Е.С.Вентцель. Теория вероятностей. М., Наука, 1964.

12. Л.А.Заде. Fuzzy sets. Inf. and control, No.8, 1965.

13. Л.В.Барташев. Технико-экономические расчеты при проектировании и производстве машин. М., Машиностроение, 1968г.

14. П.И.Бобрик. Влияние надежности технологического процесса на надежность изготовляемого изделия - В.Ж.: Надежность и контроль качества, № 8, 1973г.

15. А.Кофман. Введение в теорию нечетких множеств. М., Радио и связь, 1982г.

16. А.Н.Аверкин и др. Нечеткие множества в моделях управления и искусственного интеллекта. М., Наука,1986.

17. L.A,Zadeh. Probability measures of Fuzzy events-Journal of Math.analyses and appl., No.23, 1968, p.421-427.

18. Карапетян А.М., Саакян А.А. Математическая модель процесса производства печатных плат. - Межвузовский сборник научных трудов, серия ХУП, Ереван,1976г.

Theoretico-modellistic approach to estimation and quality--control of production in the case of uncertainity of parameters of the system

A.E. Melkonjan V.A. Barsegjan, A.A. Ajvazjan

Summary

In the paper the problem of increasing the quality of the production in machine industry is studied. For formal description of the production process the theory of statistical control and that of fuzzy sets are used. A chierarchical system of mathematical models is proposed to handle input information. For the automatization and operative control of processes PROLOG is used as the software tool. The basic element in the program is the technological operation and for the control the magnitude of a choosen dynamical criterium of the quality is used.

MTA SZTAKI Tanulmányok 202/1987 pp. 131-148

ДИНАМИЧЕСКАЯ МОДЕЛЬ ОПТИМАЛЬНОГО ВОДОРАСПРЕДЕЛЕНИЯ В ОРОШАЕМОМ ЗЕМЛЕДЕЛИИ С УЧЕТОМ РЕСУРСНЫХ ОГРАНИЧЕНИЙ

Мелконян А.Е., Манучарян С.М.

В экономико-математической модели [3] не рассматривалось влияние на урожайность сельскохозяйственных культур режимов орошения, т.е. совокупности сроков и норм полива, обеспечивающих при данных климатических и агротехнических условиях необходимый для данной культуры водный режим почвы. В вышеуказанной модели урожайность культур определялась только в зависимости от оросительных норм m_i, $i \in J_n$, т.е. от общего количества оросительной воды, необходимой для 1га определенной культуры за весь вегетационный период $[t_{ni}, t_{ci}]$, где: t_{ni} - дата посева, t_{ci} - дата сбора урожая i-ой культуры.

Таким образом, для нижеследующих задач мы вводим понятие поливных режимов сельскохозяйственных культур $m_i(t)$, $i \in J_n$, определяющих нормы полива t $t \in [t_{ni}, t_{ci}]$ и соответственно вводится понятие оптимальных режимов орошения $\omega_i(t)$ $i \in J_n$, обеспечивающих максимально возможные урожаи сельскохозяйственных культур при оптимальном уровне агротехнических работ и усредненных климатических условиях, характерных для рассматриваемого хозяйства. Последние являются известными функциями и, обычно, задаются в виде план-графиков вегетационных поливов.

Введем понятие некоторой производственной функции $E_i(t, m_i(t))$, характеризующей условное приращение урожайности i-ой культуры в момент времени $t \in [t_{ni}, t_{ci}]$ при ее поливе водой в количестве $m_i(t)$. Будем считать, что для i-ой культуры определена такая непрерывная и дифференцируемая относительно своих аргументов производственная функция, что имеет место

$$y_i(t) = \int_{t_{ni}}^{t_{ci}} E_i(t, m_i(t))dt, \qquad \forall\, t \in [t_{ni}, t_{ci}] \quad (1)$$

Введем следующие обозначения:

$$t_c = \max\{t_{ci}\}; \qquad t_n = \min\{t_{ni}\};\ i \in J_n \quad (2)$$

где t_n и t_c – соответственно общие даты посева и сбора урожая орошаемых культур для хозяйства в целом.

С учетом введенных понятий урожайность i -ой культуры будет задаваться следующим соотношением

$$y_i(t) = \int_{t_n}^{t^*_c} \bar{E}_i(t, m_i(t))\,dt, \qquad \forall\, t \in [t_n, t_c] \quad (3)$$

где

$$\bar{E}_i(t, m_i(t)) = \begin{cases} E_i(t, m_i(t)) \text{ при } & t_{mi} \le t \le t_{ci} \\ 0 \text{ при } & t < t_{ni},\ t^* > t_{ci} \end{cases} \quad (4)$$

Тогда выражение (3) для собранного урожая примет вид:

$$y_i(t_c) = \int_{t_n}^{t_c} \bar{E}_i(t, m_i(t))\,dt, \qquad i \in J_n \quad (5)$$

Заметим, что (4) в соотношениях (3) и (5) учитывает тот факт, что даты посева и сбора урожая различных культур хозяйства неодинаковы.

Таким же образом, как и в [3] зависимости урожая сельскохозяйственных культур от оросительных норм рассматриваем в виде параболы второго порядка

$$y_i(t_c) = l_i^2 z_i^2 + l_i^1 z_i + l_i^0, \qquad i \in J_n \quad (6)$$

где

$$l_i^2 = -d_i, \quad l_i^1 = 2d_i\tilde{z}_i, \quad l_i^0 = y_i^c \quad (7)$$

являются известными величинами. Здесь z_i и $\tilde{z}_i$ полностью соответствуют m_i и $\tilde{m}_i$ в [3], что и отображено на рис.I.

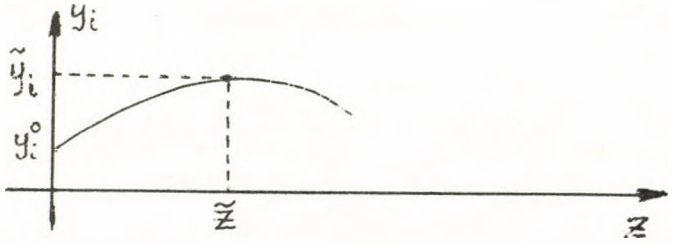


Рис.I

С учетом вышеизложенного очевидно, что:

$$z_i = \int_{t_n}^{t_c} m_i(t)dt \,, \qquad i \in J_n \tag{8}$$

$$\tilde{z} = \int_{t_n}^{t_c} \omega_i(t)dt \,, \qquad i \in J_n \tag{9}$$

Помимо производственной функции введем в рассмотрение еще одну вспомогательную функцию $\mu_i(t), i \in J_n$, представляющей из себя количество воды, израсходованной на полив I га посевной площади культуры от начала вегетационного периода до момента времени $t \in [t_n, t_c]$, или иначе, она фактически представляет собой условное значение оросительной нормы культуры от начала общего поливного сезона хозяйства до момента $t \in [t_n, t_c]$, т.е.

$$\mu_i(t) = \int_{t_n}^{t_c} m_i(t)dt \,, \qquad i \in J_n \tag{10}$$

Из сравнения (8) с (10) видно, что при $t = t_c$ имеет место

$$\mu_i(t_c) = z_i \,, \qquad i \in J_n \tag{11}$$

Для орошения всех культур хозяйства отводится оросительный фонд воды

$$M(t) = \sum_{i=1}^{n} R_i m_i(t) \tag{12}$$

Аналогичным образом в каждый момент времени величина оптимального оросительного фонда воды хозяйства, обеспечивающего всем культурам оптимальные режимы орошения, определится из нижеследующего соотношения

$$\Omega(t) = \sum_{i=1}^{n} R_i \omega_i(t) \tag{13}$$

С учетом (12) годовой оросительный фонд воды хозяйства определится как

$$M = \sum_{i=1}^{n} R_i z_i = \int_{t_n}^{t_c} M(t)dt \tag{14}$$

Аналогичным образом, оптимальный годовой фонд воды хозяйства, обеспечивающий получение максимальных урожаев по всем культурам хозяйства, с учетом соотношений (13) и (9) определится

как

$$\Omega = \sum_{i=1}^{n} R_i \tilde{z}_i = \int_{t_n}^{t_c} \Omega(t)dt \qquad (I5)$$

Хозяйство реализует собранный урожай по всем орошаемым культурам в соответствии с закупочными (колхозы) или сдаточными (совхозы) ценами c_i , $i \in J_n$ и получает валовую продукцию орошаемого растениеводства в стоимостном выражении в размере

$$S = \sum_{i=1}^{n} c_i R_i y_i(t_c) \qquad (I6)$$

Рассмотрим задачу определения рациональных режимов орошения сельскохозяйственных культур в условиях ограниченного годового оросительного фонда воды хозяйства, т.е. при

$$M < \Omega \qquad (I7)$$

Задача хозяйства заключается в максимизации вырожденного функционала (I6), т.е. валовой продукции растениеводства хозяйства в целом

$$S \to max$$

при ограничениях и связях (I8), (3-I6) на управляющие параметры при условии их неотрицательности

$$m_i(t) \geq 0 \quad , \qquad i \in J_n \qquad (I8)$$

Требование неотрицательности управляющих параметров вытекает из их физического смысла.

В анализируемой модели предполагается, что в наиболее напряженный период водопользования пропускная способность внутрихозяйственной оросительной сети допускает всевозможные перераспределения оросительной воды между культурами хозяйства с учетом рациональных режимов орошения.

За критерий оптимальности в рассматриваемой модели принят максимум валовой продукции орошаемого растениеводства хозяйства, так как нашей целью является лишь определение наиболее эффективных (рациональных) поливных режимов, что не требует никаких дополнительных затрат, а только лишь позволяет облегчить и оптимизировать процесс принятия решений по эффективному водораспределению в хозяйстве.

Для решения подобных задач требуются определенные исходные данные в виде тех или иных зависимостей урожая различных сельскохозяйственных культур от режимов орошения. Такой функцией в рассматриваемой модели является производственная функция, для определения которой принимаем ее в виде параболы второго порядка с учетом логической и физической связи с кривыми урожайности, т.е. имеем:

$$E_i(t, m_i(t)) = -a_i(t)m_i^2(t) + b_i(t)m_i(t) + p_i(t) \qquad (19)$$
$$a_i(t) > 0 \ , \ \forall\, t \in [t_n, t_c] , \quad i \in J_n$$

Неизвестные коэффициенты производственной функции определим в ходе решения исходной оптимизационной задачи.

Сформулированная экономико-математическая модель оптимального водораспределения с учетом реальных режимов орошения является сложной вариационной задачей оптимизации вырожденного функционала (I6). Проанализируем основное ограничение исходной задачи (I7) на располагаемый годовой лимит воды хозяйства. С учетом (I2 - I5) указанное основное ограничение разбивается на три частных ресурсных ограничения:

$$z_i \le \check{z}_i \ , \qquad i \in J_n \qquad (20)$$

$$M(t) \le \Omega(t) \ , \qquad \forall\, t \in [t_n, t_c] \qquad (21)$$

$$m_i(t) \le \omega_i(t) \ , \qquad \forall\, t \in [t_n, t_c] \qquad (22)$$

Причем, последнее ограничение (22) логически дополняет ограничение (I8) в виде общего ресурсного ограничения на управляющие параметры:

$$0 \le m_i(t) \le \omega_i(t) \ , \quad i \in J_n , \quad \forall\, t \in [t_n, t_c] \qquad (23)$$

Согласно ограничениям (20) и (2I) исходную оптимизационную задачу разобъем на две частные задачи, учитывающие указанные ограничения в отдельности. С учетом ограничения (20) исходная задача сводится к максимизации величин урожая каждой орошаемой культуры хозяйства в отдельности (локальная задача), т.е. к оптимальному распределению заданных оросительных норм z_i, $i \in J_n$, в виде рациональных режимов орошения в течение вегетационного периода культуры. В этом случае

оптимизируемый функционал представит собой максимум собранного урожая по каждой культуре хозяйства в отдельности.

Однако, такая постановка исходной задачи не учитывает, что в каждый момент $t \in [t_n, t_c]$ величина $M(t)$ ограничена сверху согласно ограничению (21). С учетом указанного ограничения исходная задача приобретает полноту охвата основных ограничений задачи и одновременно учитывает также общую ограниченность располагаемого фонда воды хозяйства в каждый момент времени общего вегетационного периода.

Сформулированные оптимизационные задачи решим в указанном порядке, а затем - и вторую задачу с учетом плановых заданий по выходу сельхозпродукции.

Рассмотрим исходную оптимизационную задачу с учетом ограничения (20). С целью упрощения дальнейших выкладок будем рассматривать только одну, условно взятую культуру, подразумевая те же действия и с остальными орошаемыми культурами хозяйства. Таким образом, поставленную локальную задачу можно представить как простейшую задачу оптимального управления (задача Майера [4]) следующим образом: максимизировать вырожденный функционал

$$F = y(t_c) = \int_{t_n}^{t_c} \bar{E}(t, m(t))\,dt \to max \qquad (24)$$

т.е. найти элемент максимума, $\bar{m}(t), t \in [t_n, t_c]$ при следующих ограничениях и связях на управляющий параметр $m(t)$ и фазовые координаты $y(t)$ и $\mu(t)$, $t \in [t_n, t_c]$:

$$\begin{aligned} 0 \le m(t) &\le \omega(t) \\ \frac{dy}{dt} &= E(t, m(t)) \\ \frac{d\mu}{dt} &= m(t) \end{aligned} \qquad (25)$$

с граничными значениями фазовых координат

$$y(t_n) = 0 \;,\; \mu(t_n) = 0 \;,\; \mu(t_c) = z \qquad (26)$$

где фазовые координаты: $y(t)$ - непрерывная и дифференцируемая функция $\forall t \in [t_n, t_c]$, $\mu(t)$ - кусочно-дифференцируемая функция $\forall t \in [t_n, t_c]$, а управление $m(t)$ - кусочно-непрерывная функция $\forall t \in [t_n, t_c]$.

Используя принцип максимума Л.С.Понтрягина [1] (необходимое условие оптимальности), составим гамильтониан $H(\psi, t, m)$, для которого имеет место минимум по управлению $m(t)$:

$$H(\psi, t, m) = \psi_y(t) E(t, m(t)) + \psi_\mu(t)\, m(t)$$

т.е. $$H(\psi, t, \bar{m}) = \min_{m(t)} H(\psi, t, m) \qquad (27)$$

где $\psi_y(t)$ и $\psi_\mu(t)$ - функции, удовлетворяющие сопряженной системе уравнений:

$$\dot{\psi}_y(t) = -\frac{\partial H}{\partial y} = 0 \,, \quad \dot{\psi}_\mu(t) = -\frac{\partial H}{\partial \mu} = 0$$

откуда следует, что $\psi_y = const$, $\psi_\mu = const$ так как минимизируемый гамильтониан от y и μ не зависит.

Из необходимого условия минимума гамильтониана (27) вытекает уравнение, связывающее оптимальное значение с неизвестными постоянными ψ_y и ψ_μ : $H_m = \frac{\partial H}{\partial m} = \psi_y \frac{\partial E(t, m(t))}{\partial m} + \psi_\mu = 0$

или с учетом (21) $$H_m = \psi_y(-2a(t)\bar{m}(t) + b(t)) + \psi_\mu = 0 \qquad (28)$$

Исходная задача является задачей оптимального управления с незакрепленным правым концом для фазовой координаты $y(t)$, так как нам неизвестно ее конечное значение при $t = t_c$. В таком частном виде принцип максимума уже не дает замкнутой системы уравнений для определения минимума гамильтониана. Поэтому, используя условие трансверсальности в форме, предложенной В.Ф.Кротовым [2], определим недостающее граничное условие. Для ее нахождения запишем функционал, зависящий от начального и конечного значения фазовой координаты $y(t)$:

$$\Phi(y(t_n)\, y(t_c)) = F(y(t_n)\, y(t_c)) + \psi_y(t_c)\, y(t_c) - \psi_y(t_n) y(t_n) \,,$$

где $F = y(t_c)$ - исходный максимизируемый функционал задачи и решим задачу на условный минимум функционала Φ по переменной $y(t_c)$ при условии $y(t_n) = 0$. Из необходимого условия этого минимума мы получим искомое условие трансверсальности, а именно: при $y(t_n) = 0$ следует, что $\Phi = y(t_c)(1 + \psi_y)$, откуда минимум функционала достигается при $\psi_y = -1$. С учетом этого соотношение (28) примет вид:

$$H_m = 2a(t)\bar{m}(t) - b(t) + \psi_\mu = 0$$

Проверка достаточного условия минимума гамильтониана

$$H_{mm} = 2a(t) > 0 \qquad (29)$$

позволяет убедиться в том, что $\bar{m}(t) = \frac{b(t)}{2a(t)} - \frac{\psi_\mu}{2a(t)}$; $\forall t \in [t_n, t_c]$ является элементом минимума гамильтониана (27).

Определим оптимальный поливной режим культуры $\omega(t)$ при отсутствии ограничения (20) на располагаемый водный ресурс (оросительная норма) культуры. Тогда задача максимизации вырожденного функционала (24) без учета ограничения (20), т.е. при $m(t) \geq 0$ по теореме о достаточном условии максимума дает следующую схему решения указанной задачи оптимального управления: $\omega(t)$ - является элементом максимума вырожденного функционала (24), если ее значения $\forall t \in [t_n, t_c]$ доставляют максимум подинтегральной производственной функции рассматриваемой культуры, т.е.

$$E(t, \omega(t)) = \max_{m(t)} E(t, m(t))$$

При этом точка относительного максимума производственной функции одна при $m(t) \geq 0$ и определяется из уравнения

$$\frac{\partial E}{\partial m} = -2a(t)\omega(t) + b(t) = 0$$

$$\omega(t) = \frac{b(t)}{2a(t)} \qquad \forall t \in [t_n, t_c] \qquad (30)$$

Для упрощения дальнейших преобразований принимаем

$$a(t) = a = const \qquad p(t) = p = Const$$

С учетом краевого условия $\mu(t_c) = z$ и согласно (8-9) определим значение неизвестной постоянной ψ_μ с использованием соотношения (29). Получим

$$\psi_\mu = \frac{2a(\tilde{z} - z)}{\tau} \qquad (31)$$

где $\tau = t_c - t_n$ - общий вегетационный период орошаемых культур рассматриваемого хозяйства.

С учетом (31) и общего ограничения исходной задачи на управляющий параметр получим значения рационального режима орошения рассматриваемой культуры $\forall t \in [t_n, t_c]$:

$$\bar{m}(t)=\begin{cases}\omega(t)-\frac{\tilde{z}-z}{\tau}, & \text{при } \omega(t)>\frac{\tilde{z}-z}{\tau}\\ 0, & \text{при } \omega(t)\le\frac{\tilde{z}-z}{\tau}\end{cases} \qquad (32)$$

Обозначим $\gamma=\frac{\tilde{z}-z}{\tau}$ (33)

где γ - постоянная известная величина для каждой культуры хозяйства.

С учетом (33) соотношение (32) окончательно примет нижеследующий вид:

$$m(t)=\begin{cases}\omega(t)-\gamma, & \text{при } \omega(t)>\gamma\\ 0, & \text{при } \omega(t)\le\gamma\end{cases} \qquad (34)$$

Графическая интерпретация соотношения (34) показывает, что рациональные поливные графики орошаемых культур строятся следующим образом: из оптимального режима орошения культуры $\omega(t)$ (см.рис.2) вычитается постоянное число γ для данной культуры, и в полученном рациональном поливном режиме $\bar{m}(t)$ участки с отрицательными значениями ординат заменяются нулевыми в силу ограничения (23) (см.рис.3).

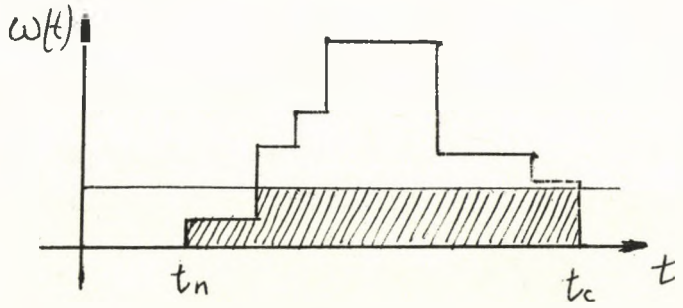


Рис.2

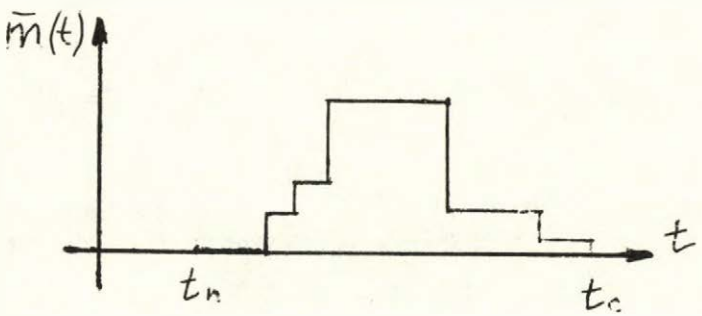


Рис.3

Заметим, что удалось получить оптимальное решение (34) исходной задачи в очень удобной форме, зависящей от коэффициентов $a(t)$ и $b(t)$, введенной нами производственной функции не непосредственно, а через функцию $\omega(t)$, имеющую очевидный физический смысл - оптимальный режим орошения культуры.

Оптимальное значение фазовой координаты $\bar{u}(t)$ с учетом (34) и соответствующего уравнения связи (I0) определится как:

$\forall t \in [t_n, t_c]$

$$\bar{u}(t) = \begin{cases} \int_{t_n}^{t_c} \omega(t)dt - \gamma(t - t_n), \text{при} \quad \omega(t) > \gamma \\ 0, \text{при} \quad \omega(t) \le \gamma \end{cases} \tag{35}$$

Оптимальное значение фазовой координаты $\bar{y}(t)$ определим после нахождения неизвестных коэффициентов производственной функции культуры. Подставляя (34) в (I9) с учетом (30), получим:

$$E(t, m(t)) = \begin{cases} a\omega^2(t) - \gamma^2 a + p, \text{при} \quad \omega(t) > \gamma \\ p, \text{при} \quad \omega(t) \le \gamma \end{cases} \tag{36}$$

Определим коэффициенты производственной функции из условия равенства фазовой координаты $y(t)$ при $t = t_c$ значению урожайности рассматриваемой культуры согласно выражения (6), т.е. с учетом рационального режима орошения.

