PREPRINT OF THE PROCEEDINGS OF THE IFIP CONGRESS 62
INTERNATIONAL FEDERATION FOR INFORMATION PROCESSING

MUNICH
AUGUST 27 TO SEPTEMBER 1, 1962

ENGLISH ABSTRACTS

NORTH HOLLAND PUBLISHING COMPANY
AMSTERDAM
PREPRINT OF THE PROCEEDINGS OF THE
IFIP CONGRESS 62
INTERNATIONAL FEDERATION FOR INFORMATION PROCESSING

MUNICH
AUGUST 27 TO SEPTEMBER 1, 1962

ENGLISH ABSTRACTS

NORTH HOLLAND PUBLISHING COMPANY
AMSTERDAM
CONTENTS

1 Business Data Processing I

Data processing in English banks, R. Hindle . . I, 1
Computer uses at Lamp Department, Canadian General Electric, A. Leigh . . . . I, 2

2 Business Data Processing II

Mathematical analysis of merge-sorting techniques, W. C. Carter . . . . . . . . II, 1
Some airline applications of Monte-Carlo system simulations, J. P. Jeanniot and P. J. Sandiford II, 2
The construction of class-teacher time-tables, C. C. Gotlieb . . . . . . . . . . . II, 3
Extending management capability by electronic computers, A. Vazsonyi . . . . . II, 4

3 Algebra

On a modification of the QD-algorithm with Graeffe-type convergence, H. Rutishauser . . III, 1
Sur certains procédés itératifs non linéaires de résolution de systèmes d'équations du premier degré, N. Gastinel . . . . . . . . III, 2
Les polynômes d'interpolation de matrices carrées à coefficients matriciels, et les méthodes itératives de résolution numérique des équations de matrices carrées de forme quelconque, A. Korganoff III, 3
A method for solving simultaneous polynomial equations, P. H. Blundell . . . . . . . . III, 4
Strategy for multidimensional neutron group diffusion computations, E. L. Wachpress . . . III, 5
On some methods for computing the roots of polynomials, J. L. Howland . . . . . . . III, 6

4 Partial Differential Equations

Automatic calculation and programming of difference equations for elliptic boundary value problems, M. Engeli and P. Lauchli . . . . IV, 1
Numerical studies of implicit iterative methods for solving elliptic difference equations, D. J. Evans
IV, 2
Numerical calculation of shock waves, L. Guerri
IV, 3
A mathematical model of drug distribution and the solution of differential-difference equations, B. Kotkin
IV, 4

5 Differential and Integral Equations
Nouvelles méthodes pour l'intégration approchée des équations différentielles, J. Kuntzmann
V, 1
Recherche des solutions d'une équation de convolution, J. Arsac
V, 2
A difference method for the approximate solution of the initial value problem for systems of quasi-linear partial differential equations of the first order, R. Albrecht and W. Urich
V, 3

6 Optimization Programming
The use of approximation methods in linear programming, J. Habr
VI, 1
Application of the steepest ascent method to convex programming, T. Pietrzykowski
VI, 2
A breakpoint technique for network problems, J. M. Bennett
VI, 3
Application de la programmation linéaire en nombres entiers à un problème de découpe, F. Genuys
VI, 4

7 Real-Time Information Processing
The control of traffic signals with an electronic computer - A new application of real-time data processing, L. Casciato
VII, 1
Utilization of an analogue-to-digital linkage system in a big scientific computing centre, A. Debroux, G. P. Del Bigio, A. Gazzano, C. Green, H. d'Hoop, A. Riotte and A. van Wauwe
VII, 2
Met-Watch. A technique for processing and scanning meteorological data with a digital computer, R. B. Stauffer and T. H. Lewis
VII, 3
Nouvelle méthode de calcul du dispatching économique d'un réseau de transport d'énergie, J. Carpentier
VII, 4

8 Information Retrieval
The multi-list system for real-time storage and retrieval, N. S. Prywes and H. J. Gray
VIII, 1
Le SYNTOL (syntagmatic organization language)
J. C. Gardin
VIII, 2
A method for using computers in information classification, R. M. Needham
VIII, 3
Interrogating a computer in natural language, D. R. Swanson
VIII, 4

9 Linguistic Analysis and Mechanical Translation of Languages
Multiple-path syntactic analyzer, S. Kuno and A. G. Oettinger
IX, 1
Some axiomatic systems for formal grammars and languages, K. Čulík
IX, 2
Rules of interpretation - An approach to the problem of computation in the semantics of natural language, M. Kay
IX, 3
Machine translation and/or an international language? K. G. Sellin
IX, 4

10 Digital Communication
Computer-to-computer communication at 2.5 megabits/sec, N. Clark and A. C. Gannet
X, 1
Dependence of speech quality on transmitted information rate in a band compression system, E. Rothauser and F. Lenk
X, 2
Self-correcting decoding circuits, K. Steinbuch and F. Zendeh
X, 3
Message protection features of the ComLogNet program, A. E. Miller, A. B. Shafritz and J. R. Smith
X, 4

11 Automata Theory
Fundamentals of a theory of asynchronous information flow, C. A. Petr
XI, 1
Finite and combinatorial automata. Turing automata with a programming tape, J. Bečvář
XI, 2
Toward inductive inference automata, L. J. Fogel
XI, 3
Generalization of an elementary perceiving and memorizing machine, E. A. Feigenbaum and H. A. Simon ..................................... XI, 4

12 Machine Learning

Computer simulations of a perceptual learning model for sensory pattern recognition, concept formation, and symbol transformation, C. Vossler and L. Uhr .................................... XII, 1
Self-organizing grouping - A learning structure, V. Kudielka ........................................ XII, 2
The development of a conditional probability computer for control applications, H. C. Ratz and G. H. M. Thomas ......................... XII, 3
Simulation of a learning machine for playing GO, H. Remus ........................................ XII, 4

13 Artificial Perception

Digital computer usage in analysis of electroencephalograph and similar quasi-rhythmic patterns, M. G. Saunders ...................... XIII, 1
Towards the automation of binocular depth perception, B. Julesz .................................. XIII, 2
The phonetic typewriter, T. Sakai and S. Doshita ................................ XII, 3
MUSE: a sound synthesizer, A. W. Slawson ................................ XIII, 4
Un système analogique-digital pour la reconnaissance de caractères, M. Nadler ....................... XIII, 5
Machine recognition of cursive writing, L. D. Earnest ................................................ XIII, 6

14 Programming Languages and their Processing

ALGOL-60 processors and a processor generator, M. Paul ....................................... XIV, 1
An algorithm for the translation of ALGOL statements, W. M. Keese Jr. and H. D. Huskey ... XIV, 2
On table operating algorithms, L. Lombardi ........................................ XIV, 3
A proposed ALGOL-60 matrix scheme, S. J. M. Denison ........................................ XIV, 4

15 Advanced Programming

Program organization and record keeping for dynamic storage allocation, A. W. Holt ........ XV, 1
Programmed control of multi-computer systems, R. Perkins and W. C. McGee .............. XV, 2
Automatic translation of programs from one computer to another, A. Opler, D. Farberman, M. Heit, W. King, E. O'Connor, R. Goldfinger, H. Landow, J. Ogle and D. Slesinger ... XV, 3
Requirements on a language for logical data processing, P. Lucas ............................... XV, 4

16 Memory Techniques

Nanosecond speed in a core memory with non-destructive read-out, J. Scharbert ........ XVI, 1
Some problems in the design of magnetic film storage systems operating at millimicrosecond speeds, J. D. R. McQuillan ......................... XVI, 2
A read-out circuit for high speed non-destructively read stores, G. H. Perry and E. W. Shallow .................... XVI, 3
Tunnel diode high-speed memory, S. Takahashi, O. Ishii, K. Nakazawa and K. Murata ...... XVI, 4

17 Circuits and Components

Size and speed of thin-magnetic-film computer units, H. J. Harlof ............................. XVII, 1
Ferrite core logic in all-magnetic technique, U. Holken ........................................ XVII, 2
New components for ferroresonant circuits, M. Alique, J. L. Lloret, I. Santos and M. A. Eced XVII, 3
Hydraulic and pneumatic switching elements, H. H. Glaettli ................................ XVII, 4

18 System Design

A very small electronic digital computer with stored program control, H. Gumin and F. K. Kroos XVIII, 1
The central control unit of the ATLAS computer, F. H. Sumner, G. Haley and E. C. Y. Chen XVIII, 2
On a flexible implementation of digital computer arithmetic, A. Avižienis ................ XVIII, 3
A comparative study of propagation speed-up circuits in binary arithmetic units, M. Lehman XVIII, 4
An algorithm for division, A. Svoboda .... XVIII, 5

19 System Design II
Design of a data processing system with built-in time-sharing, J. Oblonsky and A. Svoboda .... XIX, 1
An experimental system for logic design data accumulation and retrieval, R. J. Press .... XIX, 2
The KT pilot computer – A micro-programmed computer with a phototransistor fixed memory, H. Hagiwara, K. Amo, S. Matsushita and H. Yamauchi .... XIX, 3
System design of the ETL Mk-6 computer, S. Takahashi, H. Nishino, K. Yoshihiro and K. Fuchi .... XIX, 4
Design of an arithmetic unit incorporating a nesting store, R. H. Allmark and J. R. Lucking .... XIX, 5
Modern programming methods and problems and their influence on the design of computing instruments, I. O. Kerner .... XIX, 6

20 Switching Theory
Application of a finite set covering theorem to the simplification of Boolean function expressions, M. B. Wells .... XX, 1
Digital filters of threshold elements, G. Hotz .... XX, 2
Threshold logic with one or more than one threshold, P. Ercoli and L. Mercurio .... XX, 3
Some theorems useful in threshold logic for enumerating Boolean functions, E. Goto and H. Takahasi XX, 4

Data Processing in English Banks

R. Hindle
Martinus Bank Ltd, Liverpool, UK

The English banking system is based on settlement by means of cheques, with limited credit transfer facilities. More recently, a full credit clearing system has been instituted, complementary to the cheque system. The cheques and other vouchers written by the customers pass through the system; they are not substituted by internally produced documents.