Подставляя (36) в соотношение (5), получим значение неизвестного граничного условия для фазовой координаты $y(t)$:

$$\bar{y}(t_c) = \begin{cases} \int_{t_n}^{t_c} \omega^2(t)dt - \gamma^2 a\tau + p\tau, \text{при} \quad \omega(t) > \gamma \\ p\tau, \text{при} \quad \omega(t) \le \gamma \end{cases} \tag{37}$$

С учетом (44) имеем:

$$y(t_c) = \begin{cases} -\frac{a}{\tau}z^2 + \frac{2\tilde{z}a}{\tau}z + \left(p\tau - \frac{\tilde{z}^2 a}{\tau} + a\int_{t_n}^{t_c} \omega^2(t)dt\right), \text{при} \quad \omega(t) > \gamma \\ p\tau, \text{при} \quad \omega(t) \le \gamma \end{cases} \tag{38}$$

С другой стороны, согласно (6-7) $\bar{y}(t_c) = -dz^2 + 2d\tilde{z}z + y^0$

Приравнивая соответствующие коэффициенты при одинаковых степенях z в соотношениях (38-39), получим:

при $\omega(t) > \gamma$, т.е. при ненулевых значениях рациональ-

ного поливного режима соответственно получим

$$b(t) = 2\tau d\omega(t); \qquad p = \frac{y^0}{\tau} + \frac{\tilde{z}^2 d}{\tau} - d\int_{t_n}^{t_c} \omega^2(t)dt; \tag{40}$$

при $\omega(t) \leq \gamma$, т.е. при нулевых значениях рационального поливного режима, согласно (8) следует, что $z = 0$, т.е. $p\tau = y^0$; откуда

$$p = \frac{y^0}{\tau} \tag{41}$$

Подставляя полученные значения коэффициентов производственной функции в (37), окончательно определим недостающее граничное условие исходной оптимизационной задачи:

$$\bar{y}(t_c) = \begin{cases} y^0 + \tilde{z}^2 d - \gamma^2 \tau^2 d, & \text{при } \omega(t) > \gamma \\ y^0, & \text{при } \omega(t) \leq \gamma \end{cases} \tag{42}$$

Согласно (3) с учетом полученных соотношений оптимальное значение фазовой координаты $y(t)$ определяется как:

$$y(t) = \begin{cases} \tau d\int_{t_n}^{t_c} \omega^2(t)dt + (t - t_n)\left(\frac{y^0}{\tau} + \frac{\tilde{z}^2 d}{\tau} - \gamma^2 d\tau - d\int_{t_n}^{t_c} \omega^2(t)dt\right), & \text{при } \omega(t) > \gamma \\ \frac{y^0}{\tau}(t - t_n), & \text{при } \omega(t) \leq \gamma; \quad \forall t \in [t_n, t_c]. \end{cases} \tag{43}$$

Проверив для оптимальных фазовых координат достаточное условие оптимальности, убеждаемся, что согласно (35) и (43) выполняются граничные условия задачи, и что полученное решение (34,35 и 43) - действительно есть элемент максимума исходного оптимизационного функционала.

Точность решения рассматриваемой задачи можно повысить, если отказаться от принятых допущений в виде $a(t) = a = const$ и $p(t) = p = const$. В зависимости от удобства (от вида и способа задания оптимального поливного режима), исходная задача может решаться аналитическим или графическим методами.

Перейдем теперь к рассмотрению этой же задачи с учетом более общего ограничения (21). В отличие от предыдущей оптимизационной модели, здесь уже необходимо совместное рассмотрение всех орошаемых культур хозяйства в целом с целью мак-

симизации валовой продукции растениеводства в стоимостном выражении. Исходное ограничение данной задачи (2I) означает, что $\forall t \in [t_n, t_c]$ полностью расходуется на орошение наличный лимит воды, если он не превосходит соответствующего оптимального фонда воды или используется оптимальный оросительный фонд воды $\Omega(t)$, если это позволяет располагаемый ресурс воды. Таким образом, в рассматриваемой задаче необходимость во введении вспомогательной функции $\mu(t)$, контролирующей состояние водопотребления культуры $\forall t \in [t_n, t_c]$, отпадает, поскольку в любой момент времени общего вегетационного периода τ весь наличный ресурс оросительной воды полностью расходуется на орошение из-за его ограниченности.

В данной задаче хозяйство представляет собой динамическую неавтономную систему, управляющие и фазовые параметры которой являются явными функциями времени. Вначале решим эту задачу без учета плановых заданий по выходу продукции растениеводства, а затем уже с учетом указанных заданий.

Сформулируем задачу в математических терминах. Исходная задача формулируется как задача оптимального управления [I] следующим образом: найти элемент максимума $\bar{m}(t) = (\bar{m}_1(t), \dots, m_n(t))$ вырожденного функционала

$$F = \sum_{i=1}^{n} c_i R_i y_i(t_c) = \int_{t_n}^{t_c} \sum c_i R_i \bar{E}_i(t, m_i(t))\,dt, \quad \forall t \in [t_n, t_c] \qquad (44)$$

удовлетворяющего уравнениям связи

$$\frac{dy_i}{dt} = \bar{E}_i(t, m_i(t)), \quad i \in J_n \qquad (45)$$

ограничениям

$$0 \le m_i(t) \le \omega_i(t), \quad i \in J_n \qquad (46)$$

$$M(t) = \sum_{i=1}^{n} R_i m_i(t) \qquad (47)$$

и краевым условиям

$$y_i(t_n) = 0, \quad i \in J_n \qquad (48)$$

Сформулированная задача оптимального управления с закрепленным временем $t_n \le t \le t_c$, закрепленным левым концом и свободным правым концом решается по принципу максимума Л.С.Понтрягина [I] : для того, чтобы процесс $(\bar{m}_1(t), \dots, \bar{m}_n(t))$ давал решение поставленной задачи динамической оптимизации вырож-

ценного функционала (44) при условиях (45–48) необходимо, чтобы $\forall t \in [t_n, t_c]$ было выполнено условие минимума гамильтониана

$$H(\psi, t, \bar{m}) = \min_{m(t)} H(\psi, t, m)$$

где
$$H(\psi, t, m) = \sum_{i=1}^{n} \psi_i(t) \bar{E}_i(t, m_i(t)) \qquad (49)$$

а вспомогательные переменные $(\psi_1(t), \ldots \psi_n(t))$ – решение сопряженной системы уравнений

$$\dot{\psi}_i = -\frac{\partial H}{\partial y_i} = 0, \quad i \in J_n \qquad (50)$$

с начальными условиями $\psi_i(t_n) = -c_i R_i$, $i \in J_n$. Тогда из (50) следует, что $\psi_i(t) = const, i \in J_n$ т.е.

$$\psi_i = -c_i R_i, \quad i \in J_n \qquad (51)$$

Последний результат есть не что иное, как условие трансверсальности для правого конца фазовой координаты.

Учет ограничения в виде равенства (47) проведем с помощью метода Лагранжа [4], для чего введем в рассмотрение функцию Лагранжа

$$H = \sum_{i=1}^{n} - c_i R_i \bar{E}_i(t, m_i(t)) + \lambda \left(-M(t) + \sum_{i=1}^{n} R_i m_i(t)\right) \qquad (52)$$

для которой по необходимому условию оптимальности (минимум гамильтониана (49) имеют место условия стационарности функции Лагранжа:

$$\frac{\partial H}{\partial m} = -c_i R_i \frac{\partial \bar{E}_i}{\partial m_i} + R_i \lambda = 0; \quad \frac{\partial H}{\partial \lambda} = 0; \quad i \in J_n \qquad (53)$$

где $\lambda = \lambda(t)$ – неопределенный множитель Лагранжа, играющий здесь роль дополнительного параметра управления.

С учетом (19) и (30) стационарная точка Лагранжиана (52) определяется из соотношения (53):

$$\hat{m}_i(t) = \omega_i(t) - \frac{\lambda(t)}{2a_i c_i}, \quad i \in J_n \quad \forall t \in [t_n, t_c] \qquad (54)$$

Проверка достаточного условия минимума Лагранжиана $\frac{\partial H^2}{\partial m^2} = 2c_i R_i a_i > 0$ позволяет убедиться в том, что стационарная точка (54) отвечает минимуму Гамильтониана (49).

С учетом ограничения (46) искомые рациональные режимы

орошения культур хозяйства определятся из нижеследующего соотношения:

$$\bar{m}_i(t)=\begin{cases}\omega_i(t)-\dfrac{\lambda(t)}{2a_ic_i} & \text{при} \quad \omega_i(t)>\dfrac{\lambda(t)}{2a_ic_i} \\ 0 & \text{при} \quad \omega_i(t)\le\dfrac{\lambda(t)}{2a_ic_i}\end{cases} ; \; i\in J_n, \; \forall t\in[t_n,t_c] \tag{55}$$

Подставляя (55) при $\omega_i(t)>\frac{\lambda(t)}{2a_ic_i}$ в ограничение (47), определим неизвестный множитель Лагранжа:

$$\lambda(t)=\frac{\Omega(t)-M(t)}{\beta} \tag{56}$$

где

$$\beta=\frac{1}{2}\sum_{i=1}^{n}\frac{R_i}{a_ic_i} \tag{57}$$

постоянная величина для рассматриваемого хозяйства.

Тогда с учетом (40) (в данном случае $a_i=c_id_i$, $i\in J_n$), окончательно получим:

$$\bar{m}_i(t)=\begin{cases}\omega_i(t)-\dfrac{\Omega(t)-M(t)}{2c_id_i\tau\beta} & \text{при} \quad \omega_i(t)>\dfrac{\lambda(t)}{2a_ic_i} \\ 0, \text{ при} & \omega_i(t)\le\dfrac{\lambda(t)}{2a_ic_i}\end{cases} ; \; i\in J_n, \; \forall t\in[t_n,t_c] \tag{58}$$

Производственные фучкции орошаемых культур хозяйства с учетом найденных значений режимов орошения определятся следующим образом:

$$\bar{E}_i(t,m_i(t))=\begin{cases}a_i\omega_i^2(t)-\dfrac{\lambda^2(t)}{4a_ic_i^2}+p_i, \text{при} & \omega_i(t)>\dfrac{\lambda(t)}{2a_ic_i} \\ p_i, \text{ при} & \omega_i(t)\le\dfrac{\lambda(t)}{2a_ic_i}\end{cases} ; \; i\in J_n, \; \forall t\in[t_n,t_c] \tag{59}$$

Согласно (3) с учетом (59) определим рациональные значения фазовых координат оптимизируемой системы:

$$\bar{y}_i(t)=\begin{cases}a_i\int_{t_n}^{t_c}\omega_i^2(t)dt-\dfrac{1}{4a_ic_i^2}\int_{t_n}^{t_c}\lambda^2(t)dt+p_i(t-t_n), \text{при} & \omega_i(t)>\dfrac{\lambda(t)}{2a_ic_i} \\ p_i(t-t_n), \text{ при} & \omega_i(t)\le\dfrac{\lambda(t)}{2a_ic_i}\end{cases} ; \; i\in J_n, \; \forall t\in[t_n,t_c] \tag{60}$$

Тогда, согласно (8 - 9) с учетом (60), рациональные оросительные нормы культур хозяйства определяются как:

$$\bar{z}_i = \begin{cases} \tilde{z}_i - \dfrac{\Omega - M}{2\beta a_i c_i} \text{ при } \omega_i(t) > \dfrac{\lambda(t)}{2a_i c_i} \\ 0 \text{ при } \omega_i(t) \le \dfrac{\lambda(t)}{2a_i c_i} \end{cases} ; \quad i \in J_n, \quad \forall t \in [t_n, t_c] \qquad (61)$$

Полученное соотношение полностью совпадает с аналогичным результатом в [3] , что подтверждает правомерность согласованного определения коэффициентов производственной функции и целесообразность ее рассмотрения.

Проверка для соотношений (58) и (60) достаточных условий оптимальности исходного максимизируемого функционала согласно (45-48) позволяет нам убедиться в том, что полученное решение (58) - действительно является элементом максимума оптимизируемого функционала.

Рассмотрим эту же задачу с учетом плановых заданий по выходу сельхозпродукции с орошаемых массивов хозяйства, т.е. с учетом ограничения

$$y_i(t_c) = \int_{t_n}^{t_c} \bar{E}_i(t, m_i(t))\,dt \ge \frac{K_i}{R_i} , \quad i \in J_n \qquad (62)$$

Для упрощения рассматриваемой задачи разобъем изопериметрическое ограничение (62) на два нижеследующих:

$$y_i(t_c) = \frac{K_i}{R_i} , \quad i \in J_n \qquad (63)$$

$$\int_{t_n}^{t_c} E_i(t, m_i(t))\,dt > \frac{K_i}{R_i} , \quad i \in J_p , \quad J_p = J_n / J_H \qquad (64)$$

Фактически мы разделили весь набор орошаемых культур хозяйства на "нерентабельные" культуры с соответствующими ограничениями (6) в виде граничных условий для фазовых координат в правом конце и "рентабельные" культуры - с соответствующими интегральными ограничениями (64). Перечень вышеуказанных культур легко можно определить следующим образом: I. Решается аналогичная задача без учета ограничения (62) предыдущим методом. 2. Из сравнения объемов произведенной сельхозпродукции по орошаемым культурам с соответствующими плановыми заданиями легко определяются искомые перечни культур.

Таким образом, рассматриваемая оптимизационная задача значительно упростилась ввиду того, что распалась на две уп-

рощенные задачи оптимального управления: первая - аналогичная задача без учета ограничения (62), которая уже решена нами; вторая - аналогичная задача с учетом ограничения (63), т.е. задача оптимального управления с закрепленным временем и закрепленными концами.

Исходная задача для "нерентабельных" культур хозяйства совпадает с предыдущей задачей до момента определения соответствующих вспомогательных переменных ψ_i, $i \in J_H$, из-за различия между указанными задачами в краевом условии (63). Поэтому в данном случае вышеуказанные вспомогательные переменные мы уже будем искать не из условий трансверсальности, а из граничного условия (63).

Воспользуемся промежуточными результатами предыдущей оптимизационной задачи. Для нашего случая согласно (54) и (56) без учета значений вспомогательных переменных, следует:

$$m_i(t) = \omega_i(t) + \frac{\lambda(t) k_i}{2 a_i \psi_i} \qquad (65)$$

$$\lambda(t) = -\frac{2[\Omega(t) - M(t)]}{\sum_{i=1}^{n} k_i / a_i \psi_i} \qquad (66)$$

Подставляя выражение (65) в (19), получим:

$$\bar{E}_i(t, m_i(t)) = a_i \omega_i^2(t) - \frac{\lambda^2(t) k_i^2}{4 a_i \psi_i} + p_i, \quad i \in J_H$$

Тогда согласно (5) с учетом (66), имеем:

$$y_i(t_c) = a_i \int_{t_n}^{\tau_c} \omega_i(t) dt - \frac{k_i^2}{a_i (\psi_i \sum_{i=1}^{n} \frac{k_i^2}{a_i \psi_i})^2} \int_{t_n}^{t_c} [\Omega(t) - M(t)]^2 dt, \quad i \in J_H \qquad (67)$$

Приравнивая соотношения (63) и (67), определяем неизвестные постоянные величины ψ_i, $i \in J_H$. Имеем:

$$\left(\psi_i \sum_{i=1}^{n} \frac{k_i^2}{a_i \psi_i}\right)^2 = \frac{k_i^2}{a_i} \int_{t_n}^{t_c} [\Omega(t) - M(t)] dt \; \frac{1}{a_i \int_{t_n}^{\tau_c} \omega_i(t) dt + p_i \tau - \frac{K_i}{k_i}}, \quad i \in J_H \qquad (68)$$

Обозначим $\varphi_i = \psi_i \sum_{i=1}^{n} \frac{k_i^2}{a_i \psi_i}$, $i \in J_H$

где φ_i, $i \in J_H$ - представляют собой известные величины (см. правую часть соотношения (68).

Обозначим $\alpha_i = \frac{1}{\psi_i}$, $i \in J_H$ (69)

Тогда с учетом (69), а также того, что $\psi_i = const$, $\varphi_i = const$,

вытекает:

$$\sum_{i=1}^{n} \frac{R_i^2 \alpha_i}{a_i} = const\,,\ i \in J_H \quad \text{или} \quad \alpha_i \psi_i = const\,,\ i \in J_H \ (70)$$

Таким образом, одно из решений системы уравнений (70) можно принять, например, за единицу, т.е. $\alpha_i = 1$, а оставшиеся неизвестные указанной системы определятся следующим образом:

$$\alpha_i = \frac{\varphi_1}{\varphi_i}\,, \quad i \in J_H\,, \quad i \neq 1$$

Окончательно получим: $\psi_1 = 1$, $\psi_i = \frac{\varphi_i}{\varphi_1}$, $i \in J_H$, $i \neq 1$ (7I)

Подставляя (7I) и (65) с учетом ограничения (46), определяем искомые рациональные режимы орошения для "нерентабельных" культур хозяйства. Аналогично предыдущей оптимизационной задаче определяются соответствующие оптимальные фазовые координаты.

Литература

I. В.Г.Болтянский. Математические методы оптимального управления.М., Наука, 1969, с.325-376.

2. В.Ф.Кротов, В.И.Гурман. Методы и задачи оптимального управления. М., Наука, 1973, с.93-136.

3. А.Е.Мелконян. Детерминированная постановка задачи оптимизации процесса водораспределения на орошение в регионе.- В кн.: Труды ВЦ АН Арм.ССР и ЕрГУ, т.ХШ, с.144-152.

4. Н.Н.Моисеев. Элементы теории оптимальных систем. Наука, М., 1975, с.70-73.

A dynamical model of optimal distribution of water in agriculture under restricted resources

A.E. Melkonjan, S.M. Manučarjan

Summary

In the paper the dynamical problem of optimal irrigation regimes of plants in a given region under the assumption of restricted resources is formulated and solved. The problem is transformed to the mathematical problem of optimal control and solved in the cases of both partial and global restrictions of the resources.

MTA SZTAKI Tanulmányok 202/1987 pp. 149-158

LOGIC PROGRAMMING AND THE ALGORITHMIC TYPE PROBLEMS

Katalin Pásztorné Varga
Computer and Automation Institut of Hungarian Academy of Sciences, Budapest, Hungary.

1. Introduction

The great advantages of logic programming are known, but logic programming is not generally used. The lack of computers having logical programming languages, some weakness of logic programming languages and the fact that the logic programming languages were used in laboratories. Consequently the solutions of the problems are algorithms, then all solved problem may be considered as algorithmic type ones. On the other side there is such a statement that logic programming is good only for the not algorithmic type problems. Then the following question has importance. Is it possible to distinguish /and how/ the algorithmic type problems from the logic programming friendly problems ? Let's examine whether the algorithmic or the logical approach gives a simpler or a more effective solution of a problem. Can it be a test? We mean that because of a real gap between the classical algorithmic and the logical way of thinking the result of the test will be not significant. To contribute to the reduction of that gap it is worth to examine how a logical approach can be found in case of

Research partially supported by Hungarian National Foundation for Scientific Research, Grant No 1066

several classes of problems having algorithmic solution. We show here some illustrations with a PROLOG background.

2. Some caracteristics of the algorithmic approach

Analysing an algorithm we examine its structure and the way of the data manipulation in it. The structure of a great proportion of algorithms is a tree. We consider here two main cases

1.a. The analysis of the problem is organised by a tree /e.g. syntactical analysis based on syntax tree/.

1.b. The control of the data manipulation is given by a tree /e.g. searching problems: to find a special set of data or some relations among data sets/

Another part of algorithms are the "direct" procedures. Here the solution of the problem is given by a definit sequence of mathematical operations. The structure of these type of algorithms is a tree in many cases, but in reality the data determines the path of the tree leading to the solution /for example the solution of the equation of the second degree/.

3. A logic approach

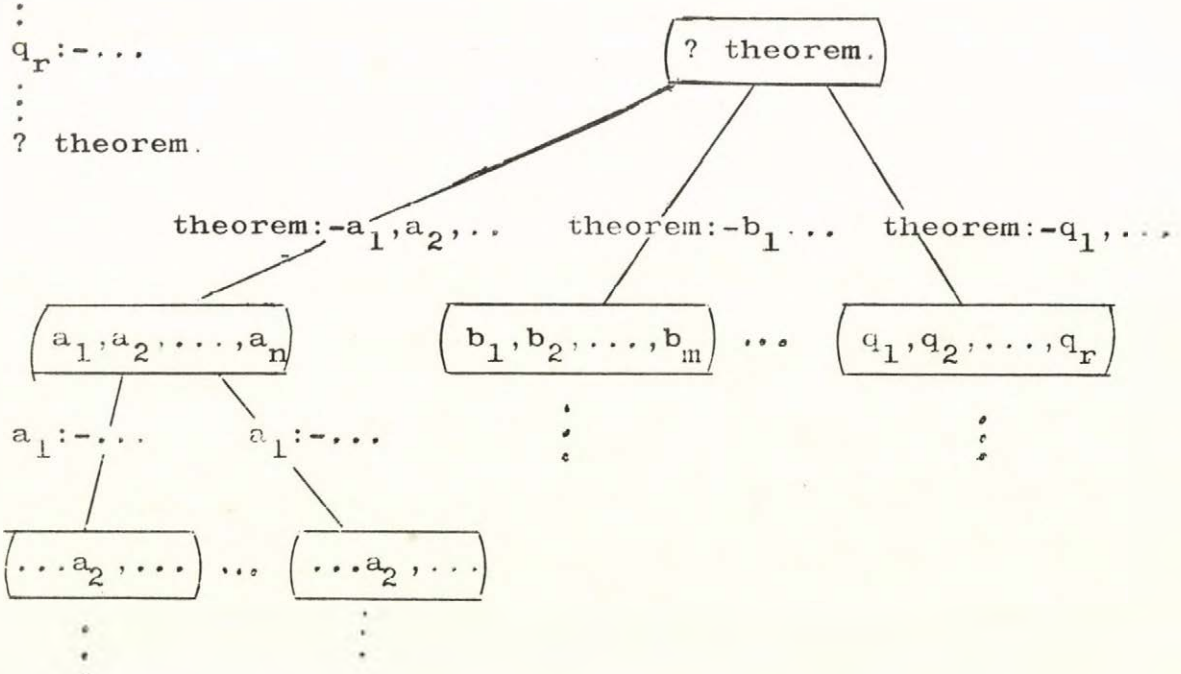
As the terminology related to logic programming is heterogeneus in the literature we give an enough general version of it.

A proof procedure may be considered as a logic programming language, where the proof procedure is a language with a logical calculus and a search strategy. The logic programming is a process to develop logic programs. A logic program is a set /or a sequence/ of special logic formulae. These formulae give such a description of the

problem which determines the solution by the proof procedure. Then this description is equivalent to the algorithm solving the problem.

The most part of the logic programming languages are PROLOG type languages. It means that the theorem proving system interpreting the logic programs is the first-order resolution principle. Generally, as in case of PROLOG, this proof procedure is the linear first-order resolution principle with a depth-first searching strategy. Then a logic program defines a tree, where the "theorem like" formulae are associated with the vertices /especially the theorem with the root/ and the "conditions" with the edges /Fig. 1./. A symbolic logic program and its searching tree is given below.

$theorem:\text{-}a_1,a_2,\ldots,a_n.$
$theorem:\text{-}b_1,b_2,\ldots,b_m.$
$\vdots$
$theorem:\text{-}q_1,q_2,\ldots,q_r.$
$a_1:\text{-}a_{11},a_{12},\ldots,a_{1t}.$
$\vdots$
$q_r:\text{-}\ldots$
$\vdots$
? theorem.



As the structure of the 1.a. and 1.b. type algorithms is a tree and the execution of these algrithms is connected with the traversing of these trees, it seems to be useful to investigate the following. How a logic program can be developped in these cases?

In the case 1.a. the tree of the algorithm gives direct help to construct the logic program. Generally the tree of an algorithm is composed of several types of subtrees containing relevant information concerning the problem. These informations /that is these subtrees/ may play the role of the axioms and the conditions of the problem. Then they give the base of the logic program. A typical example is the syntactical analysis of a CF language [1] .

Let's suppose that the grammar is given in normal form. Then the form of production rules is $R \rightarrow Q_1 Q_2 \ldots Q_n$. There may be recursive rules too. We can associate a "rule-tree" to any production rules /Fig..2./. If we allow to match the root of a rule-tree and a leaf of a rule-tree we can construct the syntax tree of the grammar /Fig. 3./. In case of recursive rules the syntax tree is potentially infinite.

Let be given a sequence of terminal symbols $J = j_1 j_2 \ldots j_m$. To prove that $J \in L(G)$ we must search such a subtree of the syntax tree whose leafs in order of the depth-first traversal give the sequence J /Fig. 4./. It follows then the rule-trees produce the formulae of a PROLOG program. If the rule is $R \rightarrow Q_1 Q_2 \ldots Q_n$ the associated PROLOG expression is $R{:-}Q_1, Q_2, \ldots, Q_n$. To express that w is a terminal symbol we use the PROLOG expression w. . Then in our case, where G has two production rules and three terminal symbols the PROLOG program solving the $J \in L(G)$ problem is the following.

```
M:-i.
M:-a,M,b.
a.
b.
i.
? aaibb.
```

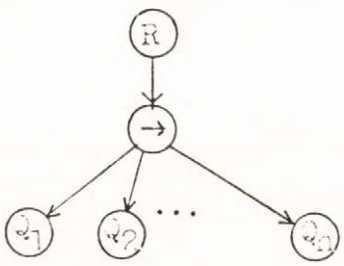


Figure 2

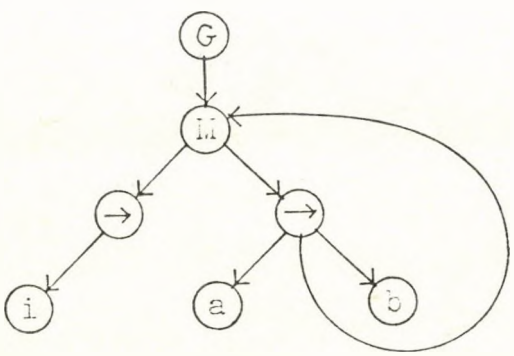


Figure 3

Syntax tree of

$G = (V_T = \langle a, i, b \rangle, V_N = \langle M \rangle,$

$M, H = \langle M \mapsto i, M \mapsto aMb \rangle)$

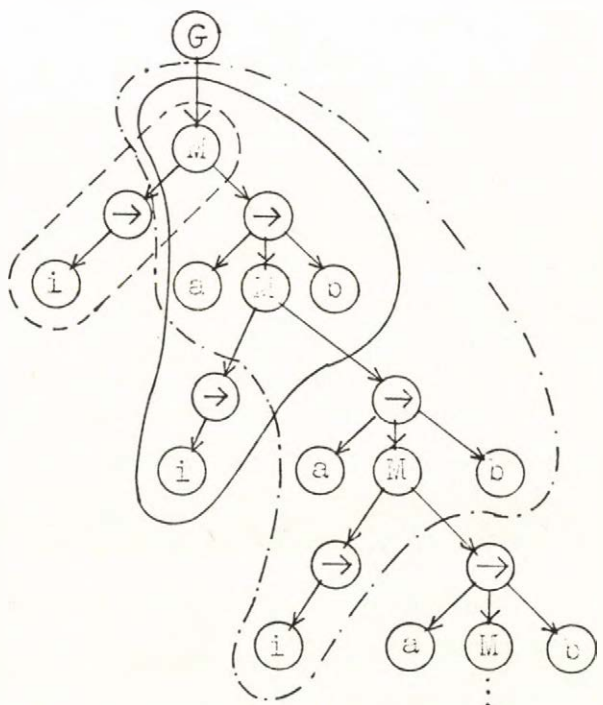


Figure 4

The subtrees giving the sentences: i - - - ,

aib ——— ,

aaibb - · - · - .

In case 1.b. the structure of the tree and the result of the data manipulations together give enough information to obtain a logic program. In this case the algorithm of

the problem depends in a natural way on the input data, but the intermediate results control the traversing of the tree [2] .

Let's investigate a significant problem to illustrate case. The recursive method of Morreale [3,4] to define all prime implicants for a DNF synthesis is well known. The recursive definitions of all prime implicants are:

$$P(0/0) = 0$$

$$P(1/1) = 0$$

$$P(1/0) = 1$$

$$P(f/g) = P(R_k(f/g)) \vee x_k P(f_{x_k=1}/g_{x_k=1},\ R_k(f)) \vee \bar{x}_k P(f_{x_k=0}/g_{x_k=0},\ R_k(f))$$

where f,g are Boolean functions, $f_{x_k=\alpha}$ $(\alpha = 0,1)$ is $f(x_1,\ldots,x_{k-1},\alpha,x_{k+1},\ldots x_n)$,(f/g) defines a partially defined Boolean function, $P(.../...)$ denotes all prime implicants of the Boolean function defined by $(.../...)$, the concatenation of two expressions denotes their conjunction. R is a reduction operator.

These definitions determine a special tree where the number of the output edges of a vertex is at most three /Fig. 5./ according to the result of the used rule from /1/. This tree is built from the subtrees of concrete forms of the rules /e.g. the subtree associated to the fourth rule of /1/ is marked with - - - sign on Fig. 5. A path ending with a 1 vertex gives the prime implicant which is the conjuction of the variables associated to the edges of this path. We can state that

1. The form of a $P(f/g)$ tree depend on the function (f/g) .
2. A depth-first strategy is suitable to traverse the tree.
3. A complete traversing is needed.

4. All results obtained on a path must be conserved until the decision according to the information being in the leaf of the path.

After all by a depth-first traversing of this tree all prime implicants are obtained.

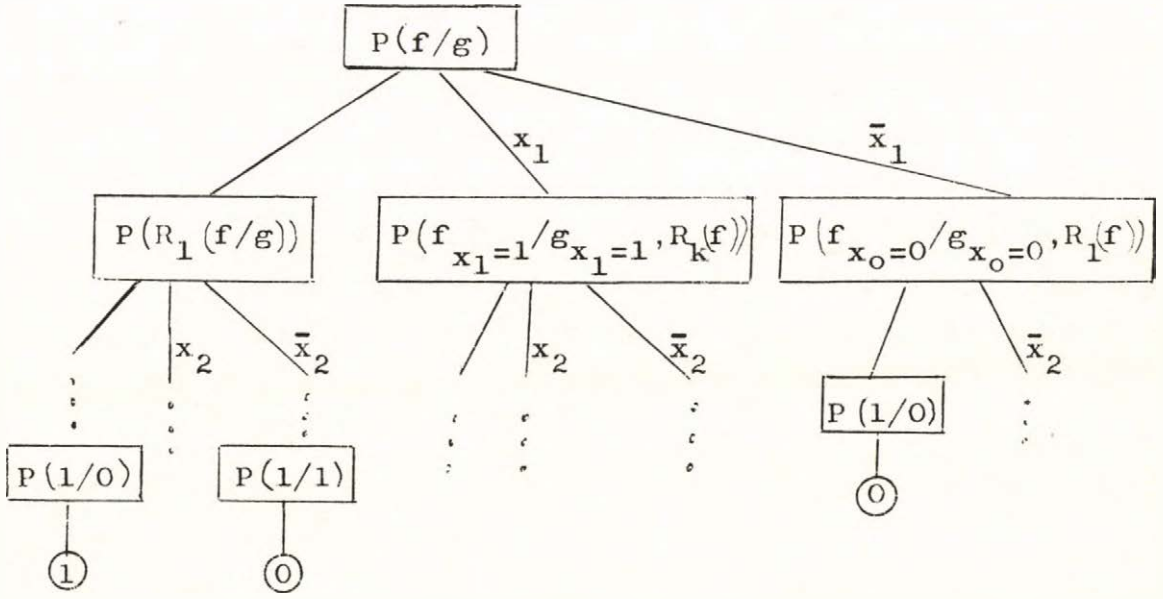


Figure 5.

Similar to the case 1.a. the form of the PROLOG program is:

P(0/0):-0.

P(1/1):-0.

P(1/0):-1.

$P(f/g):-P(R_k(f/g)) \quad ,P(f_{x_k=1}/g_{x_k=1}, R_k(f)) \quad ,$

$P(f_{x_k=0}/g_{x_k=0}, R_k(f)) \quad .$

As the data manipulations are very tiring in PROLOG

to execute the operators P and R some interface to another language is needed. We saw that the traversing of the tree is controlled by the result of the data manipulation. As in case of PROLOG it is controlled by a special search strategy an the unification technic the preservation of the resulting prime implicants is not easy in a natural way. These technical problems show that in such cases the transformation of the problem is not automatic,because of the unsuccesful and the succesful backtrack. In similar cases it would be useful to have only one sort of backtrack. After all it seems to be advisable a new investigation of the original problem, where perhaps a backtrack free solution can be obtained [5] .

References

1.Katalin Pásztorné Varga: The Use of finite Automatas in the processing certain types formal languages. Algebra és Számitástudomány Konferencia Salgótarján 1984.

2. Katalin Pásztorné Varga: Synthesis of Boolean functions using PROLOG. Proc. Fourth Hung. Computer Sci. Conf. Győr, 1985.

3. E Morreale: Recursive operators for prime implicants and irredundant normal form determination. IEEE Trans. on Computers. C-19, 1970. 504 - 509.

4. J. C. Torgue, K. Pásztor Varga, P. Azema: Couverture irredondante des fonctions Booléennes definies par leur monômes vrais et faux - fonctions simultanées Acta Cybernetica Tom 2. 1975.

5. Katalin Pásztorné Varga: Logic minimisation problems in VLSI synthesis. Proc. of the First Hungarian CCC '87 Conference. Gyöngyös 1987.

Logic programming and the algorithmic type problems

Katalin Pásztorné Varga

Summary

Logic programming gives great possibility in problem solving. As the way of thinking in the logic programming and in the classical programming are quite different, there are difficulties to solve classical problems with logic programming. Some approach to obtain logic programs in case of some classes of "algorithmic type" problems is showed. The difficulties and possibilities are illustrated by examples.

MTA SZTAKI Tanulmányok 202/1987 pp. 159-170

К АДАПТИВНОМУ СИНТЕЗУ СТРАТЕГИЙ УПРАВЛЕНИЯ

Э.М.Погосян

I.I. Формулируется проблемы адаптивного (индуктивного) синтеза решений комбинаторных оптимизационных проблем распознавания, классификации, поиска стратегий и других в заданных классах алгоритмов, в частности экспертных системах. Исследуются возможности построения методов адаптивного синтеза стратегий управления в классах конечно-порожденных и продукционно-переборных алгоритмов полиномиальной сложности.

I.2. Рассматриваемые нами оптимизационные проблемы, $L = \langle X_L, Y_L, \rho_L, F_L, J_L \rangle$, определяются множествами входных и выходных записей X_L и Y_L, соответственно, некоторым, вообще говоря, квазифункциональным отношением $\rho_L \subseteq X_L \times Y_L$, множеством $F_L = \{f \mid f: X_L \to Y_L\}$ -алгоритмов, предназначенных для моделирования[I)] ρ_L и уточненных для конкретного типа машин, например адресных [I] и оператора качества J_L алгоритмов из F_L. Требуется найти оптимальный по J_L алгоритм f_L^* из F_L.

Пусть $A = \{L\}$ -непустой класс оптимизационных проблем. Предполагается, что проблемы класса A можно охарактеризовать некоторым параметром h -размером, являющимся определенной мерой сложности задания описаний проблем. Так, размером проблем на графах может служить число вершин графов.

Исходное представление рассматриваемых нами проблем является к о м б и н а т о р н ы м, что выражается, в частности, в следующих особенностях:

I. Высокой степени индивидуальности, различия как проблем из A, так и элементов из X_L при $L \in A$. Вследствие этого

I) Отношение $\rho_L \subseteq X_L \times Y_L$ квазифункционально, если $\forall x \in X_L \; \exists y \in Y_L$ $\langle x, y \rangle \in \rho_L$ и $\forall x x_2 \in X_L$, если $X_i = \{y \mid \langle x_i, y \rangle \in \rho_L\}$, $i = 1,2$ и $X_1 \cap X_2 \neq \emptyset$, то $X_1 \subseteq X_2$, либо $X_2 \subseteq X_1$. Алгоритм f предназначен для моделирования ρ_L, если $\forall x \in X_L \; f(x) \in Y_L$ [2].

построение решения f_L^* и вычисление $f_L^*(x)$ при $x \in X_L$ связано, вообще говоря, с длительным процессом изучения, приспособления к специфике L и x требующего, вообще говоря, дополнительной информации о них извне.

2. Оператор J_L, быть может, нерекурсивен, неаналитичен, или трудновычислим, вследствие чего непосредственное использование известных методов оптимизации, основанных на аналитических свойствах J_L, становится невозможным.

2.1. В проблеме синтеза комбинаторных алгоритмов, интенсивно исследуемой в настоящее время [2,3,4], требуется для заданного класса комбинаторных проблем A найти алгоритм ω_A, который по спецификации произвольной $L \in A$ строит наилучшее в заданном классе B_A приближение $\omega_A(L) = f_L$ к решению f_L^*. При этом алгоритм f_L должен быть найден с возможно меньшими затратами вычислений и памяти.

Тем самым, если при известном разрешающем алгоритме G_A для A решение для L получается простой подстановкой описания L в G_A, в проблеме синтеза решение для L строится синтезатором в результате, вообще говоря, трудоемкого процесса.

Проблемы синтеза характеризуются способом спецификации проблем (с исходным полным описанием, с последовательным заданием пар вход-выход и др.), по методу синтеза алгоритмов (логический, основанный на поиске доказательства соответствующих теорем, на основе теорем о рекурсии, на основе обобщения примеров решения индивидуальных задач и др.), по типу вычислений ((не)детерминированные, альтернирующие, предельные и др.) по классу исследуемых проблем (с параметрическим отличием решений индивидуальных задач, вычислительные задачи, полиномиально доказуемые в заданном исчислении и др.), по классу синтезируемых алгоритмов (конечные автоматы, ЛИСП-, ПРОЛОГ-программы, продукционные системы и др.), по критериям оценки эффективности синтеза (временная, объемная и др. сложности, локальность вычислений и др.).

Нами исследуется проблема а д а п т и в н о г о с и н т е з а комбинаторных алгоритмов, выделяемая следующим образом:

а) Спецификация проблемы может быть задана либо полностью до начала вычислений, либо посредством перечисления пар вход-выход в процессе вычислений, дополненных быть может описанием работы искомого алгоритма на этих парах;

б) Метод синтеза основан на обобщении примеров решений индивидуальных задач;

в) Рассматривается модификация алгоритмов предельных вычислений (АПВ). В соответствии с [5] АПВ по спецификации проблемы посредством перечисления пар вход-выход выдает последовательность чисел-гипотез о номере соответствующей функции. Последовательность гипотез может быть, вообще говоря, бесконечной. Предел последовательности существует, если гипотезы с некоторого момента не изменяются. В противном случае значение алгоритма на данном входе не определено.

Нами вводятся следующие дополнения к схеме АПВ:

1) наряду со спецификацией проблем посредством пар вход-выход допускается ее исходно полное описание;

2) в процессе вычислений АПВ может обращаться к оракулу с вопросами из конечного списка;

3) ресурсы памяти и вычислений, затрачиваемые в процессе построения каждой из гипотез , представляются ω_A ограниченными по стоимости порциями (локально). (Данное требование порождено не только необходимостью быстрой окупаемости затрат, но также, быть может, локальностью доступного для анализа объема информации или возможностью оценивания качества лишь локальных преобразований гипотез об f_L^*);

4) время работы и объем предоставляемой памяти для гипотез, строимых АПВ, полиномиально ограничены по n . Тем самым исследуются классы проблем, решения которых или приближения к ним при заданной форме представления, например, в форме систем продукций), ограничены по времени работы и требуемой памяти.

2.2. Сформулируем проблему а д а п т и в н о г о с и н т е з а а л г о р и т м о в - решений проблем из класса A .

Пусть проблемы из A имеют один и тот же оператор качества алгоритмов J, определяющий линейный порядок на множествах F_L, $L \in A$, а алгоритмы F_L уточнены для одного и того же типа машин $\mathfrak{M}_1$.

Пусть также F_A – множество алгоритмов предельных вычислений ω, уточненных по аналогии с [5] для машины $\mathfrak{M}_2$ с оракулом, со списками C_0 и C_1 -взвешенных элементарных вычислений и вопросов к оракулу, соответственно, B_A -класс синтезируемых алгоритмов, в рамках которого ищутся решения проблем из A.

Произвольную неубывающую по каждому из своих аргументов n и i функцию $\Delta(n,i)$ назовем ф у н к ц и е й л о к а л ь н о с т и. Через $\Delta^{\omega}(n, h_i, h_{i+1})$ обозначим стоимость элементарных вычислений и вопросов к оракулу, используемых АлгВ ω от момента выдачи h_i до выдачи h_{i+1}.

Предполагается, что $\forall \omega \in F_A$, если на входе ω при $t=0$ имеется определенная информация относительно проблемы $L = \langle X_L, Y_L, \rho_L, F_L, J \rangle$ в частности ее полная спецификация или примеры решения индивидуальных задач, то:

1) гипотезы h_i на выходе ω при каждом $i \geq 0$ принадлежат $F_L \cap B_A$;

2) ω предельно вычисляет некоторый алгоритм $\omega(L) = f_L$ из $F_L \cap B_A$;

3) $\Delta^{\omega}(n, h_i, h_{i+1}) \leq \Delta(n,i)$ при $i \geq 0$.

Пусть K – заданный оператор оценки качества работы синтезаторов из F_A при построении f_L, $L \in A$.

В проблеме адаптивного синтеза $\mathcal{L}^A$, определяемой системой $\langle A, n, \mathfrak{M}_1, J; F_A, \mathfrak{M}_2, C_0, C_1, \Delta, B_A, K \rangle$ и указанными выше условиями a, b, β, требуется построить синтезатор $\omega^* \in F_A$ такой, что для каждой $L \in A$ алгоритм $\omega^*(L)$ является наилучшим решением для L в классе $B_A \cap F_L$, а ω^* при этом является наи-

лучшим решением для L в классе $B_A \cap F_L$, а ω^* при этом является наилучшим по K в классе F_A.

Замечание I. Вычисление значений операторов J, K и выбор принципов, упорядочения по ним алгоритмов из F_L, F_A сопряжен в общем случае с существенными трудностями. Поэтому решать эти вопросы целесообразно при конкретных A и F_A. Тем не менее для определенности дальнейших рассмотрений нами вводятся следующие естественные операторы и упорядочения.

Пусть $J(L,\omega)$ -значение оператора J на f_L и $J(n,\omega) = \min\limits_{\lfloor L \rfloor = n} J(L,\omega)$, где $\lfloor L \rfloor$ размер L. Пусть также $T(L,\omega)$ и $S(L,\omega)$ -число тактов выдачи гипотез и число всех номеров ячеек памяти, использованных m_{L_2}, соответственно, до момента стабилизации гипотез в проблеме L, $T(n,\omega) = \max\limits_{\lfloor L \rfloor = n} T(L,\omega)$, $S(n,\omega) = \max\limits_{\lfloor L \rfloor = n} S(L,\omega)$.

Развиваемая нами идея окупаемости очередной порции вычислений $\Delta(n,i)$ посредством возможно большого усиления гипотезы h_{i+1} относительно h_i, требует соответствующего выражения в критериях качества синтеза. Таковым для нас является частость усиления $\gamma(n,\omega) = \min\limits_{\lfloor L \rfloor = n} \gamma(L,\omega)$, где

$$\gamma(L,\omega) = \frac{\sum_{i=0}^{T(L,\omega)} \tilde{sg}(J(h_{i+1}) - J(h_i))}{T(L,\omega)},$$

$\tilde{sg}(J(h_{i+1}) - J(h_i))$ равно 1, если гипотеза h_{i+1} превосходит по J гипотезу h_i, и равно нулю, в противном случае.

В проблеме $\mathcal{L}^A$ естественно искать ω, доставляющий наилучшее по $J(n,\omega)$ и $\gamma(n,\omega)$ решение с возможно меньшими значениями $T(n,\omega)$ и $S(n,\omega)$. Однако непосредственное упорядочение ω по каждому из операторов J, γ, T или S не представляется возможным, поскольку, во-первых, неизвестны адекватные критерии упорядочения операторов вообще, во-вторых, операторы T, и следовательно γ, в общем случае невычислимы [5]. Поэтому нами рассматриваются лишь подмножества алгоритмов из F_A, допускающие упорядочение по подходящей для каждой из J, S, γ, T шкале. В частности, по порядку роста значений операторов J, S, γ, T, по шкале $n < n^2 < \cdots < \exp n < (\exp n)^2 <$ и т.д.

Наилучшее $\omega \in F_A$ для определенности предлагается понимать как максимальный элемент в линейном упорядочении, индуцированном лексикографическим упорядочением значений четверки

$\langle J, S, Ч, T\rangle$. При этом значения каждой из компонент $J, S, Ч, T$ предполагаются упорядоченными по шкалам, конкретизированным после выбора класса A.