The eleven London Clearing Banks have jointly decided to adopt the E13B character form as the code for automatic voucher reading, and have specified permissible form design and sizes. Automatic reader-sorters are already in use experimentally in the banks.

Individual banks have had to decide for themselves how to introduce automatic data processing into their internal systems. This problem is complicated by the wide decentralization of the system over a network of quite small branches spread throughout the country with consequent difficulties of communication.

A number of computer installations are now in use on a limited scale. The paper describes the methods adopted and deals with some of the minor problems (e.g. the problems of printing E13B characters, and the changes affecting the customers).
The paper attempts to show how a relatively small business can make use of a medium size computer, the GE 225. The information processing and decision making system is developed to fit the specific needs of the business. These needs arise from the large variety both of products and of kinds of customer, while geography and the competitive requirement of product availability contribute to the problem. The most economical system turns out to be one that is highly centralized, wherein the lower level decisions are programmed and all major decisions are upgraded and made by those who are best informed.

The development toward this ideal is illustrated, with particular emphasis on the inventory control and production scheduling problem. Reporting for managerial measurement and control is outlined. Some unresolved problems are sketched, suggesting that before long the computer load will be significantly greater than the present single shift. By this process of centralizing decision making and its attendant data processing, a small business with fewer than 1400 employees can justify as half-million dollar computer installation.

Mathematical Analyses of Merge-Sorting Techniques

W. C. CARTER
Internation Business Machines Corp., Poughkeepsie, NY, USA

Since von Neumann and Goldstine devised and analyzed balanced methods of merge-sorting on an electronic computer, the programming of such routines has been widely discussed. Some unbalanced methods have been proposed and programmed recently, and direct computation of special cases has shown their potential efficiency. Other unbalanced techniques cannot be directly compared by computation.

These unbalanced methods are analyzed, others are proposed and the efficiencies of all sorting methods are compared. The analysis shows that both the lengths and number of ordered subsequences satisfy difference equations. The asymptotic approximation to the solution of these equations is shown to converge quickly, to be easily computable, and to determine accurately the relative sorting efficiencies.

The analysis shows that if backward reading is available and less than eight tapes are used, unbalanced polyphase sorting is best; if eight tapes are used, an unbalanced method proposed here is best. If only forward reading is available and less than eight tapes are used, polyphase sorting is best; if eight tapes are used, balanced sorting is best. Finally, the analysis is extended to determine good methods of sorting when large-capacity random-access devices are used as secondary storage.
Some Airline Applications of Monte-Carlo System Simulations

J. P. JEANNIOT and P. J. SANDIFORD
Trans-Canada Air Lines/Air Canada, Montreal, Canada

Four examples of Monte Carlo system simulations using medium size digital computers are described in general terms. These illustrate a range of usefulness from system design to continuous planning and budgeting.

The particular examples discussed are:
1) The simulation of a proposed automatic reservation system comprising central computer, communication network and field equipment. The object was to test the adequacy of proposed operating specifications before construction of the system. Various system layouts were compared.
2) The simulation of the usage, supply and overhaul of repairable aircraft units. The object was to provide managers with an experimental device upon which operating policies could be tested.
3) The simulation of a telephone reservations office. The object was to provide managers with a tool for planning labor and equipment facilities.
4) The simulation of the immediate future events to be expected in the use, repair and overhaul of aircraft engines. The object was to provide managers with regular estimates of the range of possible future events to assist their budgeting and production planning.

The authors conclude with a discussion of the precautions to be observed if such simulations are to be successful, the costs entailed, and an evaluation of their worth to managers.

The Construction of Class-Teacher Time-Tables

C. C. GOTLIEB
Computation Centre, University of Toronto, Canada

A method suitable for constructing class-teacher time-tables on an electronic computer is described. The input is a Requirements matrix \( R \), with elements \( \tilde{r}_{ij} \), listing the number of hours per week teacher \( i \) meets class \( j \). The solution is given by a set of Scheduling matrices \( \mathbf{S}^k \), with elements \( \tilde{s}_{ij}^k \) which are 1 or 0 according as teacher \( i \) does, or does not, meet class \( j \) at hour \( k \), and which satisfy uniqueness conditions

\[
\sum_i \tilde{s}_{ij}^k \leq 1 \quad \text{and} \quad \sum_j \tilde{s}_{ij}^k \leq 1.
\]

By defining certain Availability vectors and matrices, a necessary and sufficient condition for next stage feasibility is first proved, making it possible to satisfy the requirements of any one teacher or class. This condition is then extended by an algorithm which redefines the Availability recursively. The effect of the recursion is to restrict the Availabilities by removing hours which do not allow the original conditions to be met. The Availabilities eventually converge and it is believed that the final matrices contain all the feasible time-tables.
Extending Management Capability by Electronic Computers

A. VAZSONYI
Thompson Ramo Wooldridge Inc., Canoga Park, Cal., USA

Today's manager is confronted with a paradox: within his organization there are giant brains, but he still makes daily decisions without an adequate knowledge of the facts of his enterprise. We believe that within a few years, techniques will emerge permitting a more efficient distribution of the task of decision making between man and machine; and that thereby, executive capability will be significantly extended.

In this paper a Man-Machine Communication Device is discussed which enables a manager to use a digital computer as a problem solving tool. With the aid of a special keyboard he can direct a computer to execute routines, and the machine can immediately respond by displaying visual answers.

The management decision process consists of a double looped (real-time) man-machine dialogue. The manager establishes status by a question and answer game. He also evaluates alternatives by making trial decisions, and by instructing the computer to carry out sub-optimization processes.

As an illustration, the Program Evaluation and Review Technique (PERT) used in the US Defence Department is discussed. It is shown that with the proposed man-machine system, the manager could examine a wide panorama of sub-optimum alternatives and arrive at a better decision than is possible today.

On a Modification of the QD-Algorithm with Graeffe-Type Convergence

H. RUTISHAUSER
Eidgenössische Technische Hochschule, Zürich, Switzerland

The quotient-difference algorithm, which is a method for finding the poles of a finite continued fraction, or the roots of tridiagonal matrices, has been described extensively elsewhere.

The present paper shows how the quotient-difference algorithm can be modified to yield quadratic convergence in the sense of Graeffe, that is, quadratic convergence for all roots simultaneously. This is done by constructing, for a given tridiagonal matrix $A$, another tridiagonal matrix which is similar to $A^2$, and iterating this process.

Usually the very large and very small numbers occurring in the Graeffe process lead to difficulties even with floating-point computers, but it is easy to modify the process so that no large numbers can occur.

It is worthwhile mentioning that at the same time, the process gives the decomposition of a rational function (expressed as a finite continued fraction) into partial fractions, and thus provides a simple solution to the problem of exponential interpolation.
Some Non-linear Iterative Methods for Solving Systems of Linear Equations

N. GASTINEL
Université de Grenoble, France

The paper deals with the following topics:

1. General norms-Methods of definition.
   Decomposition of a norm.
2. To each bounded decomposition of a general norm there corresponds an iterative method for solving a system of first degree equations:

   \[ Ax = b \]

   The rate of convergence is directly related to the generalized conditioning number of the matrix \( A \).
3. Examples which use classical norms:
   a) method of relaxation by projection,
   b) method of steepest descent,
   c) the correlative method of relaxation.
4. Some applications to the matrices which occur in finite difference methods for the solution of partial differential equations.

Interpolation Polynomials of Square Matrices with Matrix Coefficients, and Iterative Methods for the Numerical Solution of Equations in Square Matrices of Arbitrary Form

A. KORGANOFF
Compagnie de Machines Bull, Paris, France.

The matrix functions are defined in terms of convergent developments in integer series with matrix coefficients:

\[ f(x) = a_0 + \sum_{q=1}^{\alpha} \sum_{i_q} a_i^{[q]} \times b_i^{[q]} \times \ldots \times b_{i_{q-1}}^{[q]} \times c_i^{[q]} \]

The iterative methods for the numerical resolution of the equation \( f(x) = 0 \), based upon the development of the reciprocal function around the solution, when it exists and is unique in a neighbourhood, are implicit and of the type:

\[ \sum_{4} u^{(l)}(x_t) x_{t+1} v^{(l)}(x_t) = \varphi(x_t), \]

where \( u, v \) and \( \varphi \) are also functions of \( f(x) \) and its matrix derivatives. These are extrapolation methods which become explicit in the case of mutually commutative initial values and parameters.