С другой стороны, вычислимости операторов предлагается достичь соответствующим выбором класса A . В частности, для конечных комбинаторных проблем данное требование, как правило, выполнимо.

Итак, критерий K конкретизируется нами в форме $\langle S, Ч, T\rangle$. Ясно, что она не единственна. Естественен, например $K=\langle Д, S, Ч, T\rangle$.

2.3. Рассмотрим примеры проблем адаптивного синтеза.

Пример I. Оптимизационная проблема $Л$ п о и с к а к р а т ч а й ш е г о п у т и (ПКП) из произвольной вершины x конечного графа G со взвешенными ребрами в произвольную целевую $x_к$ из множества $X^к$, является комбинаторной. В ней $X_Л$ есть множество всех вершин G, $Y_Л$ - множество всех путей по всем $x \in X_Л$ в каждое $x_к \in X^к$, $\rho_Л = \{\langle x, y\rangle \mid x \in X_Л \,\&\, y$ - кратчайший (наименьшей стоимости) путь из x в $X^к\}$, $F_Л$ - множество алгоритмов, предназначенных для моделирования $\rho_Л$. Оператор качества $J_Л$ определяется набором $\langle T_0, T_0 \cdot S_0\rangle$, где $T_0(f)$ - оператор отклонения, равный $\sum_{x \in X_Л} c(f(x)) - c^*(x)$, где $c(f(x))$ и $c^*(x)$ -стоимости путей из x в $X^к$, построенных $f \in F_Л$ и кратчайшего, соответственно, $T_0(f)$ - максимум по стоимости вычисления алгоритмом f путей из произвольной $x \in X_Л$ до $X^к$ на заданной машине M_1 , $S_0(f)$ - память, используемая для записи f в M_1 . Лексикографический порядок на двойках $\langle T_0, T_0 \cdot S_0\rangle$ индуцирует соответствующее упорядочение $F_Л$. Выбор критерия в виде произведения $T_0 \cdot S_0$ в настоящее время обосновывается нами лишь необходимостью совместного анализа T_0 и S_0 .

При $A=\{$ПКП$\}$, h - число вершин графа, J - определенном как $\langle T_0, T_0 \cdot S_0\rangle$ и некоторых значениях остальных параметров α^A , получаем проблему адаптивного синтеза алгоритмов поиска кратчайшего пути из произвольной вершины произвольного графа.

Пример 2. Вопросы адаптивного синтеза сравнительно хорошо исследованы для проблем п о и с к а а л г о р и т м о в к л а с с и ф и к а ц и и (ПАК), формулируемых следующим образом.

Пусть X_L - конечное множество (универсум), $n=|X_L|$, $Y_L=\{0,1\}$, $T=\{(X_1,X_0)\,|\,X_1,X_0\subseteq X_L \;\&\; X_1\cap X_0\neq\emptyset\}$ - пары множеств (ПМ) из X_L (частично определенные предикаты на X_L), $T^n=\{(X_1,X_0)\,|\,(X_1,X_0)\in T \;\&\; X_1\cup X_0=X_L\}$ -абсолютные ПМ (предикаты на X_L), $T'\subseteq T^n$. Пусть также ρ_L - конкретный предикат из T', известный только оракулу, Q_L - заданный частично определенный предикат из T, определенным образом связанный с ρ_L, F_L - множество алгоритмов, предназначенных для моделирования ρ_L и именуемых а л г о р и т м а м и к л а с с и ф и к а ц и и X_L, $J_L=\langle J_0, T_0, S_0\rangle$, где $J_0(f)=\sum_{x\in X_L}(f(x)-\rho_L(x))^2$ - хеммингово расстояние классификации, полученной $f\in F_L$, от истинной ρ_L, $T_0(x)$ - максимальное по всем $x\in X_L$ время вычисления значений $f(x)$ (время распознавания), $S_0(f)$ - память, используемая при хранении f в машине. В ПАК L требуется построить оптимальный по J_L алгоритм классификации f_L^* из F_L.

Класс $A=\{\text{ПАК}\}$ определяется множеством всех конечных множеств X_L. Если f_L^* может быть построена без обращения к оракулу, с использованием лишь свойства Q_L, то ПАК называется инициально полной; в противном случае - инициально пустой. Легко видеть, что при конкретных Q_L получаем проблемы ПАК, эквивалентные ограниченным проблемам распознавания слов со свойством Q_L [6]. В частности, при X_L - множестве n-вершинных графов, $Q_L(G)$ - предикате "граф G имеет гамильтонов цикл", получаем ограниченную проблему L распознавания гамильтонового цикла в графах с числом вершин не более n (ГЦ^n).

Синтез классификаций для инициально пустых ПАК рассматривается в основном в теории распознавания образов [2,7], а для инициально полных - в теории комбинаторной оптимизации (распознавание слов с заданными свойствами. При этом итеративность в первом случае обусловлена, как правило, ограниченностью доступной для анализа информации, а во втором - требованием окупаемости.

Проблемы ФОС_i - формирования образов ситуаций типа i, рассмотренные в [2], являются разновидностью ПАК. При этом ФОС I.2 - инициально пустые, а ФОС 3,4 - полные ПАК.

Отметим также, что ПАК выделена из проблемы расшифровки

описаний (ПРО), определенной в [2] , поскольку ПРО можно исследовать в более общей постановке проблемы адаптивного синтеза в форме ПАК .

Пример 3. Э к с п е р т н ы е с и с т е м ы (ЭС) предназначены для накопления, хранения и оперативной выдачи экспертных знаний в конкретных предметных областях.

В нашей формализации синтеза ЭС в виде $\mathcal{L}^A$, класс A есть совокупность определенных предметных областей. Спецификация конкретной ЭС задается парами вход-выход, или примерами ЭС на этих парах, метод синтеза - основан на анализе и обобщении указанных пар, синтезируемые ЭС, как правило, принадлежат классу систем продукций.

3.1. В настоящее время эффективные алгоритмы синтеза известны для сравнительно узких и однородных классов A . При этом эффективность достигается в результате индивидуального изучения каждого класса и высокой специализации алгоритмов. Однако при быстром росте числа различных комбинаторных проблем вопрос построения алгоритмов синтеза, объединенных единой идеологией, сравнительно независящей от разнообразия классов, становится весьма острым. Поэтому наши исследования подчинены специальному требованию возможно большого разнообразия классов A_i при одном и том же или сходных алгоритмах синтеза

Для обеспечения указанного требования исходными для нас являются две идеи. Первая из них связана, в частности, с предположением конечной порождаемости и локальной сравнимости алгоритмов-гипотез класса A , что позволяет организовать направленный перебор всевозможных наборов малой длины этих гипотез, выделяя в каждой из них наилучшую. На этом пути нами получено полиномиальное решение $\mathcal{L}^A$ для конкретного класса проблем поиска оптимальных стратегий (ПОС) в конечных играх типа шахматных.

Вторая идея относится к формализации процедур изучения, познания новых проблем человеком. Нами рассматривается одна из формализаций класса познавательных алгоритмов, так называемые продукционно-переборные алгоритмы, при выборе которых, в целях эффективизации и универсализации поиска решений инди-

видуальных проблем, наряду с критериями достаточной структурированности, модульности и мотивов относительно принадлежности ему указанных решений, служили также аналогии с сравнительно общими элементами методов решения проблем человеком. В классе проблем ПОС выделены достаточные условия адекватности поиска оптимальных планов в системах продукций, поиску в множествах порожденных ими стратегий управления.

Приведем краткое описание указанных результатов.

4.1. Рассмотрим вопросы построения полиномиальных алгоритмов для $\mathcal{A}^{A_4}$ при $A_4 \subseteq \{ПОС\}$. Алгоритм $\omega \in F_A$ полиномиален по критериям из $\chi_1, \chi_1 \subseteq \{\mathcal{J}, \Delta, S, \mathcal{U}, T\}$, если $\mathcal{T}_0 = 0$, $\mathcal{U} = 1$, а каждый из остальных ограничен сверху полиномом от n и i. Аналогично, множество A_4 полиномиально в классе алгоритмов B, $B \cap F_L \neq \emptyset$, $L \in A_4$, если решения проблем из A принадлежат B, а T_0 и S_0 ограничены полиномом от n.

4.2. Исходным при определении ПОС является конечное дерево игры $\mathcal{D}(n)$ с длиной описания n - размером ПОС. $\mathcal{D}(n)$ определяется множествами позиций $\mathcal{P}$, правил игры $\mathcal{M}$, заключительных позиций Q. Стратегия G из позиции P заключительная, если концевые вершины G принадлежат Q. Мера выигрышности стратегий задается оператором $\Phi: \bigcup_{P \in \mathcal{P}} \mathcal{G}(P) \to [0,1]$, где $\mathcal{G}(P)$ - множество заключительных стратегий из P. Стратегия G из $\mathcal{G}(P)$ оптимальна, если она максимизирует Φ в классе $\mathcal{G}(P)$; $\mathcal{G}^*(P)$ - множество оптимальных P-стратегий.

Пусть $\tilde{F} = \{f \mid f: \mathcal{P} \to \bigcup_{P \in \mathcal{P}} \mathcal{G}(P) \;\&\; \tilde{f}(P) \in \mathcal{G}(P)\}$. Алгоритмы управления множества F реализуют функции из $\tilde{F}$, отображая ветки ходов из $\tilde{f}(P)$ в ходы. Каждому $f \in F$ соответствует единственная $\tilde{f} \in \tilde{F}$. Алгоритм f точный, если $f(P)$ - оптимальная стратегия. Отклонение $\mathcal{J}(P, f)$ алгоритма f от точного, f^1, в P равно $\Phi(\tilde{f}^1(P)) - \Phi(\tilde{f}(P))$, а среднее отклонение $\mathcal{J}(f) = \frac{1}{|\mathcal{P}|} \sum_{P \in \mathcal{P}} \mathcal{J}(P, f)$. Временная сложность $T_0(f)$ равна максимальному времени разыгрывания произвольной партии алгоритмом f, а емкостная - максимальному числу использованных при этом ячеек памяти вместе с записью f (обозначение - $S_0(f)$).

В проблеме ПОС при дереве игры $\mathcal{D}(n)$ и отношении $\rho \subseteq \bigcup \{P\} \times \mathcal{G}^*(P)$, требуется построить точный по $\mathcal{J}$ и оптимальный по произведению $T_0(f) \cdot S_0(f)$ алгоритм f^* из F.

4.3. Произвольный алгоритм управления $f \in F$ можно задать двудольным графом управления с помеченными ребрами. Пусть $A_1 \subset \{ПОС\}$ такое, что для $\mathcal{L} \in A_1$ выполняется: I. в языке первого порядка, интерпретированном для $\mathcal{L}$, существует описание реберного покрытия некоторого точного алгоритма управления $\mathcal{L}$, с длиной, не превышающей полинома от размера $\mathcal{L}$; 2. алгоритм, соответствующий описанию произвольного реберного покрытия точного алгоритма $\mathcal{L}$, также точный.

ТЕОРЕМА I. Множество A_1 полиномиально в $UF_{\mathcal{L}}$, $\mathcal{L} \in A_1$.

Пусть длина $b(h)$ зоны нестабильности в абсолютном упорядочении алгоритмов управления [2] и число партий при проведении одного матча между ними не более чем полиномиально по h. Тогда справедлива

ТЕОРЕМА 2. Для $\mathcal{L}^{A_1}$ можно построить полиномиальный по $\Delta(h,i)$ и $S(h,\omega)$ алгоритм синтеза ω точности $2b(h)$.

5. Продукции задаются тройками EgF, где E - образ класса ситуаций, g - стратегии управления из E, F - образ достигаемых при управлении g целевых ситуаций.

Продукционно-переборные ($п/п$) алгоритмы совмещают свойства продукционных систем и переборных алгоритмов. При поступлении индивидуальной задачи X некоторой проблемы $\mathcal{L}$ п/п алгоритмы в первую очередь пытаются найти решение X в рамках продукционных систем (посредством поиска в базе знаний - продукций). Лишь при неудаче этой попытки алгоритмы переходят к поиску решения посредством некоторой процедуры перебора в исходном пространстве решений, определяемом $\mathcal{L}$.

Пусть Φ - оператор выигрышности, A определен естественным образом на множестве продукций.

При заданной позиции P п/п алгоритм определяет некоторую стратегию G из P и соответствующее продукционное покрытие $\tilde{G}$, к которым предъявляются следующие требования.

$\underset{\sim}{P}$I. Если в множестве концевых вершин A_i i-го слоя продукций $\tilde{G}$ $\Phi(A_i)=\Phi(F_i)$, $F_i \subseteq A_i$, то после перехода к $i+1$-слою равенство сохраняется для порождаемых A_i и F_i стратегий.

Ф4. Выигрышность стратегии G определяется по выигрышности концевых вершин G.

Ш. Если A - множество концевых вершин Eg -стратегии, порожденной продукцией EgF, то $\Phi(A)=\Phi(F)$.

Справедливо следующая

ТЕОРЕМА 3. Если для продукционного покрытия $\tilde{G}$ с корнем P, порожденного последовательность продукций (п л а н о м) $E_1g_1F_1, \ldots, E_kg_kF_k$, выполнены условия Ш, РI, Ф4 и $P \in E_1$, $E_i = F_{i-1}$, $2 \le i \le k$, то $\Phi(G)=\Phi(F_k)$.

Утверждение I дает основу для перехода от процедур перебора в множестве стратегий, порожденных п/п алгоритмами, к процедурам поиска в множестве планов, вообще говоря, с много меньшей сложностью.

Расматривается выполнимость условий утверждения I. В частности, возможность формирования образов целевых ситуаций продукций при требовании Ш.

Литература

I. Ахо А., Хопскрофт Дж., Ульман Дж."Построение и анализ вычислительных алгоритмов".
2. Погосян Э.М. Адаптация комбинаторных алгоритмов, изд-во АН Арм.ССР,I983.
3. Автоматический синтез программ. АН ЭССР,Таллин, I983.
4. Барздинь Я.М. Некоторые правила индуктивного вывода и их применение.-Семиотика и информатика,вып.I9,I982,с.59-88.
5. Angluin D.,Smith C.H. Inductive inference:Theory and Methods. Computing Surways of the ACM,v.15,N3,1983,237-269.
6. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи", М., I982.
7. Журавлев Ю.И. Алгебраический подход к задачам распознавания, Проблемы кибернетики, в.33, I978.

the adaptive synthesis of control-strategies

E.M. Pogosjan

Summary

ıe problem of adaptive (inductive) synthesis of solutions
f combinatorial optimization problems of recognition,
lassification, strategy-searching and others are studied
thin a given set of algorithms(say, in expert systems).
ıe possibility of constructing methods of adaptive
ynthesis of a control-strategy in some classes of poly-
›mial algorithms are also investigated.

MTA SZTAKI Tanulmányok 202/1987 pp. 171-174

A SHORT NOTE ON MEASURES OF QUALITATIVE VARIABLES FOR DISEASE STUDIES

István Ratkó
Computer and Automation Institute
Hungarian Academy of Sciences

Many variables are usually treated in medical studies as categorised variables.

The aim of this paper is to describe a method for putting a metric on such a so-called "composite" variable. The method is based on Lancaster's theorem [1].

Theorem: If a bivariate distribution of (X,Y) can be obtained from a bivariate normal distribution by separate transformations on X and Y, then correlation in the transformed distribution cannot exceed in absolute value the correlation in the bivariate normal distribution. ■

How can we use this theorem?

Suppose we have two populations, one consisting of males who have already suffered from a concrete disease and the other consisiting of males who have not

---

Research partially supported by Hungarian National Foundation for Scientific Research, Grant No 1066

had this disease. We could then do a contingency table with the row classification as the disease and non disease and the columns as the classifications given by the composite variable.

The problem now is to choose a measure X for the row classification and a measure Y for the column classification to maximise the correlation between these two variables. Lancaster's theorem would indicate that the measures so obtained should be similar to that obtained by categorisation of the ranges of variables having bivariate normal distribution.

The main advantage of this technique is that it reduces the number of variables.

The steps of the procedure are[1]:

Let $C=[c_{ij}]$ be the 2 x r contingency table, where r is the number of categories of the composite variable. Let

$$c_{i.} = \sum_j c_{ij}$$

$$c_{.j} = \sum_i c_{ij}$$

$$n_{ij} = c_{ij} / (c_{i.} c_{.j})^{1/2}$$

$$N = [n_{ij}]$$

$$A = NN'$$

Let R^2 be the second largest eigenvalue of $\mathbf{A}$ (the largest is always 1) and $\underline{u}$ be its corresponding eigenvector.

If $\underline{u}' = [u_1, u_2]$ then

$$x_1 = u_1 c_{1.}^{1/2}$$

$$x_2 = u_2 c_{2.}^{1/2}$$

$$y_j = (\sum_i c_{ij} x_i)/Rc_{.j} \quad (j=1,2,\ldots,r)$$

R is the correlation between X and Y. Since $\mathbf{A}$ has rank 2 and the sum of the eigenvalues equals the trace, then R^2 is equal to (tr $\mathbf{A}$ - 1). In the case that there is a large correlation between X and Y it would be expected that the Y variable would be a good predictor of the disease. However, since prediction is a multivariate problem, such statements must be regarded as tentative.

References:

1. Kendall MG, Stuart A: The advanced theory of statistics, London: Charles Griffin,2nd Edn. Vol 2., 1967.

A short note on measures of qualitative variables for disease studies

I. Ratkó

Summary

Many variables are usually treated in medical studies as categorised variables. The paper describes a method of putting a metric on such a "composite" variable. The method is based on Lancaster's theorem concerning the correlation in the bivariate distribution (X,Y) that is obtained from a bivariate normal distribution by separate transformations on X and Y.

MTA SZTAKI Tanulmányok 202/1987 pp. 175-180

REALIZATION OF LARGE-SIZED PHARMACOLOGICAL EFFECT EXAMINATION WITH MICROCOMPUTER

I.Ratkó[1], P. Kerékfy[1], A. Kráml[1], A. Kiss[1], M. Ruda[1], J.Soltész[1], J.Duba[2], M.Csukás[2], E.Farkas[3], G.Maróti[3]

[1]Computer and Automation Institute
Hungarian Acedemy of Sciences
Budapest
H-1132 Victor Hugo u. 18-22

[2]National Cardiological Institute
Budapest

[3]Richter Gedeon Pharmaceutical Company
Budapest

The authors and their colleauges have been engaged in developing health care systems since 1972. The paper reports on the authors' special works on microcomputers.

The so-called **RGPC-register** (Richter Gedeon Pharmaceutical Company) contains data of Hungary, approximatly 2500 cases with (special) acute myocardial infarction.

The initation of patients into the examination occurs on 5-th-14-th day of sure myocardial infarction.

The aim of the examination are: a) the appreciation of efficiency of the treatment with medicines **Tobanum** and **Rabenid** in the secondary prevention of acute myocardial infarction b) examina-

---

Research partially supported by Hungarian National Foundation for Scientific Research, Grant No 1066

tion of side effects.

The study is based on fixed sample statistical methods. The method of the examination is double-blind, multicentral, random experiment controlled with **Placebo.**

Our system perform mathematical statistical analysis of the effect of medicines [1].

One record of the register-file have 1064 data-items, the length of the record is 4614 characters.

Until now we have used five kinds of data forms:

(a) form of Basic Medical Examination,which is filled in at the time of the initation of patients into the examination.

(b) two kinds of form of Following-up Medical Examinaton, which are filled in the 1-st, 2-nd, 3-rd, 4-th, 5-th, 6-th, 8-th, 10-th and 12-th month after basic medical examinations.

(c) Event-form, which is filled in, if the patient is omitted from the examination because of some kind cause.

(d) Death-form, which is filled in, if the patient dies.

On monitoring of the side-effects see [1].

The registers are realised on small microcomputers:

- eight-bit processor (Z80)
- 2 x 250 Kbyte floppy disk
- 64 Kbyte RAM (no ROM)

Our data management system developed for medical applications on small microcomputers is controlled by a monitor: micro-SHIVA. It is portable to any computer equipped with CP/M opera-

ting system, Z80 processor and 16 Kbyte RAM. The heart of this system is an extended full-screen editor. In more detail see [2].

Data manegement

The data manegement package provides the user program with functions needed in data management, and offers the user the opportunity of activating simple functions by function keys. These latter functions are the following:

- data input and data modification,
- record deletion,
- query by keys,
- display of record contents.

When the user pressed a function key the system waits for confirmation, and executes the command.

Data storage and modification

Database structure is defined by filling up a special form. It describes the record structure (names and attributes of data) and the search keys.

Data record consist of compressed fixed-length and variable-length data. Numeric data are stored as bit strings and are not aligned on byte boundary. Some data fields are assigned as keys. It is not required that a record be unambigously defined by its keys.

Data fields of a form displayed on the screen can be filled up at will, except those of R/O attribute. Names that may have

been assigned to data fields of the form (while editing it) connect the fields to fields in database record. Contents of named fields are copied into the corresponding database record.

Searching

Two basic methods of query are used: one for immediate service of the user watching the screen, the other is intended to be used in batch-like processing. The first mentioned fast retrieval can be formulated in terms of keys allowing search for undefined key values and usage of partially defined keys, too. The key values are given by filling up a form. Searching is established by the use of a key-table that contains keys of each record. The key-table may contain unchanged or coded key values. In the latter case it is not required that the key values be decodable.

The other method is more flexible. In this case a special query form is filled up. Complex queries can be formulated defining conditions that consist of value list or interval list for data items. From these conditions complex formulae can be built up by the use of logical AND, OR and NOT. This method makes use of the key-table, too.

Other programs assure the performance of more complicated works, for example special statistical tables, special lists on patients, to the run of BMDP programs the transition from the microcomputer to IBM-3031 computer,etc.

References

[1] M.Csukás, E.Farkas, A,Krámli, G.Maróti, J.Soltész: Microcomputer monitoring of the side effects in Hungarian pharmacological study, Conference on automation of information processing on personal computer, Budapest,1986.

[2] P.Kerékfy, M.Ruda: Form management by micro-SHIVA, Conference on automation of information processing on personal computer, Budapest, 1986.

Realization of large-sized pharmaco logical effect examination with microcomputer

I. Ratkó et al.

Summary

The authors and their colleagues have been engaged in developing health care systems since 1972. The paper reports on the authors' special works on microcomputers. The system developed by them performs mathematical statistical analysis of the effect of medicines given to patients who had suffered an acute myocardial infarction. The data of approximately 2500 cases are analyzed.