On the other hand, it is particularly in this latter case of commutativity, that one can usefully extend the resolution methods of scalar equations based on interpolation to the matrix equations. The iterations are then explicit and depend uniquely upon \( x \) and \( f(x) \).

In both cases—extrapolation and interpolation, round-off errors cause the commutativity to degenerate with a certain amplitude, and this must be corrected when it affects the convergence too badly.
A Method for Solving Simultaneous Polynomial Equations

P. H. BLUNDELL
The University, Sheffield, UK

Consider the equations
\[ \sum a_{ij}x^i y^j = 0; \quad \sum b_{ij}x^i y^j = 0; \quad i, j > 0; \quad i + j < n. \]

A polynomial in one variable is found by solving linear equations whose coefficients are sets of points from a mesh which satisfies an associated pair of difference equations. The roots of this derived polynomial are the values of one of the variables at all the roots of the original equations. The corresponding values of the other variables are then given by simple formulae.

Practical cases have been studied with \( n = 5 \) and \( n = 7 \). For \( n = 5 \) and with small integers for \( a_{ij}, b_{ij} \), there were no difficulties.

With more varied values, a few roots dominated the difference equations so that the linear equations for the derived polynomial became singular to the accuracy retained. The dominant roots were therefore extracted, refined, and then eliminated from the process, by forming a particular solution of the difference equations. The other roots could then be extracted. For \( n = 7 \), the solution of the difference equations was extended until some roots were dominant, since the computer available was too small to solve the linear equations for the complete derived polynomial.

There is at present no reason to expect new difficulties with more variables present.

Strategy for Multidimensional Neutron Group Diffusion Computations

E. L. WACHSPRESS
Knolls Atomic Power Laboratory, Schenectady, NY, USA

In nuclear reactor design computations, one often solves the neutron group diffusion differential equations in two and three space dimensions. Finite difference techniques are used to reduce these differential equations to large systems of simultaneous linear equations. The basic statics problem is that of finding the fundamental neutron flux mode and its eigenvalue.

Two distinct iterations are performed simultaneously, one on the fission neutron source (outer iteration) and the other on group fluxes (inner iteration) corresponding to a fixed fission source. The inter-relationship between these iterations is complex, and the optimization of strategy is a challenging task.

Successive over-relaxation or alternating-direction-implicit iteration is used for the inner iteration. Chebyshev extrapolation or Wielandt's fractional iteration is used to accelerate convergence of the outer iteration. Theoretical optimization of iteration parameters is possible for certain model problems, and such analysis guides strategy for more complex situations.

Wielandt's method enhances outer iteration convergence at the expense of inner iteration convergence to give a net increase in computer time when successive over-relaxation is used for the inner iteration. On the other hand, Wielandt's method, together with alternating-direction-implicit inner iteration, can lead to a significant reduction in computation time.
On Some Methods for Computing the Roots of Polynomials

J. L. HOWLAND
Computing Centre, University of Ottawa, Canada

Three methods for computing the roots of polynomial equations are described and illustrated. These methods are derived by writing the polynomials as symmetric determinantal equations of the form $\det(A - \lambda B) = 0$, and applying three methods due to S. H. Crandall to the associated characteristic value problem $Ax = \lambda Bx$. The matrices $A$ and $B$ are the direct sums of matrices of the forms

$$
\begin{pmatrix}
0 & -1 \\
-1 & b_j
\end{pmatrix}
\begin{pmatrix}
b_j & -1 \\
-1 & 0
\end{pmatrix}
$$

whence

$$
\det(A - \lambda B) = \sum_{i=1}^{n} b_j \lambda^{i-1} - \lambda^n.
$$

This simple representation permits the solution of linear systems of the form $(A - \lambda B)x = r$ by simple recurrence formulae which, as $\lambda$ approaches a characteristic value, yield characteristic or principal vectors.

Numerical results indicate that the choice $r = B \sum x_i$, expressible in terms of the coefficients $b_i$, is effective as a starting vector. As each root is found, the starting vector may be modified by subtraction of the corresponding $Bx_i$, this minimizing the possibility of convergence to the same root again.

Multiple roots, characterized by the conditions $(x A x) = (x B x) = 0$, are handled by evaluation of the associated principal vectors $y^{(k)}$ and formation of the bilinear forms $(x A y^{(k)})$ and $(x B y^{(k)})$.

The behaviour of these methods is independent of whether real or complex arithmetic is used.
Numerical Studies of Implicit Iterative Methods for Solving Elliptic Difference Equations

D. J. EVANS
Computing Machine Laboratory, Manchester University, UK

The solution of large order systems of difference equations, arising from 5-point finite difference approximations of elliptic partial differential equations in two space variables, can be very time consuming even on modern high speed digital computers. For large numbers of mesh points, numerical results have indicated an advantage in favour of an implicit alternating direction method, although in the case of non-rectangular regions, the convergence becomes less rapid and weaker results have to suffice.

When the finite difference equations are expressed in implicit form, with the line over-relaxation factor incorporated as a parameter, and the mesh points grouped as along the columns or rows of the network, an alternative to the successive block over-relaxation method can be realized. Furthermore, when a similar procedure is applied alternately to the rows and columns of the network with a single iteration parameter, two methods are obtained which can be considered as variants of the implicit alternating direction methods.

Five standard regions have been investigated for different mesh sizes, and extremely promising results have been obtained. The initial conclusions are that improvements have been achieved for both the non-rectangular and rectangular regions of the model problem. Further improvements for these methods are indicated.

Numerical Calculation of Shock Waves

L. GUERRI
CETIS, Euratom, Ispra, Italy

The paper is a report on the tests, carried out to solve numerically the system of partial differential equations which determine the cylindrical or spherical flow of a fluid, given a discontinuity (shock) in the initial distribution of velocity, pressure and density.

The time derivatives are approximated by forward differences and the radial derivatives by central differences. The shock is taken care of by the introduction of a pseudo-viscosity term according to the ideas of von Neumann and Richtmyer. The finite difference scheme so defined is very simple and well suited for automatic computers. The characteristic directions, which depend on the solution, are easily determined.

The convergence and stability of the finite difference scheme is proved under the hypothesis of a smooth flow. However, the numerical tests carried out on an IBM 7090 have proved the scheme to be stable even with presence of very strong shocks.

The influence of the pseudo-viscosity term on the solution is examined and is found not to vary noticeably with the form or intensity of the term itself.

The choice of the mesh width must also be taken into consideration, since under certain circumstances it can introduce small perturbations into the solution which are however, smoothed out as the solution itself is advanced in time.
A Mathematical Model of Drug Distribution and the Solution of Differential-Difference Equations

B. KOTKIN
Rand Corporation, Santa Monica, Cal., USA.

A mathematical model of drug distribution is used to study the time course of the concentration of a drug injected into a body, and of the concentration of the compounds formed in the various organs. This leads to a system of differential-difference equations which arise from the time delays inherent in the physical process of the recirculating blood. The solution of such a system presents computational difficulties not encountered in ordinary differential equations, particularly in its demands on machine storage. The paper describes some digital computer parameter experiments, designed to study a two-organ model as a guide to the biologist in selecting drugs and injection procedures for optimal effect. A new method for the computational solution of differential-difference equations is suggested.

Recent Improvements in Integration Methods for Differential Equations

J. KUNTZMANN
Laboratoire de Calcul, Université de Grenoble, France

The paper makes two suggestions for new methods of integration:

1) Using an earlier idea of F. Ceschino, P. J. Laurent suggests formulae of the Runge-Kutta type for the system \( x'' = X(x, t) \), but using \( q + 1 \) values of \( X \) in the calculation of \( x'_{t+1} \). This is made possible by the fact that \( X \) does not contain \( x' \), and gives methods which, for the same order allow a gain of one Hörner over the classical methods.

2) Following on the work of J. B. Nugeyre, J. Siret has studied formulae of the Runge-Kutta type but with backward references. For the system \( x' = X(x, t) \) these take the form

\[
x_i, \alpha = \sum_{m=0}^{l} Q_{\alpha, m} x_{i-m} + h \sum_{\beta=0}^{s-1} A_{\alpha, \beta} x_{i+1} - 1
\]

These formulae reduce to that of Runge-Kutta when \( l = 0 \) and to that of Euler when \( q = 1 \). Stable formulae exist for

\[
q + l = 2, 3, 4 \text{ and } 5
\]

where the order is \( q + l \).

The cost of these formulae is \( q \) Hörner per step. One object of the research was to compare the errors which arise in a single step, when using different methods with the same cost.

Simple examples are given of the use of the various methods.
To Find the Solutions of Convolution Equation

J. ARSAC
Observatoire de Paris, Meudon, France

The aim is to find a function which is real, bounded, non-negative and of compact support, given the square of the modulus of its Fourier transform on a compact set. This problem is equivalent to finding the solutions of the Convolution Equation

\[ f(x) \ast f(-x) = b(x) \]

\( f(x) \) can be approximated by the sum of a finite number of translations of a single function, and we can develop a finite process for solving the problem by finding the zeros of a polynomial. This method is, however, very long and gives many solutions, not all of which are acceptable. Amongst these, there is only one for which the Fourier transform is analytic in a half-plane, and the application of a Hilbert transformation allows the calculation of the phase of this transform from its modulus, which is given. From this, \( f(x) \) can be found, either by a Fourier transform or directly. This solution is very stable and its calculation is very simple.

We try to find a solution which is as close as possible to an even function, by refining a first approximation obtained by taking the modulus with an appropriate sign as the real part of the Fourier transform.

There are various ways of performing these iterations, and tests are being carried out on a standard IBM 650 to find which is the most efficient.

A Difference Method for the Approximate Solution of the Initial Value Problem for Systems of Quasi-Linear Partial Differential Equations of the First Order

R. ALBRECHT and W. URICH
Rechenzentrum, Technische Hochschule, Munich, Germany

A difference method is described for the approximate solution of the initial value problem for a system of \( m \) quasi-linear partial differential equations of the first order in \( m \) dependent and \( n+1 \) independent real variables. The method is based upon the fact that in each equation the partial derivatives of the same dependent variable define a directional derivative in a direction specified by the coefficients. This gives a system of ordinary differential equations, the integral curves of which are called quasi-characteristics. The quasi-characteristics through a given point are approximated by straight lines, and the solution at this point is found by solving a system of linear algebraic equations. The quasi-characteristics are not invariants of the system and they exist, even if the usual characteristics do not.

For numerical computation, the initial values are assumed to be known on a set of mesh points of an initial hyperplane \( H_0 \) and a corresponding lattice is considered in a parallel hyperplane \( H_1 \). At certain points of this lattice an approximate solution is calculated and these values are used as initial values to find the approximate solution in the next hyperplane \( H_2 \), and so on. A technique is described for the optimal use of storage space, and numerical results are discussed.
The Use of Approximation Methods in Linear Programming

J. HABR
Research Institute for the Economics of the Food Industry, Prague, CSSR

The Simplex method, used in conjunction with a computer represents an ideal instrument for the solution of problems in linear programming. However, such a happy combination does not always occur in practice. A mathematical model appears rather abstract to the non-specialist and the use of algorithms is somewhat mysterious. One way to popularize linear programming among practical economists may be found in approximation methods.

With these methods the computation itself is not difficult, and since they are based on simple logical considerations they allow a direct economic interpretation at any point of the computation. Thus, they may be said to enhance trust in mathematical methods.

Experience has shown that the results achieved with approximation methods are sufficiently attractive for practical use. The very inaccuracy of the data makes the expectation of an exact mathematical optimum illusory.

Many practical problems in linear programming may be solved by means of approximation methods even without an electronic computer. In this way, computer capacity may be freed for more intricate problems. Furthermore, approximate solutions may serve as initial steps for exact methods.

The following two approximation methods are discussed:

1) The bottleneck method which is a substitute for the Simplex method.
2) The frequency method which can be used for solving the transportation problem.

Application of the Steepest Ascent Method to Convex Programming

T. PIETRZYKOWSKI
Institute for Mathematical Machines, Polish Academy of Science, Warsaw, Poland

The paper describes an approximate method for finding the conditional maximum of the $n$-dimensional function $f(p)$ ($p \in E^n$), on the convex set $W \subset E$, determined by the system of $m$ inequalities $\phi_i(p) \geq 0$ ($p \in E^n, i = 1, \ldots, m$). The method reduces to finding the unconditional maximum of the function

$$G_\mu(p) = \mu f(p) + \frac{1}{\mu} \sum_{i=1}^{m} S_n(\phi_i(p))\phi_i^2(p),$$

where $S_n(x) = 0$ for $x \geq 0$, and $S_n(x) = -1$ for $x < 0$. It is shown that for $f$ and $\phi_i$ satisfying certain simple conditions, the following theorem is true: For each $\epsilon > 0$, there exists $x > 0$, such that for $0 < \mu < x$, the distance between the maximum of the function $G_\mu$ and the set of conditional maxima is less than $\epsilon$. The algorithm for the iterative maximization of the function $G_\mu$ by means of the steepest ascent method is given, and so are the necessary conditions for the convergence of the method. In the case of linear or quadratic programming, the formulae for computing the separate terms of the iteration sequence are given. The construction of this sequence may be programmed so that the transformation for one step of the iteration requires:

1. for linear programming: about $2nm$ multiplications, $2nm$ additions, and $m$ divisions.
2. for quadratic programming: about $(n^2 + 5n)m$ multiplications, $(n^2 + 6n)m$ additions, and $m$ divisions; also $m$ square roots, and one root of a third degree equation.
A Breakpoint Technique for Network Problems
J. M. BENNETT
Adolph Basser Computing Laboratory, University of Sydney, Australia

The admittance matrix of a linear electrical network can be expressed in the form $C^T Y C$ where $C$ is a connection matrix and $Y$ a diagonal matrix (when no mutual impedance exists) giving the admittances of individual elements. This can be expressed as $\sum_i Y_{ii} C_i^T C_i$.

Thus the addition of an individual element has the effect of adding a dyad to the matrix.

Suppose the admittance matrix is factorised into triangular factors $L, L^T$. Then the effect on $L$ of adding a dyad, can be computed in about $n^2$ operations (multiplications and additions to a total) for an $n \times n$ admittance matrix.

It is thus possible to increase proportionately from zero, all impressed currents in a resistive network, with capacity limits on the branches. Every time a branch current reaches capacity, this branch can be removed from the network, and all remaining capacities revised. The process can be continued until the final flow limit is reached.

The method is particularly useful when the admittance matrix is a band matrix. If the band width is $2b + 1$, then the revised triangular matrix can be computed in at most $2nb$ operations.

The process is applicable to a wide variety of problems which can be formulated as network problems—including, for example, the computation of collapse loads for engineering structures.

Application of Integer Linear Programming to a Split Problem
F. GENUYS
Compagnie IBM France, Paris, France

The problem for minimizing trim in some split problems, leads to the solution of a linear program in which the number of columns is considerable. In fact, each one corresponds to a split pattern.

As all these columns cannot be written explicitly, one has to form them as the iterations of the Simplex method proceed, using Dantzig’s classical criterion. Thus, the auxiliary problem to which one has to go back on each iteration, is an integer linear program.

The results obtained with such a code are described, using an algorithm of Gomory to solve the integer linear programs.
The Control of Traffic Signals with an Electronic Computer — 
a New Application of Real-Time Data Processing

L. CASCIATO
Traffic Research Corporation Ltd., Toronto, Canada

One of the newest applications of computers to real-time data processing is in the field of vehicular traffic signal control. In Metropolitan Toronto the search for a very versatile traffic signal system led to the experimental installation of a computer for the centralized control of a group of traffic signals in a test area within the city. The computer, connected to remote traffic detectors and signal controllers, is guided by a master control program which examines traffic data and switches the signals to best suit traffic needs. As control proceeds, the recorded traffic data can be analyzed to produce close estimates of vehicular delay and congestion. Thus, the same computer which provides very flexible signal control also serves to evaluate its own performance and to improve the effectiveness of automatic control.

The computer-controlled traffic signal system was tested for over a year. The results proved that the system was both reliable and effective. A full-scale system is now planned for the control of all the traffic signals in Metropolitan Toronto.

Utilization of an Analogue-to-Digital Linkage System in a 
Big Scientific Computing Centre

A. DEBROUX, G. P. DEL BIGIO, A. GAZZANO, 
C. GREEN, H. d'HOOP, A. RIOTTE and A. VAN WAUWE
CETIS, Euratom, Ispra, Italy

The simultaneous existence of a big digital machine and an analogue installation in the same computing centre has led the teams of specialists in both fields of electronic computation at the CETIS Centre of Ispra, to develop an integrated organisation linking the two systems.

A static linkage is achieved by means of a code, called APACHE which uses the digital computer as input to the analogue, to make accessible to the latter the auto-programming techniques which have been so successful in the digital field.

All data are introduced into the digital computer in a clear mathematical language and in terms of physical units only. The scaling and checking procedures are handled by the digital computer, which outputs a list and a set of cards for setting up the potentiometers, static check readings, etc. for the analogue computer.

The APACHE Code also allows the introduction of functions and subroutines, in the usual digital computer sense, into analogue programming.