MTA SZTAKI Tanulmányok 202/1987 pp. 181-193

О НЕКОТОРЫХ ПРОБЛЕМАХ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ МИКРОПРОГРАММНЫХ ВЫЧИСЛИТЕЛЬНЫХ СТРУКТУР

Шукурян Ю.Г.

Отличительной чертой технологии проектирования современных средств обработки данных явилось оформление в виде самостоятельного этапа процесса разработки микропрограммно реализуемой доли математического обеспечения, определяющей так называемый микросистемных уровень вычислительной машины. Микропрограммная структура ЭВМ (совокупность реализуемых на базе управляющей памяти алгоритмов и ресурсов, над состояниями которых они определены) оказывает существенное и иногда определяющее влияние на производительность и эксплуатационные характеристики машин. Ряд причин обуславливает особое внимание к микропрограммированию [9]: резкое увеличение сферы применения (микропроцессоры, реализация частей операционных систем и прикладного программного обеспечения ЭВМ, микродиагностика), необходимость развития инструментальных средств микропрограммирования. В данной работе будут рассмотрены три проблемы, связанные с проектированием микропрограммных структур: быстродействие, корректность и распределение управляющей памяти. Программный аспект микропрограммирования, характеризующийся средствами задания микропрограмм (языки микропрограммирования) и способом их реализации, выдвигает задачу оптимизации и "улучшения" алгоритмов относительно выбранного критерия (быстродействие,память).Проблема состоит в отыскании в классе микропрограммы реализующих одно и то же преобразование информационной среды(состояний совокупности ресурсов,на которых определены микрооперации),оптимального алгоритма.Важной проблемой также является обеспечение надежности и корректности микропрограммы путем тестирования или доказательства правильности.Ясно,что алгоритмические трудности здесь неминуемы,

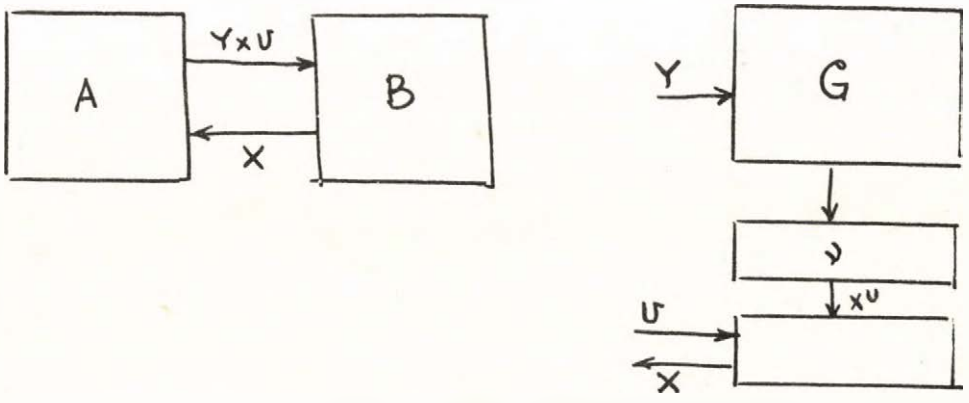
однако исследование разрешимых случаев за счет усиления форм эквивалентности алгоритмов, как показано ниже, позволяют продвинуться в направлении формулировки общих приемов. Рассматривая аппаратный аспект микропрограммирования, характеризующийся структурой микрокоманд (одновременно выполняющимися микрооперациями), необходимо учесть возникшие в настоящее время новые предпосылки использования полупроводниковых запоминающих устройств в логическом проектировании вычислительных устройств: высокая компановка электронных схем в кристалле БИС, регулярность и повторяемость структуры ПЗУ, независимость топологии от записываемой информации [40]. Задача распределения управляющей памяти, рассматриваемая ниже, относится к проблемам, возникающим при логическом проектировании с использованием ПЗУ. В основе наших рассмотрений положен автоматно-алгебраический подход к решению задач автоматизированного проектирования, развитый в [2] .

I. В этой части рассматривается проблема оптимизации микропрограммных алгоритмов по быстродействию в рамках теории дискретных преобразователей [I,5]и формулируется метод оптимизации путем ациклического структурирования.

Инициальный $X-Y$-автомат A Мили (X, Y - входной и выходной алфавиты, A -множество состояний), называется автоматом с заключительным состоянием (а.з.с.), если в A выделено заключительное состояние a^* , отличное от начального a^0 и такое, что $\delta_A(a^*, x)$ не определено для игры (a^*, x), $x \in X$ (δ_A, λ_A - функции переходов и выходов) [I] . Если задано отображение $\rho_A : A \to U$, где U - конечный алфавит условий, то A называется U-отмеченным $X-Y$-автоматом с заключительным состоянием [3] . Зафиксируем информационное множество B , элементы которого являются перерабатываемые алгоритмом объекты, и выделим подмножество B_0 начальных объектов. Интерпретация множеств X, Y, U на B состоит в определении системы отображений $\mu_u : B \to X$ $(u \in U)$ и системы преобразований $f_y : B \to B, (y \in Y)$. Тогда A становится дискретным преобразователем, а B может рассматриваться как автомат с входным алфавитом $U \times Y$, выходным алфавитом $-X$, функцией переходов $\delta_B(b, (y,u)) = f_y(b)$,

функцией выходов $\lambda_B(b,(u,y)) = \mu_B(b,u) = \mu_u(b)$. В частности, интерпретацией является операционный автомат микропрограммного процессора, для которого y и u являются микрооперациями, осуществляющими преобразования внутренних памятей и проверку условий соответственно.

Структурно дискретный преобразователь может быть представлен в виде схемы из взаимодействующих автоматов A и B (рис.Ia).



а) б)

Рис.I. Дискретный преобразователь (а), операционный автомат с варьируемой функцией выходов (б).

Алгоритмический процесс, определяемый указанной моделью, состоит в применении автомата A (запуске с начального состояния) к элементам b из B_0 информационного множества (автомата) B . Таким образом с A связывается функционал , определенный на тех $b \in B$, к которым A применим (оказывается в a^*) и равный состоянию b' , в котором оказывается B при взаимодействии. При фиксированной интерпретации B автоматы A_1 и A_2 эквивалентны тогда и только тогда, когда $f_{A_1(U)} = f_{A_2(U)}$. Проблема эквивалентности дискретных преобразователей исследована в [I] и естественно продвижение получено в случаях усиления приведенной формы эквивалентности. Плодотворным оказывается эквивалентность относительно класса информационных множеств, определяемых классом автоматов с варьируемой функцией выходов. Пусть G - инициальный (с начальным состоянием q_0), всюду определенный Y -автомат. Для

любой функции $\nu \in (X^U)^G$ обозначим $\gamma(G_\nu)$ инициальный $Y \times U$ - X-автомат, полученный по схеме на рис.1б. Если $L \subseteq (X^U)^G$, то пусть $\gamma(G, L)$ обозначает класс всех операционных автоматов вида $\gamma(G_\nu)$, $\nu \in L$. Про класс $\gamma(G, L)$ говорим, что он получен из G варьированием функции выходов в L. Содержательно, в случае рассмотрения операционного автомата (информационного множества) B микропрограммного процессора и определяемого им множества $\mathfrak{B}$ всех инициальных подавтоматов автомата B, автомат G учитывает все те соотношения в полугруппе преобразований B, которые выполняются для всех подавтоматов.

Пусть $\mathfrak{B}$ -класс инициальных $Y \times U - X$ - операционных автоматов и $A(U)$ -отмеченный автомат с заключительным состоянием, который применим к автоматам $\mathfrak{B}_0 \subseteq \mathfrak{B}$.

Абсолютным быстродействием $A(U)$ относительно $\mathfrak{B}$ называется отображение $T_{A(U)}: \mathfrak{B}_0 \to \mathcal{N}$, , где $\mathcal{N}$ -натуральный ряд чисел, определенное равенством $T_{A(U)}(B) = n$, n -количество шагов, выполняемых при применении $A(U)$ к B [3].

Пусть на $\mathfrak{B}$ определена σ -алгебра $W_{\mathfrak{B}}$ и вероятностная мера $\mathcal{P}$ на системе $W_{\mathfrak{B}}$. Среднее быстродействие $ET_{A(U)}$ автомата $A(U)$ относительно вероятностного пространства $(\mathfrak{B}, W, \mathcal{P})$ определяется равенством $ET_{A(U)} = \mathcal{P}^{-1}(\mathfrak{B}) \sum i \mathcal{P}(T_{A(U)}^{-1}(i))$, если ряд, стоящий в правой части, сходится.

Пусть $A_1(U)$ и $A_2(U)$ эквивалентны относительно $\mathfrak{B}$, тогда их можно естественным образом сравнивать по быстродействию, считая A_1 "лучше" A_2, если для любого $B \in \mathfrak{B}_0$, $T_{A_1(U)}(B) \leq T_{A_2(U)}(B)$ $(ET_{A_1(U)} \leq ET_{A_2(U)})$. Задача состоит в построении оптимального автомата в заданном классе эквивалентности $\mathfrak{A}$. Результаты, полученные в этом направлении, имеются в [3,4]. Здесь же будет изложен подход, основанный на общности приемов, используемых для "чистого" решения проблемы оптимизации для конкретных типов операционных автоматов. Общность состоит в том, что для заданного класса $\mathfrak{A}$ эквивалентности определяется подкласс $\mathfrak{A}^0$ автоматов с канонической формой и путем эквивалентных преобразований любой представитель из $\mathfrak{A}$ сводится к представителю из $\mathfrak{A}^0$. При этом $\mathfrak{A}^0$ выбирается таким образом, чтобы он содержал оптимальный автомат, и его построение

можно было осуществить перебором.

Схема метода ациклического структурирования, ориентированного прежде всего на класс микропрограммных алгоритмов с большим количеством длинных ациклических участков, такова. После предварительного анализа или структурирования исходного автомата с заключительным состоянием на ациклические блоки путем расширения памяти операционного автомата осуществляется оптимизация взаимосвязей между блоками (глобальный уровень), затем - оптимизация самых блоков (локальный уровень). Рассмотрим лишь случай, когда $|U|=1$.

Пусть G -вполне определенный Y -автомат, все подавтоматы которого попарно изоморфны. Обозначим $Y^*=\{y^* \mid y\in Y\}$. Пусть S -бесцикловый Y^* -инициальный автомат (для любого состояния s и входных слов q_1, q_2 $(q_1\neq q_2)$ имеет место $sq_1\neq sq_2$. Тогда расширением Y -автомата G с помощью Y^* -автомата S называется $Y\cup Y^*$-автомат G^* такой, что $G^*=G\times S$ $\delta_{G^*}((g,s),z)$ равно (gy,s) , если $z=y$ и (g_0,sy^*), если $z=y^*$; существует отображение $\gamma: S\to G$, такое, что $\gamma(sy^*)=\gamma(s)y$.

Каждому $X-Y\cup Y^*$-автомату A^* с заключительным состоянием сопоставим $X-Y$ -автомат $\eta(A^*)$, полагая $\eta(A^*)=A^*$, $\delta_{\eta(A^*)}(a,x)=\delta_{A^*}(a,x)$, $\lambda_{\eta(A^*)}(a,x)=y$, если $\lambda_{A^*}(a,x)\in\{y,y^*\}$, где $a\in A^*$.

Обозначим $[t,a,(g,s),z]$ ситуацию, когда в момент t при взаимодействии G^*_μ и A^* оказывается в состояниях (g,s) и a соответственно и A^* выдает $z\in Y^*$. Тогда $A_1^*\equiv A_2^*(G^*)\Leftrightarrow A_1^*\sim A_2^*(G^*)$ и из $[t',a',(g',s'),z']$ и $[t'',a'',(g'',s''),z'']$ $(a'\in A_1^*,\ a''\in A_2^*)$, если $s'=s''$, то $g'=g''$.

Легко показать, пользуясь существованием для каждого $Y\cup Y^*$-автомата $G^*_\mu, \mu\in X^G$ гомоморфизма на некоторый Y-автомат G_ν $\nu\in X^G$, что из $\equiv$ относительно G^* следует эквивалентность относительно G и если A_1 "лучше" A_2 по быстродействию относительно G^* , то это же отношение сохраняется между $\eta(A_1^*)$ и $\eta(A_2^*)$ относительно G .

Для фиксированный g $(g\in G)$ и целого $k>0$ обозначим $G^{(k)}(g)=\{g'\in G \mid g'=gp,\ p\in F_Y,\ |p|\le k\}$ где F_Y - полугруппа слов в алфавите Y . Образуем конечный Y -автомат $F^{(k)}(g)$, взяв в качестве состояний элементы множества $G^{(k)}(g)$

$\cup\{h\}$ и определив функцию переходов $\delta_{F^{(\kappa)}(g)}(g',y) = g'y$, если $g'y \in G^{(\kappa)}(g)$ и $\delta_{F^{(\kappa)}(g)}(g',y) = h$ в остальных случаях. Поскольку Y -автоматы $F^{(\kappa)}(g)$ изоморфны в силу изоморфности подавтоматов G , то введем для них единственное обозначение G_κ . Расширение Y -автомата G_κ с помощью бесциклового Y^* -автомата S обозначим G^*_κ и назовем k -приближением G^* .

Пусть A есть X-Y-автомат с заключительным состоянием, который посредством множества $\hat{A}$ "разделяющих" состояний структурирован на ациклические блоки [4] . Построим X-$Y\cup Y^*$ -автомат A^*, полагая $A^* = A$, $\delta_A = \delta_{A^*}$, $\lambda_A = \lambda_{A^*}$, если $\delta_A(a,x) \notin \hat{A}$ и $\lambda_{A^*}(a,x) = (\lambda_A(a,x))^*$, если $\delta_A(a,x) \in \hat{A}$, $a \in A$, $x \in X$. Пусть $k(\alpha)$ - максимум длин путей в ациклическом блоке α, $k = \max k(\alpha)$ по всем блокам α . С использованием существования гомоморфизма, о котором говорилось выше, можно показать, что для X-Y -автомата A и k -приближения G^*_κ справедливо утверждение: если A^* эквивалентно A_1^* относительно G^*_κ и A^* "лучше" A_1^* относительно G^*_κ , то $\eta(A)$эквивалентен $\eta(A_1)$ и $\eta(A^*)$ "лучше" $\eta(A_1^*)$ относительно G .

С учетом приведенных соображений, общая стратегия оптимизации формулируется следующим образом. Исходный X-Y -автомат структурируется на ациклические участки. Путем анализа операторов y^* строится расширение автомата G с помощью бесциклового автомата S . Выбор S важен. Простейший случай имеет место, когда в качестве S берется свободный Y^*- автомат, т.е. происходит игнорирование наличия каких-либо нетривиальных соотношений между преобразованиями множества G .Поэтому целесообразно выбрать S таким, чтобы сохранить с помощью отображения γ (см.определение расширения) "бесцикловые" соотношения между преобразованиями G . После структурирования каждый ациклический блок описывается соответствующей таблицей решений [4], затем строится так называемая огрубленная стратегия, которая представляет из себя автомат, состояниями которого являются таблицы решений, а переходы осуществляются от блока к блоку. Огрубленная стратегия выда т операторы из множества Y^* на бесцикловую компоненту S расширения G^*_κ , получая на свой вход набор отметок ε из $X^{G^*_\kappa}$. Известно [I],

что путем применения преобразований устранения ветвлений, устранения слияний, устранения единицы и сокращения линейных участков можно построить наилучшую по времени конечную огрубленную стратегию. Далее по таблицам решений строятся оптимальные реализации для блоков и подставляются в оптимизированную огрубленную стратегию.

Указанный метод, как показано для случая автомата G, разлагающегося в прямое произведение бесциклового автомата, все подавтоматы которого попарно изоморфны, и конечного автомата дает точное решение задачи оптимизации [3,4].

2. Ошибки, вкравшиеся в микропрограммы, могут привести к разрушительным последствиям, поэтому проблемы обеспечения корректности микропрограмм безусловно является актуальной. Естественными путями решения этой проблемы является применение традиционных для программирования приемов - доказательство правильности и отладка на полной системе тестовых примеров. Приметы доказательства частичной корректности микропрограмм с использованием метода Флойда приведены в [5]. Здесь же приведены некоторые результаты по тестированию, полученные под руководством автора.

Достаточным на практике является использование набора тестовых примеров, на которых активизируются все в принципе активизируемые линейные участки тестируемого алгоритма. Такая система примеров называется полной системой. Следуя [8], приведем некоторые определения. Элемент β информационного множества B называется допустимым, если A применим к β. Пусть $\beta_{a^0,a^*}(\beta)$, активизируемый в дискретном преобразователе A при его применении к β называется реализуемым. Пусть $\mathfrak{D}(A)$ - множество реализуемых путей. Переход $a \xrightarrow{x} a'$ в дискретном преобразователе A называется реализуемым, если он принадлежит реализуемому пути. Конечное множество β^* примеров называется полной системой примеров, если любой реализуемый переход принадлежит пути, активизируемому некоторым примером $\beta \in B^*$. Существенные результаты, связанные с исследованием алгоритмической проблемы построения полной системы примеров, получены в [8]. В частности, рассмотрен дискретный

преобразователь, информационное множество которого определяется состояниями конечного множества X входных лент, Y выходных лент, конечного множества V внутренних ячеек и счетсичов Z. Операторами (преобразованиями) являются элементы системы K_1, содержащие пересылки между внутренними ячейками, считывание (запись) из (на) входных (выходные) ленты во (из) внутренние ячейки (внутренних ячеек и счетчиков), засылка констант в счетчики и внутренние ячейки, увеличение счетчиков на единицу, сравнения: v_c v' v и z с константой, тождественный оператор и оператор останова. Показано, что существует алгоритм построения полной системы примеров для любого дискретного преобразователя в системе команд K_1 с нулевыми начальными значениями счетчиков. В [6] доказано, что проблемы построения полной системы примеров разрешима и в случае добавления к K_1 операторов считывания из входных лент в счетчики. Исследованы программы с групповыми операторами присваивания с задержками в системе команд M. Эти операторы имеют вид y_1,t_1 ; y_2,t_2;... y_k,t_k. , где y_j - команды из M, t_j - целые неотрицательные числа $(j=1,\dots,k)$. Выполнение всех команд $y_1,\dots,y_k$, входящих в групповой оператор присваивания с задержками, начинается одновременно. Число t_j (задержка команды y_j) указывает число тактов, необходимое для завершения y_j (тактом является время выборки одного безусловного оператора программы). Программа в системе M -ориентированный граф, каждая вершина которого отмечена условной командой из системы M или групповым оператором присваивания с задержками. Если с M связано подмножество команд присваивания , одновременное выполнение которых приводит к аварийному завершению (конфликту в программе), то конечная система Σ примеров называется конфликтно полной, если конфликт при выполнении программы возможен тогда и только тогда, когда в Σ существует пример, приводящий к конфликту. Доказано, что если в M имеются команды пересылки, и для программ в M проблема построения полной системы примеров разрешима, тогда для программ в M с групповыми операторами присваивания с задержками разрешима проблема построения конфликтно полной системы примеров. Более того, если в M имеются пе-

ресылки, то для любой программы с операторами задержки существует эквивалентная программа с нулевыми задержками.

3. В этой части вводится модель микропрограммируемого автомата и приводится эвристическое решение задачи распределения памяти ПЗУ, реализующего этот автомат [7].

Пусть X, Y, Z - конечные алфавиты, M -конечное линейно упорядоченное множество адресов, называемое управляющей памятью. Память характеризуется длиной n микрокоманды ω, принадлежащей множеству $\Omega = Z^n \cup \{\emptyset\}$. По каждому адресу M записывается некоторая микрокоманда из Ω, совокупность всех микрокоманд, записанных в M, определяется отображением $\pi : M \to \Omega$, называемым микрокодом.

(n, Z, M) - микропрограммируемый X-Y -автомат определяется парой отображений:

$$\varphi : \Omega \times M \times X \to M$$

$$\lambda : \Omega \times M \times X \to Y$$

X-Y -автомат с микрокодом π есть X-Y -автомат с множеством M состояний и функциями φ_π и λ_π переходов и выходов:

$$\varphi_\pi : M \times X \to M \quad , \quad \varphi_\pi(m, x) = \varphi(\pi(m), m, x)$$

$$\lambda_\pi : M \times X \to Y \quad , \quad \lambda_\pi(m, x) = \lambda(\pi(m), m, x)$$

Структурная схема этого автомата приведена на рис. 2а

Пусть A - есть X-Y -конечный автомат Мили и задан (n, Z, M) -микропрограммируемый X-Y -автомат. Отображение $\varepsilon : A \to M$ назовем реализацией автомата A с помощью (n, Z, M) -микропрограммируемого автомата, если существует X-Y -автомат с программой π такой, что $\varepsilon(a, x) = \varphi_\pi(\varepsilon(a), x)$, $\lambda_A(a, x) = \lambda_\pi(\varepsilon(a), x)$, $x \in X$, $a \in A$.

Ниже исследуется класс микропрограммируемых автоматов ее ступенчатой (сегментированной) организацией памяти, распространенный в микропрограммных структурах ЭВМ [2].

Память $M = \{\alpha_{ij} \mid i = 1, \ldots, s;\ j = 1, 2, \ldots, 2^k\}$ разбита на последовательные сегменты $M_i = \{\alpha_{ij} \mid j = 1, \ldots, 2^k\}$, $i \in P = \{1, \ldots, s\}$ (линейный порядок $\alpha_{ij} < \alpha_{i'j'}$ определен условием $i < i'$ или $i = i'$ и $j < j'$). Считается фиксированным конечное множество условий U, $|U| = k$.

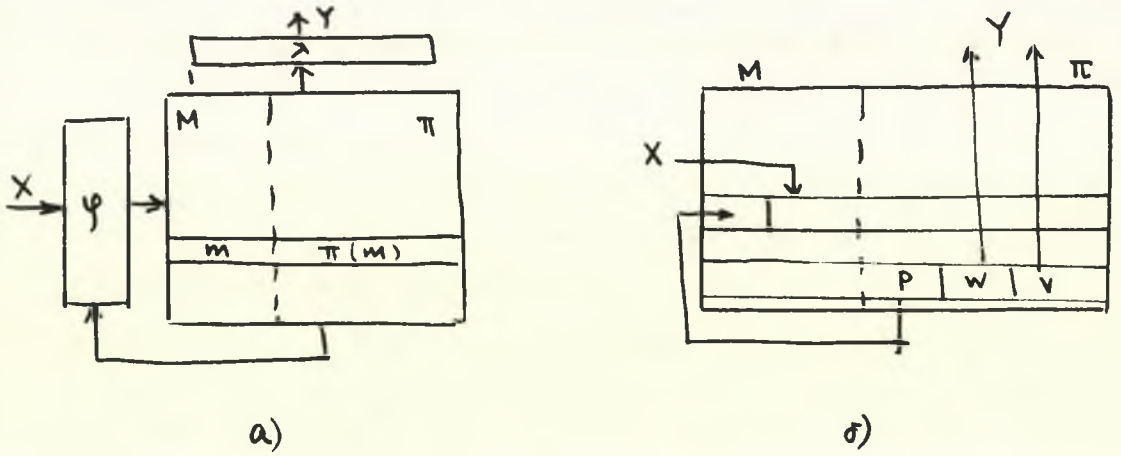


Рис.2

Пусть $Y=2^U$. В рассматриваемом случае $\Omega = P \times W \times V$, где W -конечное множество микроопераций-преобразователей, V - множество микроопераций, осуществляющих проверку значений условий из некоторого подмножества τ множества U. Алфавиты X и Y имеют специальную структуру: в качестве X берутся все двоичные наборы длины, не превышающей $|U|=k$, а в качестве Y - множество $W \times V$.