Complementary to the above, a Dynamic Linkage has been devised which makes use of electronic analogue-to-digital (and inverse) converters, to achieve combined computation. Thus adding continuous memory to the analogue machine, and allowing a more sophisticated organisation than the pure parallel scheme. An asynchronous conversion system has been studied to obtain an economic use of digital computing time.
Met-Watch. A Technique for Processing and Scanning Meteorological Data with a Digital Computer

R. B. STAUFFER
United Aircraft Corp., East Hartford, Conn., USA

T. H. LEWIS
3rd. Weather Wing, Offutt AFB, Neb., USA

A method of processing meteorological information is described, with emphasis on scanning a portion of the data to obtain early warning of weather conditions which present a possible hazard to aviation safety. Twenty million characters of data from world-wide sources, transmitted daily to a central location over wire communications, are collated, sorted, edited and re-formatted prior to use. From this, pre-selected information is extracted to monitor weather conditions.

The monitoring process uses a master table of 13000 entries, of about 30 digits each, giving the safe operating limits for weather conditions: — minimum cloud height, visibility and wind velocity. From this table, approximately 150 stations are selected daily for real time Met-Watch processing. The observations for the selected stations are compared against these standards, and against the corresponding values in forecasts. If dangerous conditions exist, a signal is issued to a human monitor so that safety precautions can be initiated.

The interface equipment used between the communication lines and the computer is described. Special features of the data processing and Met-Watch programs are discussed, and so is the storage which forms a link between the Met-Watch system and a large digital computer employed for automatic weather analysis and prediction.

A New Method for Economic Load Dispatching in Power Systems

J. CARPENTIER
Electricité de France, Paris, France

The object of economic load dispatching in power systems is to arrange the output from the power stations to match the load on the system and to minimize the production costs.

The method described gives, for the first time, a full and exact solution to the problem. It takes into account all the physical restrictions on the system and determines the voltages and reactive VAs which give the lowest cost.

The method applies the theorem of Kuhn and Tucker and involves non-linear programming. An algorithm is given for finding an optimum solution in a very short computing time.

The method will be put into practice when a digital computer is installed in the National Dispatching Office of l'Electricité de France. The computer will receive information from the network through telemetering equipment and will schedule the output for the following day. This will present a practical example of system optimization using a digital computer working effectively in real-time.
The Multi-List System for Real-Time Storage and Retrieval

N. S. PRYWES and H. J. GRAY

Moore School of Electrical Engineering, Philadelphia, Pa, USA

The object of the research was to increase efficiency by several orders of magnitude, in terms of speed of processing and of cost, and to ease communication with the computer. The system conceived is in the medium size class.

The following techniques are used to obtain these ends:

a) The use of an associative memory for storing, deleting and reading information without addressing;

b) A hierarchy of memories varying in speed and capacity;

c) Processor organisation and timing to minimize instruction retrieval and housekeeping routine times;

d) Processor list type instructions, where a single instruction processes an item of data of varying length;

e) Automatic retrieval of programs by name, thus allowing a greater vocabulary and easing communication with the computer.

The complexity of requests made of an associative memory leads to the use of an addressable memory in which the associative memory is simulated using Multi-List techniques.

The general organization of an arbitrary information retrieval file into the Multi-List uses balanced trees, and stratification of descriptor language. The analysis of an information retrieval problem into such a structure is not always possible by human means and must be mechanized. An algorithm is given for the construction of balanced trees, and processes are described for the analysis of file data into exclusive (non-overlapping) groups.

SYNTOL (Syntagmatic Organization Language)
Its Properties and Applications in Automatic Documentation

J. C. GARDIN

Centre National de la Recherche Scientifique, Paris, France

SYNTOL is a linguistic system for information storage and retrieval. It implies the use of a hierarchically organized vocabulary but does not impose any particular one. Moreover, recorded sentences themselves provide syntagmatic associations, which the machine can use in addition to the paradigmatic relations found in the vocabulary, when processing a given question.

These sentences are chains of syntagms, each composed of two terms connected by an oriented relation, $R_n$ ($n = 4$). The meaning of each relation can be specified, when necessary, by operators, attached to one of the poles of a syntagm. Finally, in order to achieve a univocal syntactical reduction, each word in the vocabulary is ascribed to one of four quasi-grammatical categories, which command permissible constructions.

The analysis of a document therefore, contains two kinds of term: isolated words, belonging to the normalized vocabulary, and syntagms, expressing logical relationships between such words. Variations, in the analysis are defined, ranging from a mere enumeration of key-words to the construction of full sentences, in order to test their respective efficiency for information retrieval.

Rules are given for modulations, i.e. controlled changes in the lexical or syntactical features of a given question, in order to reduce or increase the range of expected answers.
A Method for using Computers in Information Classification

R. M. NEEDHAM
Cambridge Language Research Unit, Cambridge, UK

The paper describes now a difficulty encountered in semi-mechanised information retrieval experiments led to work on computer techniques for the classification and grouping of large bodies of data. A definition of a closely associated group or clump of objects is given, and algorithms are described for finding these groups. The programs used on the Cambridge University Computer EDSAC II are discussed, and results are described for two experiments, one on material from the original information retrieval work, and one on linguistic material. The paper concludes with a section on the problems to be expected if the procedures are to be usable on a very large scale, and makes suggestions for overcoming them.

Interrogating a Computer in Natural Language

D. R. SWANSON

R.W. Division, Thompson Ramo Wooldridge Inc., Canoga Park, Cal., USA

This paper describes experiments to measure the effectiveness of searching natural language physics texts by computer. A retrieval question, expressed in natural language, is first presented to the computer. The computer transforms this question into a search instruction composed of word and phrase combinations, with synonyms and near-synonyms taken into account, by means of an automatic thesaurus look-up.

The search instruction thus formulated, is then used by the computer as a basis for searching the full text of a collection of physics articles. The results are presented in a printed list of responsive articles, ordered on a scale of relevance with respect to each of the retrieval questions.

Independent relevance estimates based on human judgment are used to evaluate the accuracy of the computer process. In an earlier paper, the author described a set of similar experiments, which differed in that the transformation of the original natural language question into a computer search instruction was a human rather than a machine process. The retrieval results thus obtained are compared with those of the work reported here.

These experiments provide insight into the syntactic and semantic problems of automatic indexing and information retrieval.
Multiple-Path Syntactic Analyzer

S. KUNO and A. G. OETTINGER
Harvard Computation Laboratory, Cambridge, Mass., USA

Multiple analyses of syntactically ambiguous sentences have, for the first time, been effectively and economically realized, by a new extension of the method of predictive syntactic analysis. Branchings caused by homography (membership of a given word form in more than one syntactic word class), and by multiple functions of a given word class are followed in a systematic loop-free sequence in which each partial path is traversed once only. Different paths that reach the last word in a sentence, correspond to different acceptable syntactic structures of the sentence.

The prediction pool for this method is a pushdown store in the strict sense; the topmost prediction in the pool is matched against the class of the next word form of the sentence, by look-up in an internally stored table of grammatical rules, whose content may be varied at will without affecting the program for the analysis algorithm.

Satisfactory results have been obtained with programs for the analysis of English. The basic principles of the new method offer a convenient framework for the development of powerful techniques for the syntactic analysis, not only of English, but also of Russian. The availability of alternative acceptable syntactic structures clarifies some of the issues lumped under the heading of semantic ambiguity.

On Some Axiomatic Systems for Formal Grammars and Languages

K. ČULÍK
Mathematical Institute of the Czechoslovak Academy of Science, Prague, CSSR

A semi-algorithm is essentially the normal algorithm in the sense of A. A. Markoff, or the combinatorial system in the sense of M. Davis, and is determined by elementary symbols (words), deductive rules and selection conditions (grammatical rules). A semi-algorithm is a mapping, which joins the sets of strings of elementary symbols (languages) to the strings of elementary symbols (sentences). The elementary symbols themselves may be strings of other symbols (letters).

All the languages generated by phrase structure grammars of N. Chomsky may be generated by some class of semi-algorithm and conversely. Analysis of the axiomatic system of Chomsky proves that axioms 1–3 are unnecessary.

Analysis of the new model of syntactic description, of A. F. Parker-Rhodes, shows that it is not clear whether it is possible to introduce formally the notions of segment, substituent and paradigm, in such a way that the system of all paradigms is a lattice.
Rules of Interpretation — An Approach to the Problem of Computation in the Semantics of Natural Language

M. KAY
University of California, Berkeley, Cal., USA

This paper distinguishes sharply between rules of formation, hitherto the principal interest of descriptive linguists, and rules of interpretation which belong partly to semantics and partly to syntax. A model is proposed in which the learning of a new word is viewed as a process whereby the grammatical function as well as a semantic index for the word are entered against it in an inventory stored inside the learner. The semantic index of a phrase is then constructed, as required, from the semantic indices of the constituents.

A computer program is outlined which will use rules of interpretation to associate words and phrases with positions in a net which will be grown piecemeal to accommodate incoming data. The computer will be able to assemble and select from its data in different ways. Eventually, it is hoped that the net may be usable as a dictionary for mechanical translation; a dictionary which will be able to provide a phrase in the target language to represent a word in the source language, and vice versa, without the phrase being stored explicitly in the machine and without the necessity for the dictionary maker to have foreseen the particular use.

Machine Translation and/or an International Language?