Адрес очередной микрокоманды определяется заданием указателя $p \in P$ сегмента и значением входного сигнала x, вырабатываемого в зависимости от условий $\tau \in V$.

Пусть $\tau \in V$, тогда считается, что задано разбиение ρ_τ множества $S=\{1, \ldots, 2^K\}$ на классы эквивалентности, инвариантное относительно указателя сегмента. При этом выполняются условия: количество элементов в любом классе разбиения ρ_τ равно $2^{|\tau|}$ и $\rho_{\tau \cap \tau'} = \rho_\tau \cap \rho_{\tau'}$, $\rho_{\tau \cap \tau'} = \rho_\tau * \rho_{\tau'}$, где $\cap, *$ - операции пересечения и сложения разбиений.

Во введенных обозначениях функции φ и λ для (n, z, M)-микропрограммируемого автомата определяются следующим образом. Пусть $\pi : M \to \Omega$ - микрокод и $\pi(\alpha_{ij}) = (p, w, \tau)$ (рис. 2б), тогда $\varphi(\pi(\alpha_{ij}), \alpha_{ij}, x)$ определено, если $|x|=|\tau|$ и из $\varphi(\pi(\alpha_{ij}), \alpha_{ij}, x_1) = \alpha_{i'j'}$, $\varphi(\pi(\alpha_{ij}), \alpha_{ij}, x_2) = \alpha_{i''j''}$ $(x_1 \neq x_2)$ следует $i'=i''=p$, $j', j'' \in \rho_\tau$, $j' \neq j''$. Содержательный смысл разбиения ρ_τ состоит в том, что если в микрокоде π с адресом α_{ij} имеется указатель на сегмент p, $p \in P$ и проверяется подмножество условий τ, то ад-

реса переходов, формируемые под действием значений x проверяемых условий y ,принадлежат одному и тому же классу эквивалентности ρ_y .

Важной характеристикой реализации ε X-Y-автомата A с помощью (n, k, m) -микропрограммируемого автомата является занятость $\Delta_A(\varepsilon)$, определяемая следующим образом:

$$\Delta_A(\varepsilon) = |\{p \in P \mid M_p \cap \varepsilon(A) \neq \varnothing\}|$$

Задача нахождения оптимальной реализации состоит в построении такого ε_0 , что $\Delta_A(\varepsilon_0) = \min_{\varepsilon} \Delta_A(\varepsilon)$

В [7] показано, что эта задача для X-Y-автомата A сводится к задаче нахождения минимального числа внутренне устойчивых множеств специального вида во взвешенном графе, связанной с A . Эвристически алгоритм решения этой задачи был запрограммирован и для управляющей памяти в 8 кслов с пятью уровнями сегментации дал практически оптимальную реализацию.

Литература

I. Глушков В.М., Летичевский А.А., Теория дискретных преобразователей. Избранные вопросы алгебры и логики. Новосибирск, СО, "Наука", 1971.

2. Глушков В.М., Капитонова Ю.В., Летичевский А.А. Теоретические основы проектирования дискретных систем. Кибернетика, № 6, 1977.

3. Шукурян Ю.Г., Тер-Акопов А.К. Табличные программы и дискретные преобразователи I, Кибернетика № 3, 1978.

4. Шукурян Ю.Г. Табличные программы и дискретные преобразователи II, Кибернетика № 6, 1978.

5. Шукурян Б.Г., Тадевосян А.Г. О доказательстве корректности микропрограмм. Вопросы проектирования ЭВМ, Киев,1974.

6. Тадевосян А.Г. О некоторых разрешимых случаях проблемы построения полной системы примеров. Доклады АН Арм.ССР, 1983, т.72, № 2.

7. Петросян А.В., Маркосян С.Е., Шукурян Ю.Г. Математические вопросы автоматизации проектирования ЭВМ, Ереван, АН Арм.ССР, 1977.

8. Барздинь Я.М., Бичевский Я.Я., Калниньш А.А. Построение полной системы примеров для проверки корректности программ. Теория алгоритмов и программ. Рига, Латв.ун-т, 1974.

9. С.Хассои. Микропрограммное управление, I,: М., 1974.

10. С.Мурога Системное проектирование сверхбольших интегральных схем, П, М., 1985.

On some problems of automatized planning of computing structures

Ju. G. Šukurjan

Summary

In the paper the problems of automatized planning (projecting) of computers for wide use are discussed. A theoretico-automated approach of projecting computers which are optimal with regard to the speed and the correctness of micro-programme structures are studied in detail.

MTA SZTAKI Tanulmányok 202/1987 pp. 195-208

AN EXPERT SYSTEM FOR TEST DESIGN OF DIGITAL CIRCUITS

József SZIRAY, Zsolt NAGY, Levente L. MÁTÉ *

Computer Research and Innovation Center,
P. O. Box 19, 1251 Budapest, Hungary

* Computer and Automation Institute,
P. O. Box 63, 1502 Budapest, Hungary

Abstract

A general concept is presented for the structure and functions of an expert system in the field of computer aided test design of digital circuits. The expert system, called **DIEX**, has to cooperate with a traditional test design program, where the following goals are considered: **DIEX** is meant above all for globally controlling the test design program, by utilizing the expertise of a human operator. On the other hand, it has to incorporate human knowledge as well that has been gained in the field of manual test design.

The paper briefly outlines the test calculation procedures applied in the test design program, then it gives an overview of **DIEX**, where the basic subsystems, the control strategies, and the data bases will be shown.

Keywords: Computer aided design, automatic test design, digital circuits, expert system.

1. Introduction and general features

The rapid advances in the digital integrated circuit technology, the steady growth of the complexity of LSI-VLSI chips have imposed severe demands on designing the test procedures of these circuits. In order to meet these demands, we need to utilize efficient test calculation and fault simulation methods which are implemented on computer. In this solution a test design program receives the logic diagram of the circuit, and calculates an input test sequence that is capable of detecting a predefined set of circuit faults. Such a test generation program can easily be fitted in a complex design automation system, where logic simulation and layout design are also carried out for the same digital circuit.

During the past 15 years numerous heuristic and algorithmic methods have been developed for the calculation of fault detecting and diagnostic tests of digital circiuts [1]-[2]. These methods proved to be computationally feasible and efficient for circuits of medium-scale integration (MSI) range. However, the technological evolution of microelectronics has brought a permanent need for new approaches of designing test and diagnostic procedures.

Generally, a computer program for test calculation makes use of more than one calculation procedures. Experience shows, however, that it is very difficult to select the most appropriate procedure at a given phase of the test design process. The difficulty can be overcome by the intensive use of interactivity where an expert is involved in the process with the task of making the necessary decisions. Here the expert analyses the process and taking into consideration the actual fault detection rate (fault coverage) decides over the procedure to be selected in the next step. Also, the expert controls the selected procedures by supplying the appropriate parameter values for them.

In general, it can be shown that the decision making expert (operator) plays a key role in an interactive system in terms of efficiency and in fault coverage achieved.

Our primary goal is the emulation of the above process as completely as possible by applying the basic techniques of expert systems [3]-[8]. However, it should be emphasized that we consider the existing test calculation procedures to be further of key importance, whereas the selection among them and the scaling of the selected procedure are entrusted to an expert system.

As a secondary goal, the expert sytem has to incorporate human knowledge and methodology as well, gained in the field of manual test design. This kind of knowledge can also exploit the computational power of a test generation program, especially in terms of fault simulation.

This paper briefly outlines the test calculation procedures applied in the test design program, then it gives an overview of the expert system, where the basic subsystems, the control strategies and the data bases will be shown.

2. The test design program

Our existing test design program is called **DIAS** (**DI**agnostic **A**id **S**ystem). The program system **DIAS** serves for designing the fault detection/diagnosis procedures of digital LSI and VLSI circuits, especially of gate-array type integrated circuits. The aim of the computations is to automatically generate a test set that is capable of detecting all predetermined faults in the circuit.

Fault model

DIAS is able to generate tests for the following set of circuit faults:

- Single and multiple stuck-at 0/1 faults at the input and output points of each circuit module.
- Single bridging (short-circuit) faults between two circuit nodes.
- Single stuck-at high/low impedance faults at the output of tri-state elements.

It should be added that most of the open-circuit faults are equivalent with the logic stuck-at faults, thus they are implicitly included in the fault model.

Circuit modeling

DIAS handles both combinational and sequential logic circuits. The logic building elements (modules) of a circuit are modeled at a functional level. It means that the logic values of a module are evaluated with knowledge of its functional behavior. This approach enables to include new logic elements in the program system, according to the requirements.

Fault simulation

DIAS includes a fault simulator which simulates the behavior of the fault-free circuit and each possible faulty circuit. Fault simulation is performed in order to determine all the faults that are detected by the individual test patterns. This information is utilized

- for calculating the test responses,
- for controlling the test calculation process,
- for evaluating the fault coverage of the entire test set,
- for diagnostic purposes during the concrete measurement process.

The fault simulation technique used in DIAS is the so-called **concurrent approach** [1]. This approach is very efficient and is ideally suited to functional level modeling.

Test calculation

In order to utilize the computer resources at best, it has been developed three automatic test calculation methods which are implemented in DIAS [2], [9]-[10]:

- **Random and derived tests:**

  The tests are generated in a random way, where fault simulation is used for deciding whether a test is accepted or rejected. Also, there are useful tests that are drived heuristically from a random one.

- **Adaptive random approach:**

  The essence of this approach is that the probabilistic properties of a random signal source are controlled by the actual circuit behavior. Here the behavior is monitored by means of logic simulation.

- **Deterministic test calculation:**

  The tests are calculated in a deterministic way by applying the so-called **composite justification algorithm** [2], [9]-[10]. The algorithm is based on calculations performed simultaneously in the faulty and fault-free circuit versions. The calculations utilize a multi-valued logic algebra. This deterministic approach has been extended to the whole fault model in **DIAS**.

Manual tests

Manually designed test sequences are also processed and simulated by **DIAS**. The description of these sequences is facilitated by a high level, easy-to-use test language called **TESTART** [11].

Operational environments

The program system **DIAS** has been written in FORTRAN-77, and runs on VAX-11/730-780 or MicroVAX II computers, under VMS operating system.

3. System structure

As outlined previously, the entire test design system will consist of a traditional test calculation-fault simulation program, as well as an expert system for controlling and exploiting the former one. At present the expert system, which is called **DIEX**, is under development.

The overall structure and communication links of the complete **DIEX-DIAS** system are shown in Figure 1.

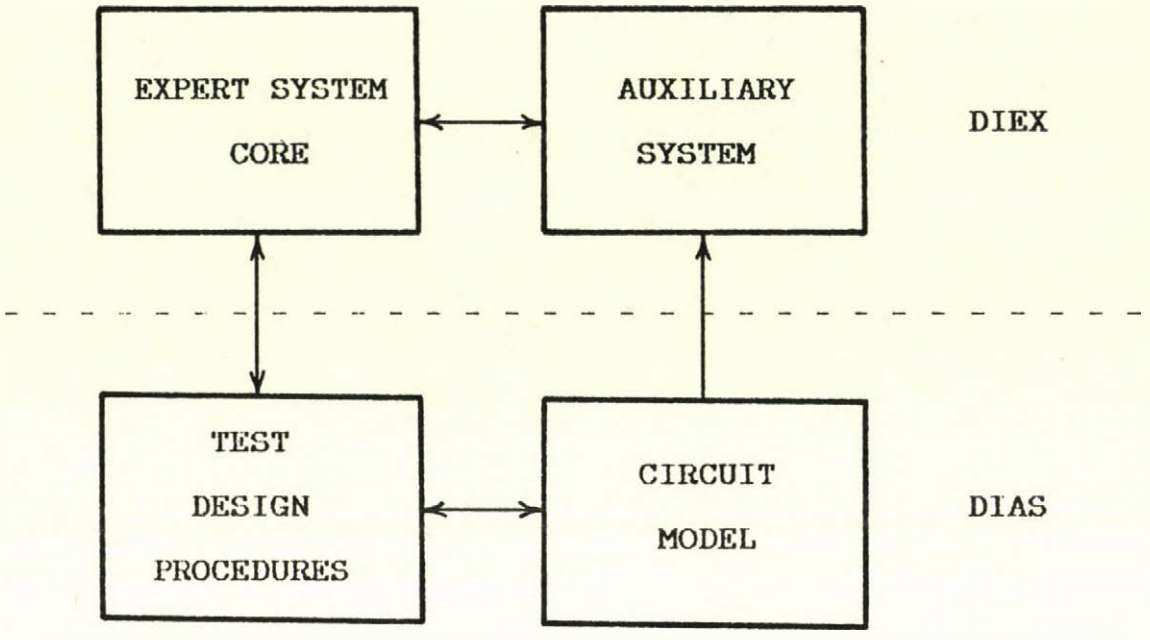


Figure 1. The overall system structure.

As we have seen, **the test design procedures** serve for performing all the algorithmic computations that result in the required tests. Also, the **TESTART** compiler for the manually designed tests, as well as the fault simulator belong to this group.

The circuit model contains the data for representing the logic diagram of the actual circuit: it describes the different circuit modules and the connections between them. The functional (behavioral) description of the modules is also included in the model.

The expert system core is that part of the **DIEX-DIAS** system which is provided with artificial intelligence. It monitors and controls the whole system, where knowledge representation, inference mechanism and human interface are all realized within this block.

The auxiliary system comprises various programs that generate auxiliary data for the expert system core. Part of these data are permanent, whereas most of them vary depending upon the actual demands during the test design process.

In the following the two building blocks of **DIEX** will be discussed in a more detailed way.

4. The expert system core

The expert system core is hierarchically organized, where the main component is the so-called "Strategist Program". The Strategist controls the whole system, makes the most important decisions, while cooperating with a group of expert subsystems. The programs belonging to this subexpert group are at the same hierarchical level, each being in a subordinate relation with the Strategist. The group has the following members:

- Initializer.
- Random Controller.
- Adaptive Random Controller.
- Deterministic Controller.
- Test Master.
- Aid Getter.
- Librarian.
- Adviser.

It is important to note that each component within the core, including the Strategist too, has its own separate knowledge base and inference mechanism. As it can be seen, this organization represents a distributed way of processing.

Next the individual components will be described one by one.

Strategist

The Strategist is the main controller of the whole system. It has the right to activate any member of the subexpert group at any phase of the test design process, with the following opportunities:

- After analysing the actual situation, it decides which subexpert is to be selected to take over the control of test calculation.

- It prescribes the limitation conditions for an expert, i.e. the conditions for leaving off the computations.

- It has the opportunity any time to turn to the user for help. However, it is also able to go on with its job without getting the help.

- The Strategist is not authorized to supervise the decisions made by the user, however, it can always make proposals to alter such decisions. The same applies to the subexperts, with respect to their relation to the Strategist.

- The Strategist can also consult its special expert (namely, the Adviser) on solving a problem.

It should be added that a decision is always made under uncertainty of a varying degree. When making a decision the Strategist takes into consideration the following main viewpoints:

- The achieved fault coverage.

- The consumed CPU time.

- The structural, topological and control features of the circuit.

- The subsystems that have already been thrown in, and also the outcome of their activity.

- The estimated degree of the momentary uncertainty.

- The number of unknown storage-state values in the circuit.

- Recorded experiences and results of previous program runs.

As far as the knowledge representation is concerned, the Strategist is provided with a rule base. This rule base is structured in that the rules are divided into different groups. The inference mechanism operates on this rule base by applying backward chaining in the evaluation process [7]. It should be added that the same organization principle holds true in respect of the other experts in **DIEX**. This kind of knowledge representation will be based on the use of the Prolog language [5], [7]-[8].

Initializer

The function of this subsystem is

- to bring a sequential circuit to a known storage state, by applying clear/preset signals or by generating an initializing sequence [1],
- to synchronize the clock signals,
- to reach a prescribed storage state by generating a homing sequence [1].

Random Controller

The subsystem is for controlling the random test generation component of **DIAS**. It is done by setting the parameter values which determine the length of unsuccessful random test sequences. The Random Controller is capable of activating more than one random cycles with different parameters, depending on the actual situation, CPU time and the program performance.

Adaptive Random Controller

The subsystem supervises the adaptive random test generator of **DIAS**. Here the task is to carry out experiments in order to determine the probabilistic data for the random signal source, and then perform test generation. This is a cyclical process controlled by various limitation parameters.

Deterministic Controller

This subsystem deals with the deterministic test calculation program of **DIAS**. Its task consists of selecting the next fault for test calculation, designating an appropriate output for fault propagation, as well as setting the limitation parameters for the composite justification algorithm.

Test Master

The Test Master is meant for utilizing the human expertise accumulated in the area of manual test design. It makes use of topological knowledge, generates regular test patterns for the regular circuit parts (e.g., memory, bus, counter), handles the control and data inputs separately, moreover it applies various useful tricks taken from the practice.

Aid Getter

The function of this component is to ask for user's assistance in a "gloomy situation". When doing it, the Aid Getter informs the user about all the relevant data characterizing the situation and the nature of the assistence (e.g., strategic problem, manual tests are needed). The Aid Getter also gives prompt information for the user about the results of his interaction.

Librarian

The Librarian collects the characteristic structural and topological data related to the circuit, then it generates a concise identifier code from these data. It also registers all the relevant activities and results during the test design process, and establishes a library file for storing them. This file of "case story" is maintained for the purpose of utilizing its content in subsequent program runs with other similar circuits. It can be seen that such an archivation serves for a certain self-learning in **DIEX**.

Adviser

The Adviser's job involves

- analysing the case-story file selected by the Librarian, and searching for results that present a basis for a useful decision in the actual situation, and then
- giving advice to the Strategist as to which measure is to be taken in the next step (e.g., the procedure to be activated and its parameter values).

5. The auxiliary system

The auxiliary system comprises traditional programs that are meant for providing the expert system core with information required to decision making. This information can be devided into three partitions (data bases) which are as follows.

Topological data base

It contains data related to the structural and topological characteristics of the circuit. They are partly permanent data for each circuit, and partly dynamic data depending upon the concrete requirements.

Some permanent data are:

- Number of inputs, outputs, bus lines.
- Number of equivalent gates.
- Number of storage bits and feedbacks.
- Combinational depth.
- Sequential depth.
- Distinguished inputs (e.g., control, data).
- Structural-topological information provided by the user.

Some dynamic data are:

- The storage elements/outputs that are reached from a circuit node.
- The storage elements/outputs from which a circuit node can be reached.
- Description of parallel lines belonging to a bus.
- Site of RAM/ROM elements, counters, shifters, decoders, multiplexers, etc.

State-description data base

It represents information pertaining to the logic values within the circuit. These data are always varying during the test calculation. For example: list of storage elements with unknown state value, list of nodes which have not had yet logic 0 or logic 1 on them.

Data base for the results achieved

It summarizes information relating to the faults in test design, all the controlling parameters, as well as the results achieved by the various expert subsystems. Such data are the set of detected, undetected and conditionally detected faults, CPU-time consumption, the faults detected by the individual subsystems, etc.

REFERENCES

[1] M. A. Breuer, A. D. Friedman, "Diagnosis and Reliable Design of Digital Systems", Computer Science Press, USA, 1976.

[2] J. Sziray, "Test calculation for logic networks by composite justification", Digital Processes, Vol. 5, No. 1-2, pp. 3-15, 1979.

[3] V. Begg, "Developing Expert CAD Systems", Unipub, USA, 1984.

[4] R. O. Duda, "Applications of expert systems", Proceedings of the IEEE 3rd Automatic Test Program Generation Workshop, pp. 111-125, San Francisco, March 1983.

[5] G. Cabodi, P. Camurati, P. Prinetto, "The use of Prolog for executable specification and verification of easily testable designs", Proceedings of the 16th Fault Tolerant Computing Symposium, pp. 390-395, Vienna, July 1986.

[6] J. P. Laurent, "Control structures in expert systems", Technology and Science of Informatics, Vol. 3, No. 3, pp. 147-162, 1984.

[7] D. A. Waterman, "A Guide to Expert Systems", Addison-Wesley Publishing Company, USA, 1986.

[8] P. Jackson, "Introduction to Expert Systems", Addison-Wesley Publishing Company, USA, 1986.

[9] J. Sziray, "Functional level test calculation and fault simulation for logic networks", Discrete Simulation and Related Fields (Edited by A. Jávor), pp. 223-234, North Holland Publishing Company, Amsterdam, 1982.

[10] J. Sziray, L. Koczkás, T. Kerekes, "Algorithms of a Test Design Program System for Logic Networks at the Functional Level", SzKI Study (in Hungarian), Budapest, November 1982.

[11] J. Sziray, Zs. Nagy, "TESTART: A test-description language for digital circuits", 10th IMEKO World Congress, Prague, April 1985.

MTA SZTAKI Tanulmányok 202/1987 pp. 209-226

The measure of covering the space by a set A and the projection successive minima of A-A.

B. Uhrin
Computer and Automation Institute
Hungarian Academy of Sciences
1502 Budapest, P.f. 63, Hungary

1. Introduction

Let $\Lambda \subset R^n$ be a point lattice (discrete subgroup of full dimension) and $\mathcal{B}$ be the set of all bases of Λ. Let V denote the Lebesgue measure in R^n and $|H|$ be the cardinality of the finite set H. Denote by A+B the algebraic (or Minkowski) sum of the sets $A,B \subset R^n$, in particular $\delta A := A-A := A+(-A)$. Given a measurable set $A \subset R^n$ we say that R^n is covered by lattice translates of A (shortly R^n is covered by A), if

$$V(R^n \setminus \bigcup_{u \in \Lambda} (A+u)) = 0. \tag{1.1}$$

The condition (1.1) is equivalent to

$$V(P \setminus \bigcup_{u \in \Lambda} ((A+u) \cap P)) = 0 \tag{1.2}$$

where P is a unit cell of Λ, $P := \{ x \in R^n : x = \sum_{i=1}^{n} \lambda_i b_i ,$

$0 \leq \lambda_i < 1,\ i=1,\dots,n\}$, $B := (b_1,\dots,b_n) \in \mathcal{B}$.

By definition we call the number

$$\alpha(A) := V\left(\bigcup_{u \in \Lambda} ((A+u) \cap P) \right) \tag{1.3}$$

Research partially supported by Hungarian National Foundation for Scientific Research, Grant No 1066

the measure of covering R^n by A.
One can see easily that $\alpha(A)$ is independent on the choice of the basis $B \in \mathcal{B}$, i.e. it depends only on Λ.
Ir is clear that

$$\alpha(A) \leq d\Lambda = V(P). \tag{1.4}$$

A more flexible upper bound for $\alpha(A)$ has been proved in [1].

As to lower bounds for $\alpha(A)$, one can see that

$$\alpha(A) = V(\varphi(A)) \tag{1.5}$$

where $\varphi(A)$ is defined as $\varphi(A) := \bigcup_{x \in A} \varphi(x)$,

$$\varphi(x) := \sum_{i=1}^{n} \{\xi_i\} b_i \quad \text{if} \quad x = \sum_{i=1}^{n} \xi_i b_i \tag{1.6}$$

and $\{\xi_i\}$ is the fractional part of ξ_i i.e. the number $0 \leq \{\xi_i\} < 1$ such that $\xi_i - \{\xi_i\}$ is integer.

The φ is usually called the canonical map of R^n into P. It depends on the choosen basis, but the volume $V(\varphi(A))$ is again independent on B.

In [2] we have proved (see also [3],[4])

$$|\mathcal{O}A \cap \Lambda| \geq 2 \frac{V(A)}{V(\varphi(A))} - 1, \tag{1.7}$$

that gives via (1.5) a lower bound for $\alpha(A)$. Using (1.5), two more lower bounds for $\alpha(A)$ can be found in the literature. Namely, let $K \subset R^n$ be a bounded convex body. Then

$$V(\varphi(\nu_n K)) \geq V(K) \cdot \prod_{i=1}^{n} \nu_i \tag{1.8}$$

where ν_i are successive minima of $\mathcal{O}K$ (see [5], p.54).

A refinement of (1.8) has been proved for symmetric convex bodies K (i.e. when K=-K) ([6],p.215):

$$(1.9)\qquad V(\varphi(tK)) \begin{cases} = t^n V(K) & \text{if } 0 \le t \le \lambda_1/2 \\ \ge t^{n-k} V(K) \prod_{i=1}^{k} (\lambda_i/2) & \text{if } \lambda_k/2 \le t \le \lambda_{k+1}/2 \\ & k = 1,2,\cdots,n, \end{cases}$$

where λ_i are successive minima of K.
The aim of this paper is, after introducing the so called projection successive minima, to prove two more inequalities of type (1.9), not assuming the symmetry of K. While (1.7) is a refinement and extension of Minkowski convex body theorem, the inequlaity (1.9) is a refinement of the Minkowski second theorem on successive minima. Similarly, our two new inequalities give two new type successive minima theorems.

2. A basic inequality

Let us take a basis $B:=(b_1,b_2,\cdots,b_n)\in\mathcal{B}$ and $I\subseteq J:=(1,2,\cdots,n)$ with $|I|=k$. The B and I give a decomposition of R^n into two subspaces $T\oplus S=R^n$, where T:= linear hull $(b_i, i\in I)$, S:= linear hull $(b_i, i\notin I)$. The parts of B defining T and S, respectively, then define point lattices $L\subset T$ and $M\subset S$. Dentoe by $V_1, Q, \varphi_1, \alpha_1$ and $V_2, W, \varphi_2, \alpha_2$ the volume, unit cell, canonical map and the measure of covering in T and S, respectively. Let π_1 and π_2 be projections of R^n into T(along S) and into S(along T), respectively. The set M+z is a point lattice in $S+z$, $z\in T$, when we take $S+z$ as the space R^{n-k} with z as the origin. The unit cell in S+z is W+z and we shall use φ_2 and α_2

for denoting the canonical map and the measure of covering in S+z, although these were defined in S. Strictly speaking, the canonical map of a set $B \subset S+z$ is equal to $\varphi_2(\pi_2(B)) + z$ but we shall denote it shortly $\varphi_2(B)$.

It is well known that for L-measurable $A \subset R^n$ the set $A \cap (S+z)$ is L-measurable in S+z for almost all $z \in T$, so the function $V_2(A \cap (S+z))$ is meaningful only for almost all $z \in T$. By definition we put $V_2(A \cap (S+z)) = 0$ for any $z \in T$ where $A \cap (S+z)$ is not measurable. The projection $\pi_1(A)$ is defined as $\{z \in T: A \cap (S+z) \neq \emptyset\}$. Now the Fubini theorem gives

$$V(A) = \int_{\pi_1(A)} V_2(A \cap (S+z))\,dz \,. \tag{2.1}$$

Analogously, we have

$$\alpha(A) = V(\varphi(A)) = \int_{\pi_1(\varphi(A))} V_2(\varphi(A) \cap (S+y))\,dy \,. \tag{2.2}$$

One can see easily that

$$(\pi_1(A) - y) \cap L = \emptyset \iff y \notin \pi_1(\varphi(A)) \,, \tag{2.3}$$

hence

$$\alpha(A) \leq \int_{\pi_1(\varphi(A))} |(\pi_1(A) - y) \cap L| \cdot V_2(\varphi(A) \cap (S+y))\,dy \,. \tag{2.4}$$

The clue to our results is the following sharpening of (2.4).

Lemma 2.1. For L-measurable $A \subset R^n$ we have

$$\alpha(A) \leq \int_{\pi_1(A)} \alpha_2(A \cap (S+z))\,dz \leq \int_{\pi_1(\varphi(A))} |(\pi_1(A) - y) \cap L| \cdot V_2(\varphi(A) \cap (S+y))\,dy \,. \tag{2.5}$$

□

Proof: The definitions of φ and φ_2 easily show that

$$\varphi(A) \cap (S+y) = \bigcup_{v \in L} \varphi_2(A \cap (S+y+v) - v) \quad . \tag{2.6}$$

By the definition of $\pi_1(A)$ we have

$$(\pi_1(A) - y) \cap L = \{ v \in L : A \cap (S+y+v) \neq \emptyset \} , \tag{2.7}$$

hence

$$\varphi(A) \cap (S+y) = \bigcup_{v \in (\pi_1(A)-y) \cap L} \varphi_2(A \cap (S+y+v) - v) \quad . \tag{2.8}$$

Using a well known identity (see, e.g., [7], p. 36)

$$\int_T f(z)\,dz = \int_Q \Big(\sum_{v \in L} f(v+y) \Big)\, dy \tag{2.9}$$

and the relations (2.3) and (2.7), we get

$$\int_{\pi_1(A)} \alpha_2(A \cap (S+z))\,dz = \int_{\pi_1(\varphi(A))} \Big(\sum_{v \in (\pi_1(A)-y) \cap L} V_2(\varphi_2(A \cap (S+v+y))) \Big)\, dy. \tag{2.10}$$

Now we have only to observe that

$$1 \leq \frac{\sum_{v \in (\pi_1(A)-y) \cap L} V_2(\varphi_2(A \cap (S+v+y)))}{V_2\Big(\bigcup_{v \in (\pi_1(A)-y) \cap L} \varphi_2(A \cap (S+v+y) - v) \Big)} \leq |(\pi_1(A)-y) \cap L| \tag{2.11}$$

and after that integrating (2.11) over $\pi_1(\varphi(A))$ we get (2.5). ■

In the sequel we shall need some simple corollaries of (2.5) rather than (2.5) itself.

Corollary 2.2. If

(2.12) $$|(\pi_1(A)-y)\cap L|=1 \quad \text{for a.e. } y\in\pi_1(\varphi(A)),$$

then

(2.13) $$\alpha(A)=\int_T \alpha_2(A\cap(S+z))\,dz \quad . \qquad \square$$

Proof: Clear. ■

Before formulating two more corollaries, let us write two more identitites.

Let $C\subset T$ be L-measurable and denote

(2.14) $$C_i := \{y\in Q: |C\cap(L+y)|=i\}, \quad i=0,1,2,\dots$$

It is clear that

(2.15) $$\varphi_1(C)=\{y\in Q: C\cap(L+y)\neq\emptyset\}=\{y\in Q:(C-y)\cap L\neq\emptyset\}$$

and

(2.16) $$|C\cap(L+y)|=|(C-y)\cap L| .$$

Hence, we have

(2.17) $$\alpha_1(C)=\sum_{i=1}^{\infty} V_1(C_i) .$$

Further, one can see easily (or write (2.9) for the characteristic function of C) that

(2.18) $$V_1(C)=\sum_{i=1}^{\infty} i\cdot V_1(C_i) .$$

The identities (2.17) and (2.18) hold in a much more general setting as well, [4].

Corollary 2.3. If

(2.19) $$\alpha_1(\pi_1(A)) = V_1(\pi_1(A))$$

then (2.13) is true. □

Proof: Using (2.17) and (2.18) the result is a simple consequence of the previous corollary. ■

Corollary 2.4. If

(2.20) $$\pi_1(\partial A) \cap L = \{\theta\}$$

then (2.13) is true. □

Proof: It is clear that $\pi_1(\partial A) = \partial \pi_1(A)$, so (2.20) is equivalent to $\partial \pi_1(A) \cap L = \{\theta\}$ and this is in turn equivalent to the condition

(2.21) $$|(\pi_1(A) - y) \cap L| \leq 1 \quad \text{for all } y \in Q .$$

Hence the Corollary 2.2 and the implication (2.3) give the result. ■

Remark 2.5. The conditions (2.12), (2.19) and (2.20) are sufficient conditions of the equality in both inequalities in (2.5) simultaneously. The inequalities (2.11) can be used for formulating necessary conditions of equalities in (2.5) in terms of intersections $\varphi_2(A \cap (S + v_1 + y) - v_1) \cap \varphi_2(A \cap (S + v_2 + y) - v_2)$, $y \in \tilde{\pi}_1(\varphi(A))$. □

3. A property of $\alpha(A)$ when A is convex.

Taking into account (1.5), after some simple transformations the inequality (14) on p.217 of [6] says:

If a symmetric bounded convex body $K \subset R^n$ is such that its $(n-k+1)$-th successive minimum is at least 2, i.e.

$$\lambda_{n-k+1} \geq 2, \tag{3.1}$$

then for $0 \leq \lambda \leq 1$ we have

$$\alpha(\lambda K) \leq \lambda^k \alpha(K). \tag{3.2}$$

(The just mentioned inequality (14), p.217, [6], was an important step in proving (1.9)).
Below we shall prove (3.2) not assuming the symmetry of K and using a condition different from (3.1).

Lemma 3.1. Let $K \subset R^n$ be a bounded convex body. If

$$\alpha_1(\pi_1(K)) = V_1(\pi_1(K)), \tag{3.3}$$

then for $0 \leq \lambda \leq 1$ we have

$$\alpha(\lambda K) \leq \lambda^k \alpha(K). \tag{3.4}$$ □

Proof: The quantities involved are translation invariant, so we can assume without the loss of generality, that K contains the origin $\Theta \in R^n$. Denote $U := \pi_1(K)$

U is convex and contains the origin $\Theta \in T$, hence

(3.5) $$\pi_1(\lambda K) = \lambda \pi_1(K) = \lambda U \subseteq U .$$

Denote

(3.6) $$U_i := \{ y \in Q : |U \cap (L+y)| = i \} , \quad i = 0, 1, 2, \cdots .$$

Using the identities (2.17) and (2.18) the condition (3.3) is equivalent to

(3.7) $$V_1(U_i) = 0 \quad \text{for} \quad i \geq 2 .$$

By (3.5) we have $\lambda U \cap (L+y) \subseteq U \cap (L+y)$ and this implies

(3.8) $$(\lambda U)_i \subseteq \bigcup_{j \geq i} U_j ,$$

consequently

(3.9) $$V_1((\lambda U)_i) \leq \sum_{j \geq i} V_1(U_j) ,$$

yielding

(3.10) $$V_1((\lambda U)_i) = 0 \quad \text{for} \quad i \geq 2 .$$

This implies, using again (2.17) and (2.18), that

(3.11) $$\alpha_1(\pi_1(\lambda K)) = V_1(\pi_1(\lambda K)) .$$

By the Corollary 2.3 we have

(3.12) $$\alpha(K) = \int_T \alpha_2(K \cap (S+z))\, dz ,$$

(3.13) $$\alpha(\lambda K) = \int_T \alpha_2(\lambda K \cap (S+z))\, dz \quad .$$

One can see easily that

(3.14) $$\alpha_2(\lambda K \cap (S+z)) \le \alpha_2(K \cap (S + \tfrac{z}{\lambda}))$$

Indeed, let us fix $y_0 \in \tilde{\pi}_2(\lambda K \cap (S+z))$. To each $y \in \tilde{\pi}_2(\lambda K \cap (S+z))$ there is a unique $x \in K$ such that $\lambda x = y + z$. Hence $x_0 = y_0/\lambda + z/\lambda$ and $x = y/\lambda + z/\lambda$ and the convexity of K implies

$$\lambda x + (1-\lambda) x_0 = (1/\lambda - 1) y_0 + y + z/\lambda \in K \quad .$$

But $y_0, y \in S$, hence

(3.15) $$(1/\lambda - 1)\, y_0 + \tilde{\pi}_2(\lambda K \cap (S+z)) + z/\lambda \subseteq K \cap (S + z/\lambda)$$

and this implies (3.14).
So we have

(3.16) $$\alpha(\lambda K) \le \int_T \alpha_2(K \cap (S + z/\lambda))\, dz$$

and changing the variable in the last integral, i.e. taking $w = z/\lambda$, $dz = \lambda^k dw$, we get

(3.17) $$\alpha(\lambda K) \le \lambda^k \int_T \alpha_2(K \cap (S+w))\, dw. \qquad \blacksquare$$

Remark 3.2. The only place in the proof, where the convexity of K plays a role is (3.14) (not counting (3.5) where we used only a "star-shapedness" of a convex set containing the origin). An interesting problem would be to find some star-shaped non-convex sets satisfying (3.14) for a.e. $z \in T$. For such sets, both the lemma and the main result below would hold. □

We shall need also a following weaker version of the previous lemma.

Lemma 3.3. If $K \subset R^n$ is a bounded convex body such that

(3.18) $$\pi_1(\partial K) \cap L = \{\theta\}$$

then (3.4) holds. □

Proof: Clear, the condition (3.18) implies (3.3) via (2.21). ■

If K is symmetric, then one can compare the conditions (3.3) and (3.18) with (3.1). We shall turn to this question in the last section.

4. The projection successive minima

We recall that $\mathcal{B}$ is the set of bases of Λ and $J := (1,2,\ldots,n)$. Let $A \subset R^n$ be a measurable set. Let $t_0(A) := +\infty$ and for $1 \leq k \leq n$ define

(4.1) $$t_k(A) := \sup \{ t : t \geq 0, \text{ there is } B \in \mathcal{B} \text{ and } I \subseteq J, |I| = k \text{ such that } \alpha_1(\pi_I(tA)) = V_I(\pi_I(tA)) \}.$$

Similarly, $s_0(A) := +\infty$ and for $1 \leq k \leq n$ define

(4.2) $$s_k(A) := \sup \{ t : t \geq 0, \text{ there is } B \in \mathcal{B} \text{ and } I \subseteq J, |I| = k \text{ such that } \pi_I(\partial tA) \cap L = \{\theta\} \}.$$

We shall call these quantities projection successive minima of A (for justification see below).
It is clear that

$$0 \le t_n(A) \le t_{n-1}(A) \le \cdots \le t_1(A) \le t_0(A) := +\infty \,, \tag{4.3}$$

$$0 \le s_n(A) \le s_{n-1}(A) \le \cdots \le s_1(A) \le s_0(A) := +\infty \tag{4.4}$$

and

$$s_k(A) \le t_k(A) \,. \tag{4.5}$$

Let us write some equivalent forms of s_k and t_k (we use (2.17), (2.18)).

$$s_k(A) = \sup \{ t : \exists\, B \in \mathcal{B} \,,\ I \subseteq J, |I| = k \ \text{s.t.} \ |\pi_I(tA) \cap (L+y)| \le 1 \ \forall\, y \in Q \} \,, \tag{4.6}$$

$$s_k(A) = \inf \{ t : |\pi_I(\mathring{\mathcal{O}}tA) \cap L| > 1 \ \text{for all } B \in \mathcal{B} \text{ and } I \subseteq J, |I| = k \} \,, \tag{4.7}$$

$$t_k(A) = \sup \{ t : \exists\, B \in \mathcal{B}, I \subseteq J, |I| = k \ \text{s.t.} \ |\pi_I(tA) \cap (L+y)| \le 1 \ \text{for a.e. } y \in Q \} \tag{4.8}$$

$$t_k(A) = \inf \{ t : \forall B \in \mathcal{B} \text{ and } \forall I \subseteq J, |I| = k \ \text{ there is } H \subseteq Q \ V_1(H) > 0 \ \text{s.t.} \ |\pi_I(tA) \cap (L+y)| \ge 2 \ \text{for } y \in H \} \,. \tag{4.9}$$

Now we prove

Theorem 4.1 If $K \subset R^n$ is a bounded convex body then

$$\alpha(tK)\begin{cases} = t^n V(K) & \text{if } 0 \le t \le s_n(K) \\ \ge t^k V(K) \prod_{i=k+1}^{n} s_i(K) & \text{if } s_{k+1}(K) \le t \le s_k(K) \\ & k = 0, 1, \dots, n-1 \end{cases} \tag{4.10}$$

□

Proof: All quantities are translation invariant, hence we can assume that $\Theta \in K$. Further, we can assume that K is open, because one can see easily that all quantities are invariant under the closure operation i.e. $V(K)=V(\overline{K})$, $\alpha(tK)=\alpha(t\overline{K})$ and $s_k(K)=s_k(\overline{K})$. This implies that also $V(K_o)=V(K)$, $\alpha(tK_o)=\alpha(tK)$ and $s_k(K_o) = s_k(K)$ where $K_o \subseteq K$ is an open convex set such that $\overline{K}_o=\overline{K}$ (the open kernel of K). Denote for short $s_k=s_k(K)$. For open K the supremum in the definition of s_k is attained, i.e. there is $B \in \mathcal{B}$ and $I \subseteq J$, $|I|=k$ such that

$$\pi_I(\partial(s_k K)) \cap L = \{\Theta\} . \tag{4.11}$$

Indeed, the set $\pi_I(\partial(s_k K))$ is open, hence any lattice point in it is an inner point, consequently, if

$$|\pi_I(\partial(s_k K)) \cap L| > 1 \tag{4.12}$$

for all $B \in \mathcal{B}$ and $I \subseteq J$, $|I|=k$ then for a sufficiently small $\varepsilon > 0$ also

$$|\pi_I(\partial((s_k - \varepsilon)K) \cap L| > 1 \tag{4.13}$$

for all $B \in \mathcal{B}$ and $I \subseteq J$, $|I|=k$ that is a contradiction.

For all $0 \le t \le s_n$ we have $t\mathcal{B}K \cap \Lambda = \{\theta\}$. This implies that $|(tK - x) \cap \Lambda| \le 1$ for all $x \in P$, consenquently $\alpha(tK) = V(tK)$ by (2.17) and (2.18) (applied for $T = R^n$). Hence (4.10) is true. Assume that we have proved (4.10) for $t = s_{k+1}$. If $s_{k+1} = s_k$ then $\alpha(s_k K) \ge s_k^{k-1} V(K) \prod_{i=k}^{n} s_i$, so (4.10) is true for $t = s_k$ as well. Let $s_{k+1} < s_k$ and $t = \tau s_{k+1}$, $1 < \tau \le s_k / s_{k+1}$. Hence there are $B \in \mathcal{B}$ and $I \subseteq J$, $|I| = k$ such that $\pi_I(\mathcal{B}tK) \cap L = \{\theta\}$. By the Lemma 3.3 we have

$$\tau^{-k} \alpha(tK) \ge \alpha((t/\tau)\cdot K) = \alpha(s_{k+1} K) \ge s_{k+1}^k V(K) \prod_{i=k+1}^{n} s_i = \tag{4.14}$$

$$= \left(\frac{t}{\tau}\right)^k V(K) \prod_{i=k+1}^{n} s_i$$

that proves (4.10). ■

Using the Lemma 3.1, we can prove analogously the following

Theorem 4.2. If $K \subset R^n$ is a bounded convex body then

$$\alpha(tK) \begin{cases} = t^n V(K) & \text{if } 0 \le t \le t_n(K) \\ \ge t^k V(K) \prod_{i=k+1}^{n} t_i(K) & \text{if } t_{k+1}(K) \le t \le t_k(K) \end{cases} \tag{4.15}$$

$$k = 0, 1, \dots, n-1 . \qquad \square$$

Taking in (4.10) and (4.15) specific values of t, say t=1, we get useful lower estimations for $\alpha(K)$. For k=0 we get

$$\prod_{i=1}^{n} t_i(K)\, V(K) \le \alpha(t_1(K) K) \le d\Lambda , \tag{4.16}$$

$$\prod_{i=1}^{n} s_i(K)\, V(K) \le \alpha(s_1(K) K) \le d\Lambda . \tag{4.17}$$

These two inequalities can be compared with the Minkowski successive minima theorem (more exactly with the inequality (1.8), see below for the details).

5. Remarks

The proof of (1.9) which can be found in [6], pp. 215-218 seems to work also in non-symmetric case, i.e. we think the following refinement of (1.8) holds (here $\lambda_{n+1}(\partial K) := +\infty$): for any bounded convex body $K \subset R^n$ we have

$$(5.1)\quad V(\varphi(tK)) \begin{cases} = t^n V(K) & \text{if } 0 \le t \le \lambda_1(\partial K), \\ \ge t^{n-k} V(K) \prod_{i=1}^{k} \lambda_i(\partial K) & \text{if } \lambda_k(\partial K) \le t \le \lambda_{k+1}(\partial K) \\ & k = 1, 2, \dots, n \end{cases}$$

This result can be compared with (4.10) and (4.15) (in the sense which of them gives sharper lower estimations for $\alpha(tK)$).