K. G. SELLIN
Regnecentralen, Copenhagen, Denmark

In this paper the question is raised whether M.T. should be regarded as a final solution of the language barrier problem. The author emphasizes that research into a common international language should be carried on at the same time. The paper introduces a proposal for such a logical computer-related language, rid of most of the inconsistent and irrational constructions with which most national languages are burdened.

This language could be learned in a fraction of the time required for most national languages. It would not be politically biased, would be impartial to all nations, and could be used as a basis for the various symbolic computer languages.

Further, it is proposed that smaller countries with little economic possibility of their own M.T. project could join a project for translation into the international language. It is predicted as a future and desirable development that scientific literature would be published and read in this language without further translation. This would be not only a solution to the scientific language barrier problem, but also a rational way of communication for all computer-related sciences.
Computer-to-Computer Communication at 2.5 Megabits/sec.

N. CLARK and A. C. GANNET
Control Data Corp., Minneapolis, USA

A detailed description is given of a system which allows two or more asynchronous digital computers to communicate over a single half-duplex communication channel. Communication is provided by means of special terminal equipment which connects each computer to a communication channel with 3 Mc/sec bandwidth.

The terminal equipment provides direct access to the computer memory through program control. Parallel binary information is converted into time sequential signals for transmission on a wide-band channel. Peak rates of 2.5 million bits per second are achieved, with an average rate of 2.1 million bits per second (on variable block length transfers, up to full memory length).

Results of experimental data transmission tests are given for a standard microwave link and also for direct transmission by coaxial cable. Tests are also being carried out on a 100 mile length of standard television channel. The programs are described which are used in these tests and which enable the establishment of communication with a large number of computers on a single channel. Results of some system tests are given which demonstrate acceptably low error probabilities.

Dependence of Speech Quality on Transmitted Information Rate in a Band Compression System

E. ROTHAUSER and F. LENK
I.B.M. Austria, Vienna, Austria

The well-known band compression systems for speech transmission, e.g. the Vocoder, allow for a great reduction in capacity of the transmission channel compared with the channel capacity necessary for direct speech transmission. In Vocoder systems, only a description of the speech signal is transmitted. Usually one part of this description, in this paper called the aggregate function, corresponds to the frequency-quantized short-time energy spectrum, while the other part, the excitation-function, is determined by the pitch frequency and the character of the sounds.

While intelligibility of the transmitted speech signals depends mainly on the aggregate function, quality of speech is strongly correlated with the transmitted excitation function. The channel capacity for the transmission of the aggregate function can be reduced to very low values. Measurements in a Vocoder system, adapted for this purpose, show that for a given speech quality, the information-rate for transmission of the description of the excitation function cannot be reduced below a certain level. If speaker recognition is desired, the characteristics of the speaker must be permanently transmitted in spite of recurring redundancy, in order to secure recognition in the case of short speech signals and different speakers.
Self-Correcting Decoding Circuits

K. STEINBUCH and F. ZENDEH
Technische Hochschule, Karlsruhe, Germany

To increase the reliability of the output information of the coordinating matrix, a second matrix called the correcting matrix is introduced. The m output rows of the coordinating matrix are connected to the m input columns of the correcting matrix, while the m output rows of the latter form the inputs to the maximum detector. In special cases, a threshold discriminator may be inserted between the coordinating and the correcting matrices.

This tandem connection of the matrices permits the identification of the highest excited output rows of the coordinating matrix, even when malfunctioning precludes conventional identification. The identification process involves the utilization of the states of excitation of all the other output rows of the coordinating matrix.

The self-correcting circuit appears superior to conventional diode circuits in the following ways:

a) For \( n > 5 \) the cost of the self-correcting circuit is lower.

b) The self-correcting circuit is substantially more reliable.

c) The reliability can be increased still further and the system down-time minimized by an early-warning device indicating the loss of capacity for self-correction.

d) The maximum detector of the self-correcting circuit regenerates the output signal. On the other hand, conventional circuits need an additional amplifier stage to achieve the same result; this additional stage increases the cost and the probability of failure, thus decreasing the reliability.

Message Protection Features of the ComLogNet Program

A. E. MILLER and A. B. SHAFFRITZ
Auerbach Corporation, Philadelphia, Pa, USA

J. R. SMITH
Radio Corporation of America, Camden, NJ, USA

One of the prime requisites of a military communications system is protection of its traffic. The program design for the message-switching facilities of the Air Force Data Communications System (Phase 1 – ComLogNet) takes this prime requirement into full account. The on-line message-processing program is organized not only to process the traffic, but also to give assurance that there will be no loss of traffic due to equipment failure. This organization is illustrated by the features of the program for protecting the traffic against loss resulting from the failure of the Communications Data Processor (CDP), the controlling element of the message-switching facility. Problems associated with a possible failure of the CDP are significantly more complex than those associated with failure of other equipment in the facility.

The relevant characteristics of the ComLogNet system, the implications of the message-protection requirement, and the data flow and cyclic organization of the program are described first. The status of information available to the program subsequent to a CDP failure follows. Features embedded in the on-line program to effect the required message protection are presented, while a description of the recovery procedures concludes the paper.
Fundamentals of a Theory of Asynchronous Information Flow

C. A. PETRI
Institut für Angewandte Mathematik, Universität Bonn, Germany

A general mathematical model for information flow is developed from a set of basic assumptions and requirements. It is assumed that there is (only) one type of elementary event, formally comparable to the physical collision of two distinguishable particles in kinetic gas theory, and (only) one mode of interdependence of events; that such a mode has the same minimal complexity as an event; and that events satisfy a conservation law and a reaction principle. The theory is required to describe total information flow without presupposing further properties of events, and to relate information flow to observational or behavioral invariants.

The elementary events are interpreted as actions of switching elements, and it has already been shown that every finite automaton can be composed of a universal switching element for which no metric properties need be known. Moreover, the partial interaction of otherwise independent systems can be treated for an unlimited number of systems. This leads to the concept of multiple activation; and as an example, the logical structure of an unlimited storage device is given. Any description of this device as “passing through a sequence of states” destroys its topology, and either its physical possibility or its actual operation.

Finite and Combinatorial Automata. Turing Automata with a Programming Tape

J. BEČVÁR
Mathematical Institute, Czechoslovak Academy of Science, Prague, CSSR

A generalization of the concept of finite automata is given, based on the use of neutral symbols for input and output. Several equivalent representative types of such automata are introduced. These realize more general operators than do sequential machines.

A hierarchy of infinite, combinatorial automata with input and output is introduced, starting with the simplest and ending with automata as powerful as the general Turing machine. They all use an internal tape and differ only in the restrictions which are imposed upon the treatment of the tape by means of Turing-type operations.

A Turing machine is defined with a special tape on which different programs can be stored. It is based upon a variant of the general Turing machine which performs more elementary atomic acts than does the standard formulation.
Toward Inductive Inference Automata

L. J. FOGER
Astronautics Division, General Dynamics, San Diego, Cal., USA

Induction, the creative act which yields hypotheses, is recognized as a process of generalization wherein uncertainty, consistent with sensory experience is added to a set of descriptor-symbols. An automaton is described which continually edits an artificial language of such symbols to maximize the information content of its experience. The longevity of each symbol remains dependent upon evidence of its usefulness as indicated by its relative frequency of occurrence. The automaton operates in fast time, generating an infinitude of possible models in simulation of the probabilistic structure of its environment. Within the available time for decision, the automaton finds points on this environment-modified pseudorandom waveform which are in agreement with the most recently sensed data. In general, each of these finite models of the near future has a probabilistic structure somewhat different from the experience. These models are rank-ordered with respect to increasing disparity. The automaton examines its own response to that future environment predicted by the first-order model, translating this response into a measure of the relative information it would hold at each point. The probability of survival is directly related to the relative information possessed by the automaton about its environment; a significant decrement justifies the automaton in taking preventive action determined by reference to past experience.
Computer Simulations of a Perceptual Learning Model for Sensory Pattern Recognition, Concept Formation, and Symbol Transformation

C. VOSSLER
System Development Corporation, Santa Monica, Cal., USA

L. UHR
University of Michigan, Ann Arbor, Mich., USA

A neuron net model for the discovery, generalization and transformation of patterned information has been examined, in simplified form, by means of computer simulations. One version of the model is a sensory pattern recognition program that generates its own operators, or measures, as a function of the unknown inputs presented to it. The program attempts to find measures and to improve its methods on the basis of its successes and failures. A second program, when presented with sentences in two different languages, attempts to discover and develop its own translation procedures. Both programs are instances of a model that builds itself into a structure that reflects, in usable form, information it extracts from the external world.

The pattern recognition program has been tested for its ability to learn to recognize such patterns as handprinted and handwritten letters, Arabic handwriting, cartoon faces, photographs, abstract shapes, and degraded speech. Results are presented for these tests, and for comparisons between simulation and human subjects. Results are also presented for the language transformation program, with several different language pairs.

Finally, the model is discussed as a method for studying and automating processes of discovery and induction of the sort that seem to be needed in the development of thinking organisms.

Self-Organizing Grouping — A Learning Structure

V. KUDIELKA
IBM Austria, Vienna, Austria

Learning structures can be considered as systems which build up coordinations between inputs and outputs. Self-Organizing Grouping builds up coordinations between sets of a given number of logical variables. The assumption is made that the incoming sets have great redundancy and the aim of the Self-Organizing Grouping procedure is to assemble sets with small Hamming-distances between each other into groups, and to represent such a group by the set with the highest rate of occurrence. The input of a set with a relatively small rate of occurrence causes the output of that set which represents the group to which the input set is assembled. In order to prevent the predominance of past events, which may lead to an inflexible structure, an exponential function of forgetting is applied to the stored rates of occurrence, so that the coordinations continually adapt themselves to the actual state. A second mode of forgetting is introduced by storage limitations.

Variations of the procedure are discussed which depend on special characteristics of the input data and some results of practical applications are given.
The Development of a Conditional Probability Computer for Control Applications

H. C. RATZ and G. H. M. THOMAS
University of Saskatchewan, Saskatoon, Canada

Attempts have been made recently to apply theories of selforganizing systems to problems in automatic process optimization. If such a system is to obtain adequate statistical information about the behavior of complex processes, it must utilize not only linear relationships among the observed data, but also correlations of higher order. This principle has been applied to the processing of data in multivariable control problems, in which the output is to be optimized.

A special purpose computer has been constructed which continuously measures time-weighted frequencies of binary patterns, and from these deduces certain conditional probabilities. These probabilities provide the basis for making control decisions. The Conditional Probability Computer has been applied to the control task of optimizing a process simulated by an analogue electrical network. Experimental results show that this computer remains useful in the face of considerable interaction among the control variables. In this system, information is processed into a form that facilitates control and furthermore, the procedure provides for continuous adaptation to changing characteristics of the response surface.

Simulation of a Learning Machine for Playing GO

H. REMUS
IBM Laboratories, Boeblingen, Germany

The rules of GO are formulated in mathematical language so that a program for playing the game can be developed. A precise determination of the best move in a given situation is not possible however, because of limitations of time and storage. By applying three operators successively — a lexicon, a heuristic computer and a random number generator — that move is selected which, according to present experience, is the most favourable of those permitted.

The lexicon and heuristic computer are built up and continuously improved by the machine itself, using the success or failure of previous moves. (The success of a move is deduced from the rules for the final scoring which are valid for each position). Thus, the effectiveness of these two operators increases in the course of time analogously to learning processes; i.e. during the initial games the moves will be determined mainly by the random number generator, but subsequently and to an increasing extent, by the weighting table of the heuristic computer, and eventually by the lexicon. With the aid of examples it is shown how far the experience gained will improve the quality of the game in formerly unencountered situations.
Digital Computer Usage in Analysis of Electroencephalograph and Similar Quasi-Rhythmic Patterns

M. G. SAUNDERS
Winnipeg General Hospital, Canada

The electroencephalogram (EEG) is a multi-channel record of voltage fluctuations over different areas of the brain, each area showing a different pattern. The patterns are usually quasi-sinusoidal but can have their locations, frequencies and/or amplitudes changed by disease or varying physiological states.

EEG analysis is performed by visual inspection of the record and the significance of different patterns is indicated by experience. Although clinically adequate, visual analysis is too inaccurate to quantify changes in early disease, analyse mechanisms producing brain waves, or define mathematical models.

The large amount of data and the search for optimum analytical methods necessitate digital computer usage. To change the EEG from analog to digital form, a multi-channel converter has been built to scan the EEG record and punch computer tapes. Digitizing the record permits analysis of stored data of past experiments and adequately diagnosed patients.

Search for mathematical representation of EEG patterns has shown that high resolution Power Spectral Analysis is probably suitable. However, some patterns appear to represent sinusoidal activity modulated by noise. Evidence suggests that this noise carries important information possibly reflecting random network characteristics of the brain, and its presence must be considered in any mathematical analysis or model of the EEG.

Towards the Automation of Binocular Depth Perception

B. JULESZ
Bell Telephone Laboratories, Murray Hill, NJ, USA

A reconstruction of the three-dimensional environment from its two-dimensional projections was tried out by computer simulation. Several psychological findings, which were derived by using random stereo pairs devoid of all contextual cues except binocular parallax, were incorporated in the experiments. The applied method is free from the inherent limitations of many previous attempts (using zone-matching) because the entire presentation is processed simultaneously.

The point-by-point differences between the left and right fields are formed with all possible horizontal shifts prior to subtraction. This produces a number of difference fields in which adjacent points, whose values lie below a certain predetermined threshold, are connected into clusters. The search for these clusters, having points with identical parallax shifts, immediately gives the cross sections (contourlines) of objects in a natural way. In ambiguous cases, several strategies imitating human behavior are adopted. An attention mechanism assigns different weights to the various strategies and the selection depends on the properties of the ambiguous point domains (such as their size, connectivity to unambiguous point domains, etc.). A computer program called AUTOMAP was written along these lines. Its performance is demonstrated by several compiled contour maps.
The Phonetic Typewriter

T. SAKAI and S. DOSHITA
Kyoto University, Japan

The phonetic typewriter is a device for converting the sounds of the human voice to typed letters or other coded symbols. For the development of such equipment it is necessary to consider two processes: the automatic segmentation of continuous speech sounds, and the pattern recognition of these segmented sounds.

This paper discusses the principles involved and some experiments related to the automatic segmentation of continuous speech sounds into discrete sound-segments corresponding to phonemes. For this process two criteria of stability and distance are applied to the results of a zero-crossing analysis expressed in binary form.

Pattern recognition of speech sounds by means of distinctive features is also discussed. Logical combinations of the binary representation of the signals derived from many parallel filter circuits are used to determine phoneme classifications of the input speech.

Zero-crossing analysis is made in the $f_1$-$f_2$ domain for vowels and for consonants; analyzing channels are combined in a manner suitable for distinguishing phonemes belonging to the same phoneme group. This phonetic typewriter is flexible and is useful for processing speech sounds from other languages besides Japanese.

The system uses 3000 transistors and 5000 diodes for the speech input, phoneme classification, analyzing channels, analog-to-digital conversion, logical decision and symbol output equipment.

MUSE, A Sound Synthesizer

A. W. SLAWSON
The Mitre Corp., Bedford, Mass., USA

An IBM 7090 computer program called MUSE has been written that produces a digital representation of a sound pressure level waveform from specifications of relatively slow moving acoustic quantities. The acoustic specifications language, also called MUSE, is composed of statements that describe time functions of acoustic parameters. Each statement contains a spectrum number, a time, a fundamental frequency with an amplitude multiplier, and the frequencies and bandwidths of up to ten resonances. The system of resonances is excited by pulses at the fundamental frequency or by randomly spaced pulses. Each spectrum (there can be up to 100 spectra) consists of a series of MUSE statements with the same spectrum number ordered with respect to time. The translation process proceeds by interpolation between the values of the various parameters on successive statements. Each output waveform ordinate is the sum of the instantaneous amplitudes contributed by each resonance in each spectrum. An experiment testing the uses of synthesized speech in the operation of the computer itself is described.
An Analog-Digital Character Recognition System

M. NADLER
Cie des Machines BULL, Paris, France

A brief critique of existing approaches to character recognition is given. The author does not consider that the immediate quantization of the information supplied by the optical scanner, and subsequent processing by pulse, digital or logical techniques of the quantized raw data is efficient.

In the present system there is an acquisition unit consisting of the scanner and a special purpose analog computer. The scanning is carried out over interlaced micro-matrices, and the output of the analog computer constitutes information on the most probable orientation of the character outline in each micro-matrix. This acquisition unit is highly efficient and economical, and constitutes the core of a variety of existing and projected BULL optical character readers.

The information at the output of the acquisition unit is in coded form and permits processing in a digital logic unit, whose complexity will depend on the variety and types of characters to be read. An example is given here of the logic for the simplest case, handling numerical information only.

A BULL computer has been programmed for logic simulation, to permit the rapid design of logic units for any given set of characters.

Machine Recognition of Cursive Writing

L. D. EARNEST
The Mitre Corp., Bedford, Mass., USA

Some 10,000 of the more common English words have been categorized on the basis of salient features in their handwritten shapes. Script samples of words to be recognized are processed to determine their position and scale in both dimensions, and are categorized on the same basis as the dictionary. A list of properties is extracted from each word pattern and is compared with the corresponding dictionary entries to reduce ambiguity, sometimes yielding unique recognition, but more often a short list of words. For non-unique cases, further isolation is undertaken by attempting to synthesize and match each of the alternatives with the pattern under observation. The technique of finding the position and scale of the pattern, and its application in identifying and comparing the coordinates of key features with the dictionary entries is discussed.
ALGOL 60 Processors and a Processor Generator

M. PAUL
Johannes Gutenberg Universität, Mainz, Germany

In constructing various ALGOL 60 processors for the translation of the formal language ALGOL into the machine codes of different computers, a logically identical decoding and generating mechanism has been used. This mechanism applies in general to the translation of other formal languages, its structure being defined only by the syntactic description of the language.