One can see easily that

$$(5.2)\qquad s_{n-k}(K) \le \lambda_{k+1}(\partial K), \quad k = 0, 1, \dots, n-1.$$

If, say,

$$(5.3)\qquad s_1(K) = \lambda_n(\partial K),$$

then (1.8) (or (5.1) for $t = \lambda_n(\partial K)$) is stronger than (4.17).

But if

$$s_1(K) < \lambda_n(\partial K) \tag{5.4}$$

then it is not quite clear, which of the results (4.17) or (1.8) is sharper, because we can have

$$\prod_{i=1}^{n} s_i(K) \cdot V(K) \leq \alpha(s_1(K)K) < \alpha(\lambda_n(\partial K)K) \tag{5.5}$$

and on the other hand we have

$$\prod_{i=1}^{n} s_i(K) \cdot V(K) < \prod_{i=1}^{n} \lambda_i(\partial K) \cdot V(K) \leq \alpha(\lambda_n(\partial K)K) \tag{5.6}$$

i.e. (5.5) cannot be derived from (5.6). The comparison of finer results (4.10) and (4.15)with that of (5.1) seems to be much more complicated,because it depends not only on the successive minima themselves, but also on their intervals, say, how $[s_{n-k+1}(K), s_{n-k}(K)]$ can be compared with $[\lambda_k(\partial K), \lambda_{k+1}(\partial K)]$.

References

[1] B Uhrin, A remark to the paper of H. Hadwiger "Überdeckung des Raumes durch translationgeliche Punktmengen und Nachbarnzahl", Monatshefte f. Math, to appear.

[2] B. Uhrin, Some useful estimations in geometry of numbers, Periodica Math. Hungar., 11(1980), 95-103.

[3] B. Uhrin, Minkowski convex body theorem and the measure of covering R^n by a set, MTA SZTAKI Közlemények (Trans. of Comp. Automat. Inst. of HAS), 29/1983,115-121.

[4] B. Uhrin, Some remarks about the lattice points in difference sets In: J. Szabados, Ed.: "Proc. of the Haar Mamorial Conference", Budapest, 1985. Colloquia Math. Soc. J. Bolyai, Vol. 49, North-Holland, Amsterdam-New York, 1987, 929-937.

[5] C.G. Lekkerkerker, "Geometry of Numbers", Biblio. Math., Vol VIII, North-Holland, Amsterdam-New York, 1969.

[6] J.W.S. Cassels, "An Introduction to the Geometry of Numbers" Grundl. Math. Wiss., 99, Springer, Berlin-New York, 1959.

[7] A. Weil, "Basic Number Theory", Grundl. Math. Wiss., 144, Springer, Berlin-New York, 1967.

The measure of covering the space by a set A and the projection successive minima of A-A

B. Uhrin

Summary

Let $\Lambda \subset R^n$ be a point lattice, $P \subset R^n$ be a unit cell of Λ. For a measurable set $A \subset R^n$ the measure (volume) V of the set $\bigcup_{u \in \Lambda} (A+u) \cap P)$ is called the measure of covering R^n by the A (denote this by $\alpha(A)$). If A is convex and symmetric about the origin and $\lambda_k \leq 2 \leq \lambda_{k+1}$, where λ_j are the successive minima of A, then from a known estimation we get that $\alpha(A) \geq V(A) \prod_{i=1}^{k} (\lambda_i/2)$. In the paper, after introducing so called projection successive minima, two new estimation of similar type are proved, not assuming the symmetry of A. The proofs depend on an integral inequality that seems to be interesting also in itself.

MTA SZTAKI Tanulmányok 202/1987 pp. 227-232

A NOTE ON FACTORING POLYNOMIALS MODULO SPECIAL PRIMES

by

Lajos Rónyai *

Computer and Automation Institute
Hungarian Academy of Sciences
Budapest, P.O.B. 63
H-1502 Hungary

Abstract

We consider the problem of factoring polynomials over $GF(p)$ for those prime numbers p for which all prime factors of $p-1$ are small. We show that if we have a primitive t-th root of unity for every prime t dividing $p-1$ then factoring polynomials over $GF(p)$ can be done in deterministic polynomial time.

1. Introduction

J. von zur Gathen [1986] had considered the problem of factoring polynomials over $GF(p)$ in the case when $p-1$ has small prime factors. Following his definition, the *smoothness* $S(k)$ of an integer k is the largest prime factor of k. He showed that the problem of factoring polynomials over $GF(p)$ and the problem of finding a primitive element in $GF(p)$ are polynomial time equivalent via Cook reductions. Here 'polynomial time' means polynomial time in the input size plus $S(p-1)$. Also he proved that if one assumes the Extended Riemann Hypothesis (ERH), then primitive elements can be found in time $(\log p + S(p-1))^{O(1)}$. Thus, under ERH, one can factor polynomials over $GF(p)$ using $(n + \log p + S(p-1))^{O(1)}$ bit operations, where n is the degree of the polynomial to be factored.

Before formulating our result, we introduce some notation. In this paper p denotes an odd prime and

$$p-1 = {p_1}^{e_1} {p_2}^{e_2} \cdots {p_r}^{e_r}$$

denotes the prime factorization of $p-1$. The *socle* $soc(p)$ of $GF(p)$ is defined as

$$soc(p) = \{\ \zeta \in GF(p),\ \zeta \textit{ is a } t-\textit{th root of unity for some prime } t \mid p-1\ \}.$$

Clearly we have $m = |soc(p)| = p_1 + p_2 + \cdots + p_r - r + 1 \leq \log pS(p-1)$.

In this note we prove the following

* *Research partially supported by Hungarian National Foundation for Scientific Research, Grant 1812.*

Theorem 1.1. *Let p be an odd prime and suppose that $soc(p)$ is given. Then we can factor polynomials over $GF(p)$ in time $(n + \log p + S(p-1))^{O(1)}$, where n is the degree of the polynomial to be factored.*

Remark 1.2. We improve the factoring result of von zur Gathen [1986, Section 4] in that we use an element from $GF(p)$ of order $p_1 p_2 \cdots p_r$ instead of an element of order $p-1$. Putting it an other way, for primes with $S(p-1)$ small we can factor polynomials if we can factor the cyclotomic polynomials $\frac{x^{p_i}-1}{x-1}$.

Let M_n denote the set of invertible n by n matrices over the field $GF(p)$ which are similar over $GF(p)$ to a diagonal matrix. D_n denotes the set of n by n invertible scalar matrices over $GF(p)$ (i.e. the matrices of form αI where $0 \neq \alpha \in GF(p)$ and I is the n by n identity matrix).

The multiplicative group $GF(p)^\times$ of $GF(p)$ can be written as the direct product of its Sylow p_i subgroups S_i

$$GF(p)^\times = S_1 \times S_2 \times \cdots \times S_r.$$

For an element $\alpha \in GF(p)^\times$ the multiplicative order of α is denoted by $o(\alpha)$. $o(\alpha)$ is the smallest positive integer k such that $\alpha^k = 1$. Every element $\alpha \in GF(p)^\times$ can be expressed uniquely as

$$\alpha = \alpha_1 \alpha_2 \cdots \alpha_r$$

where $\alpha_i \in S_i$. The elements α_i can be obtained from α using $(\log p)^{O(1)}$ bit operations if the primes p_i are known. Indeed, we can efficiently compute the multiplicative inverse $1 \leq c_i \leq {p^{e_i}}_i$ of the integer $d_i = \frac{p-1}{{p_i}^{e_i}}$ modulo ${p_i}^{e_i}$. Next, using fast exponentiation, we can compute $\alpha_i = \alpha^{d_i c_i}$.

This observation is readily generalized to matrices as follows. If $A \in M_n$, then A can be written as

$$(*) \qquad A = A_1 A_2 \cdots A_r$$

where $A_i = A^{d_i c_i} \in M_n$, the characteristic roots of A_i are in S_i and the relations $AA_i = A_i A$ hold. The matrices A_i can be obtained from A in time $(n + \log p)^{O(1)}$ if the primes p_i are known.

The problem of factoring polynomials over $GF(p)$ is closely related to the problem of finding invariant subspaces of matrices over $GF(p)$. Indeed, if

$$f(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} + x^n \in GF(p)[x]$$

is a polynomial to be factored then we can form the *companion matrix* A_f of f

$$A_f = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-1} \end{pmatrix}.$$

A_f is an n by n matrix over $GF(p)$ and for the characteristic polynomial we have $det(A_f - xI) = (-1)^n f(x)$. If we have a nontrivial invariant subspace U of A_f (acting on the linear space of column vectors of length n over $GF(p)$), then we can consider the action of A_f on U. In this way we obtain a linear transformation of U, and its characteristic polynomial is a nontrivial divisor of f. To obtain a nontrivial invariant subspace of A_f, we try to find a matrix B such that $BA_f = A_f B$ and the characteristic polynomial f_B of B has at least two different roots in $GF(p)$. If α is a root of f_B then $ker(B - \alpha I)$ is a nontrivial invariant subspace of A_f. A particularly important special case is when $f_B \mid x^p - x$ (i.e. f_B splits in $GF(p)$ and has no multiple roots). Then f also splits in $GF(p)$ and from the roots of f_B we can obtain the roots of f because of $dim_{GF(p)} ker(B - \alpha I) = 1$ for every root α of f_B. An other useful special case is when $B \in M_n \setminus D_n$ and f_B has multiple roots. Then for the minimal polynomial h of f_B we have $h \mid x^{p-1} - 1$ and $deg\ h < n$. If we can find a nontrivial factor h_1 of h then we can find a nontrivial factor of f, for $ker(h_1(B))$ is a nontrivial invariant subspace of the matrix A_f. A matrix B satisfying $BA = AB$, $B \in M_n \setminus D_n$ and f_B has multiple roots is a *splitting matrix for* f .

We note that the linear algebraic computations mentioned in the preceding discussion (finding characteristic polynomial, minimal polynomial, computing the kernel, computing the action on an invariant subspace) can be done using $(n + \log p)^{O(1)}$ bit operations.

2. Matrices and polynomials

In this section we collected some facts and results related to the factoring algorithm to be presented in Section 3.

Fact 2.1. *Suppose that* $H, K \subset GF(p)$, $|K| + |H| = k \leq p - 2$ *and let* $\alpha \neq \beta \in GF(p)$. *Then there exists an integer* i, $0 \leq i \leq k + 1$ *such that* $i \notin K$ *and* $\frac{\alpha+i}{\beta+i} \notin H$.

Proof. It is clear that for $\gamma \in GF(p)$ the equation

$$\frac{\alpha + x}{\beta + x} = \gamma$$

has at most one solution x_γ. Also, $\beta + x = 0$ holds only for $x = -\beta$. Thus, for the number of forbidden values of i we have $|K \cup H \cup \{-\beta\}| \leq k + 1$. The elements i, $0 \leq i \leq k + 1$ are different modulo p, hence the result follows. ■

Fact 2.2. *Let* $\alpha \neq \beta \in GF(p)^\times$. *If* $\alpha^{p_i} = \beta^{p_i}$ *for some* $1 \leq i \leq r$ *then* $\alpha/\beta \in soc(p)$.

Proof. If $\alpha^{p_i} = \beta^{p_i}$ then $(\alpha/\beta)^{p_i} = 1$ hence $\alpha/\beta \in soc(p)$. ■

For a matrix $A \in M_n$ and a natural number i we put $A_{(i)} = A + iI$ where I denotes the n by n identity matrix. The decomposition (*) of $A_{(i)}$ is written as

$$A_{(i)} = A_{i1} A_{i2} \cdots A_{ir}.$$

Lemma 2.3. *Let $A \in M_n \setminus D_n$ and suppose that $n + |soc(p)| \leq p - 2$. Then there exist integers i, j, $0 \leq i \leq m+n+1$, $1 \leq j \leq r$ such that one of the following statements holds:*
(a) $n \neq p_j$ and $A_{ij} \in M_n \setminus D_n$.
(b) $n = p_j$ and $A_{ij}^{p_j} \in M_n \setminus D_n$.

Proof. As $A \in M_n \setminus D_n$, it has at least two different eigenvalues $\alpha, \beta \in GF(p)^\times$. Applying Fact 2.1 with $H = soc(p)$ and $K = \{-\gamma \ ; \ \gamma \ is \ a \ characteristic \ root \ of \ A \ \}$, we obtain that there exists an integer i, $0 \leq i \leq m+n+1$ such that $A_{(i)} \in M_n$ and $A_{(i)}$ has two characteristic roots $\gamma, \delta \in GF(p)^\times$ for which $\gamma/\delta \notin soc(p)$. Now we consider the matrices A_{ij}. If at least two of them are in $M_n \setminus D_n$ then (a) holds. We can therefore assume that there exists exactly one j such that $A_{ij} \in M_n \setminus D_n$. If $n \neq p_j$ then we have (a) again. Now suppose that $n = p_j$. Clearly $A_{ij}^{p_j} \in D_n$ if and only if $A_{(i)}^{p_j} \in D_n$. By Fact 2.2 the latter is impossible and hence (b) follows. ■

Remark 2.4. For a given matrix $A \in M_n \setminus D_n$ the matrices A_{ij} can be computed using $(m+n+\log p)^{O(1)}$ bit operations, hence we can efficiently find a matrix A_{ij} satisfying (a) or (b) of Lemma 2.3.

Fact 2.5. *Let t be a prime and $A \in M_t$ such that $A^t = diag(\alpha, \ldots, \alpha) \in D_t$. Suppose further that the characteristic polynomial f_A of A has no multiple roots. Then $det(A) = (-1)^{t+1}\alpha$.*

Proof. If f_A has no multiple roots, then we have $f_A = (-1)^t(x^t - \alpha)$. Using the fact that $det(A)$ is the constant term of f_A, the statement follows. ■

Fact 2.6. *Let t be a prime and $A \in M_t$ such that $A^{t^2} = diag(\alpha, \ldots, \alpha) \in D_t$. Suppose that A^t has no multiple characteristic roots. Then $det(A)^t = (-1)^{t+1}\alpha$.*

Proof. Using Fact 2.5, we obtain that $(-1)^{t+1}\alpha = det(A^t) = det(A)^t$. ■

Fact 2.7. *Let $t = p_i$ be an arbitrary prime divisor of $p-1$ and $A \in M_n$, $n < t$ such that $A^t = diag(\alpha, \ldots, \alpha) \in D_n$ for $1 \neq \alpha \in S_i$. Then $det(A) \in S_i$ and $o(det(A)) > o(\alpha)$.*

Proof. From $gcd(n,t) = 1$ we obtain $1 < t^k = o(\alpha) = o(\alpha^n)$. Also we have $\alpha^n = det(A^t) = det(A)^t$. This implies that $det(A^{t^{k+1}}) = 1$ hence $det(A) \in S_i$. Finally, we observe that $det(A^{t^k}) = 1$ is impossible because it would imply $\alpha^{t^{k-1}} = 1$, a contradiction. ■

Lemma 2.8. *Let t be a prime dividing $p-1$ and suppose that we have elements $\alpha, \beta \in GF(p)^\times$ such that $o(\alpha) = t^k < o(\beta) = t^l$. Then the roots of the polynomial $x^t - \alpha$ are in $GF(p)$ and can be found in time polynomial in $t + \log p$.*

Proof. The conditions imply that α is in the multiplicative subgroup generated by β. More precisely, there exists a natural number $j \leq t^l$ such that $\alpha = \beta^j$ and j is divisible by t. This exponent j can be computed in time polynomial in t and $\log p$ using essentially the Tonelli - Shanks algorithm (Tonelli [1891], Shanks [1972], Adleman, Manders, Miller [1977], Huang [1985, Section 2], von zur Gathen [1986, Lemma 3.1]). Also, using β, we can find a primitive t-th root of unity γ from $GF(p)$ in time $(t+\log p)^{O(1)}$. Now observing that the roots of $x^t - \alpha$ are $\beta^{j/t}\gamma^i$ where $1 \leq i \leq t$, the statement is proved. ■

3. The factoring method

Now we are in a position to describe our factoring procedure. The input is a polynomial $f \in GF(p)[x]$, $deg(f) = n > 1$ such that the roots of f are in $GF(p)$. We shall also assume that $f(0) \neq 0$ and that f has no multiple roots. The algorithm either produces the complete factorization of f or a splitting matrix $B \in M_n$ for f. We assume that $soc(p)$ is explicitly given. We can also assume that $n+m \leq p-2$, otherwise one can use Berlekamp's algorithm (Berlekamp [1968], [1970], Knuth [1981], Lidl, Niederreiter [1983]) to obtain the complete factorization of f.

Step 1. *Form the companion matrix $A = A_f \in M_n$ and compute the matrices A_{ij} for $0 \leq i \leq n+m+1$ and $1 \leq j \leq r$. Find indices i, j for which one of the alternatives of Lemma 2.3 holds and put $B = A_{ij}$, $t = p_j$. Next generate the sequence of matrices*

$$B, B^t, B^{t^2}, \ldots, B^{t^k}$$

until we obtain a scalar matrix $B^{t^k} = diag(\alpha, \ldots, \alpha)$. Compute the characteristic polynomial g of $B^{t^{k-1}}$.

Step 2. *If the characteristic polynomial g of $B^{t^{k-1}}$ has multiple roots then $B^{t^{k-1}}$ is a splitting matrix for f,* **return**$(B^{t^{k-1}})$.
(The polynomial g has no multiple roots, consequently $n \leq t$. *)*

Step 3. *If $n \neq t$ then skip the rest of this step.*
(For the rest of Step 3 we have $n = t$ and $k \geq 2$. *)*
If t is odd, then by Fact 2.6 $det(B^{t^{k-2}})$ is a root of g, and having $soc(p)$ (and thus a primitive t-th root of unity) at hand, we can find all the roots of g and then find the complete factorization of f. If $t = 2$, then first we find a root γ of the polynomial $x^2 + 1$, using the algorithm of Schoof [1985] for taking modular square roots of small integers. Now $\gamma det(B^{t^{k-2}})$ is a root of g and we can proceed as in the o ld case to find the roots of f. In all cases we return the complete factorization of f.

Step 4. *(* Here we have $n < t$ and $k > 0$. *)*
If $\alpha = 1$ then the roots of g are in $soc(p)$, therefore they can be found by computing the elements $g(\zeta)$, $\zeta \in soc(p)$. In this way we obtain the complete factorization of f. If $\alpha \neq 1$ then by Fact 2.7 $det(B^{t^{k-1}}) \in S_j$ and $o(det(B^{t^{k-1}})) > o(\alpha)$, so by Lemma 2.8 we can factor $x^t - \alpha$ and thus find the roots of g. Again, we obtain the complete factorization of f.
We return the complete factorization of f.

Lemma 3.1. *Let $f \in GF(p)[x]$ be a polynomial such that $deg\ f = n > 1$, $f \mid x^{p-1} - 1$. Suppose further that the set $soc(p)$ is explicitly given. Then the preceding algorithm either finds the roots of f or produces a splitting matrix for f. It runs in deterministic time $(n + S(p-1) + \log p)^{O(1)}$.*

Proof. If we finish at Step 2 then we have a splitting matrix for f. If we finish at Step 3 or at Step 4 then we have found all the roots of f. Observing that $k \leq \log p$ and $t \leq S(p-1)$ the timing follows. ■

Now we can prove Theorem 1.1.

Proof of Theorem 1.1. The problem of factoring $f \in GF(p)[x]$, $deg\ f = n$ can be reduced in time $(n + \log p)^{O(1)}$ to finding roots in $GF(p)$ of at most n polynomials of degree at most n, using Berlekamp's reduction (Berlekamp [1968], [1970], Knuth [1981], Lidl, Niederreiter [1983]). The polynomials obtained split in $GF(p)$. We can assume that $f(0) \neq 0$ and by computing $gcd(f, x^{p-1} - 1)$ we can assure that f has no multiple roots. This can be done using $(n + \log p)^{O(1)}$ bit operations.

Clearly it suffices to show that f can be factored into at least two nonconstant factors in time $(n + S(p-1) + \log p)^{O(1)}$. To this end we apply the algorithm of Lemma 3.1. If it finds the roots of f then we are done. Otherwise it returns a splitting matrix B_1 for f. Let f_1 denote the minimal polynomial of B_1. We have $f_1 \mid x^{p-1} - 1$ and $1 < deg\ f_1 < deg\ f$. We can apply the algorithm of Lemma 3.1 to f_1 and so on. More formally we compute a sequence of matrices $B_1, \ldots, B_i$ and a sequence of polynomials $f = f_0, f_1, \ldots, f_i$ such that B_{j+1} is a splitting matrix for f_j, $0 \leq j < i$ and $f_j \mid x^{p-1} - 1$ is the minimal polynomial of B_j, until f_i is completely factored by our algorithm. Using the fact that $deg\ f_j > deg\ f_{j+1} > 1$, these sequences can be generated in time $(n + S(p-1) + \log p)^{O(1)}$. Then, proceeding backwards, from the factorization of f_i we obtain (partial) factorization of $f_{i-1}, \ldots, f_0 = f$ using repeatedly the technique described at the end of Section 1. This task can be done in time $(n + \log p)^{O(1)}$. We have obtained that f can be factored into at least two nonconstant factors using $(n + S(p-1) + \log p)^{O(1)}$ bit operations. The proof is complete. ■

References

L. Adleman, G. Miller, K. Manders, On taking roots in finite fields; *Proc. 18th IEEE Symp. on Foundations of Computer Science (1977), 175-178.*

E. R. Berlekamp, Algebraic coding theory;*McGraw-Hill, 1968.*

E. R. Berlekamp, Factoring polynomials over large finite fields; *Math. Computation 24 (1970), 713-735.*

J. von zur Gathen [1986], Factoring polynomials and primitive elements for special primes; to appear in *Theoretical Computer Science.*

M. A. Huang, Riemann Hypothesis and finding roots over finite fields; *Proc. 17th ACM Symp. on Theory of Computing, (1985), 121-130.*

D. E. Knuth, The art of computer programming; Vol. 2, Seminumerical algorithms *Addison-Wesley Publishing Co. 1981.*

R. Lidl, H. Niederreiter, Finite fields; *Addison-Wesley Publishing Co. 1983.*

R. J. Schoof, Elliptic curves over finite fields and the computation of square roots mod p; *Mathematics of Computation 44(1985), 483-494.*

D. Shanks, Five number-theoretic algorithms; in *Proc. 1972 Number Theory Conference, University of Colorado, Boulder 1972, 217-224.*

A. Tonelli, *Göttingen Nachrichten (1891), 344-346.* Also in *L.E. Dickson, History of the theory of numbers, Chelsea, New York, Vol. I, 215.*