Therefore, a general processor generator can be constructed, whose input is the syntactic description of the formal language in a suitable algorithmic notation, e.g. the Backus notation for ALGOL. If the syntactic description of the formal language is unique, the processor generating algorithm or a coded processor generator, produces from it a processor whose input is the formal language and whose output is a sequence of macro-instructions. Only the machine coding of the single macro-instructions depends on the semantics of the language, and will generally be different from machine to machine. At this stage, the problem of an optimal and unique syntactic and semantic description of ALGOL, possibly by refining its structure, becomes important.

An Algorithm for the Translation of ALGOL

W. M. KEESE Jr. and H. D. HUSKEY
University of California, Berkeley, Cal. USA

The paper describes a simplified version of the translator portion of a three-stage compiler, which handles virtually all ALGOL 60 statements of a static nature, not involving lists or recursiveness. The output is a machine-independent intermediate language, designed for efficient assembly into the particular language of any individual computer.

The translator is aware, at any given moment, of no more than four entities from the source language. There is neither back-up nor look-ahead. Translation occurs through the use of a push-down list of operators and associated addresses, the process being governed by a comparison of the strengths of the operators. This comparison is mainly effected by the numerical relationship of the internal representations of the operators, and usually occurs without having to determine the identity of either operator. The operation part of a command is normally generated by automatic manipulation of the responsible operator, again making it unnecessary to identify the operator.

The complete ordering, representation, and placement lists on which automation of the process depends are given and the algebraic system is developed in detail.
On table operating algorithms

L. LOMBARDI
CETIS, Euratom, Ispra, Italy.

The paper is devoted to the discussion of a complete set of data system language elements which allows for a concise representation in declarative form, of arbitrary table operations.

The first section contains a selection of recursive functions representing address arithmetic algorithms, and notation rules for the description of the structure of tables.

The second section is devoted to the discussion of a method of representing table operations as functions of conditional forms, involving one index.

The third part deals with the implementation of these language elements for Turing-von Neumann type computers.

A Proposed ALGOL 60 Matrix Scheme

S. J. M. DENISON
English Electric Co. Ltd., Stafford, UK

Programming schemes which enable matrix operations to be specified directly have been of great value, and an attempt to incorporate one into ALGOL therefore seems worthwhile. The proposed scheme uses ALGOL 60 conventions. Matrices are represented by identifiers which are used, where necessary, as parameters of procedures representing matrix operations. However, the scheme enables many operations to be specified without parameters, the relevant matrices being understood from the context. Intermediate results need not be named.

Since the scheme is in ALGOL, scalar operations can be performed among the matrix operations, and provision is made for building up matrices from the results of such operations and for extracting matrix elements for scalar manipulation.

Arrays enable partitioned matrices to be specified by a suffix notation, so that operations on such matrices are easily programmed.

The allocation of storage space can be made completely automatic, but a program involving many operations on large matrices can be made more efficient if the writer indicates when a matrix is no longer required, and a procedure for doing this is included.

Some examples, a table of suggested operations and a simplified version of the master program are given.
Program Organization and Record Keeping for Dynamic Storage Allocation

A. W. HOLT
Applied Data Research, Inc., Princeton, NJ, USA

Dynamic storage allocation refers to the process of making storage allocation decisions in the course of problem solution on the computer (in contradistinction to pre-planned storage allocation, as normally accomplished by compilers, or by programmers using machine code). We consider the moments in computation time at which allocation decisions must be made and/or applied to the re-arrangement of the contents of various storage media. These events are intimately related to another class of events; namely control passage from one program to another. Control passages may result from at least the following:

a) the logic of the problem;
b) the needs of parallel computation;
c) externally enforced and fluctuating priorities on waiting tasks.

This paper describes a logic of program organization, and a related logic of program system organization, which makes dynamic storage allocation possible (in the strong sense of economically possible) and provides centrally organized means for handling control passages resulting from any of a), b) or c) above. The aspects of program system organization to be discussed therefore overlap heavily with what is usually covered under the heading of operator system or executive program. The assumed computer environment is a large-scale binary digital computer (e.g., IBM 7090) with secondary storage such as drums or discs.

Programmed control of multi-computer systems

R. PERKINS and W. C. McGEE
Thompson Ramo Wooldridge Inc., Canoga Park, Cal., USA

This paper discusses the automatic control of multi-computer systems. These systems contain a variable number of equivalent computers in contrast to two-computer back-to-back systems, and systems consisting of a large central computer and one or more satellite computers performing peripheral tasks. The potential advantages of such systems in reliability, expandibility, flexibility, and in servicing multiple on-line users are considered.

Multi-computer systems are operated under an executive or master program which exercises dynamic cognizance of, and control over, system operation. Some of the problems encountered and the techniques developed for the programmed control of multi-computer systems are described. These include the question of distributed versus centralized system control, effects of system elements which must be shared by several programs, effects of intended system application and configuration on control program design, inter-module communication requirements, and the multi-computer system as seen by the applications programmer.

An existing multi-computer system based on a large electronic switching exchange and an operating control program are described.
Automatic Translation of Programs from One Computer to Another

ASCHER OPLER, D. FARBMAN, M. HEIT, W. KING and E. O’CONNOR

Computer Usage Company, New York, USA

R. GOLDFINGER, H. LANDOW, J. OGLE and D. SLESINGER

International Business Machines Corp., New York, USA

It has generally been considered impossible or, at best, impractical to translate automatically a program from the code of one machine to that of another. One obstacle to translation has been the absence of functional information in a machine language program. Furthermore, there are computer words with no function, with multiple function and with changing function. Once the source computer has been mapped into the target computer, the conversion of the original program (including modifications) for operation on the second machine presents many difficulties.

The automatic translator recovers the lost functional information by dynamically activating the source program and recording and analyzing its performance. From this analysis, the lost information is either recovered in usable form or information about its availability is obtained. Translation algorithms are developed for the largest practical units of the source program. Information concerning dynamic modification of instructions is obtained in the analysis phase and suitable algorithms have been devised for translating modification.

A proposed program is described for translating IBM 705 computer programs to IBM 7074 programs making intermediate use of the symbolic assembly languages of each machine.

Requirements on a Language for Logical Data Processing

P. LUCAS

IBM Austria, Vienna, Austria

The problems which are discussed in the present paper arose out of the necessity to describe algorithms in combinatorial logical algebra, sequential logical algebra and the simulation of non-numerical processes. Out of a consideration of the data and operators of these problem fields are derived the requirements on the corresponding concepts of storage and instructions.

The attempt to simulate the required properties by means of ALGOL 60, shows that only a few of the desired properties can be obtained.

As a result of the investigation of the description of logical algorithms, an algorithmic language LOGALGOL, has been defined and is described. The storage organisation differs essentially from that of ALGOL and exactly fulfils the requirement described.

ALGOL and LOGALGOL can be used simultaneously with the help of a narrow information channel established between the stores of the two languages.

The translation problems which are induced by the new storage concept of LOGALGOL are discussed. Finally, examples from the above-mentioned problem fields illustrate the results.
Nanosecond Speed in a Core Memory with Non-Destructive Read-out

J. SCHARBERT
Siemens und Halske AG, Munich, Germany

Non-destructive read-out by reversible magnetization processes in magnetic-core memories makes very high operating speed possible. In square-loop ferrites, reversible magnetization changes take place if the energizing field impulses are smaller than the coercive force, or if the duration of the impulses is short enough. In this case there are reversible magnetization changes independent of the remanent state of the cores (first type), and others which depend on the remanent state (second type). Non-destructive read-out is based on the magnetization changes of the second type. In order to exclude the effects of magnetization changes of the first type when reading, it is possible to use two cores per stored bit for cancellation.

Magnetization changes of the second type were investigated. In particular, measurements were made of the maximum induction swing obtainable and of the maximum interrogation current permissible. Empirical relations have been found for these quantities, indicating the effects of the duration of the interrogation impulse, the dimensions of the core, and the biasing current super-imposed on the interrogation impulse. These relations permit the comparison of ferrites in order to ascertain their suitability for memories with non-destructive read-out. For this mode of operation, repetition frequencies of up to about 20 Mc/sec appear feasible.

Some Problems in the Design of Magnetic Film Storage Systems Operating at Millimicrosecond Speeds

J. D. R. McQUILLAN
Royal Radar Establishment, Malvern, UK

The problems involved in designing a multiple conductor matrix for pulse transmission are discussed. Although the specific case of a megabit storage system is given detailed attention, normalized design curves are presented, in order that transmission and distortion of millimicrosecond transients in a complex matrix based on strip line geometry can be readily evaluated. The conductor mesh considered is one where transmission and detection channels can be made cheaply by printed circuitry techniques and yet offer precision, as regards performance, to millimicrosecond transients. With the current development of fast, planar storage elements of a ferromagnetic nature, the above mesh would appear to be the logical choice of network in order to use such units in a working assembly.

The analysis of the conductor mesh for a megabit storage system shows that a permissible impedance level of the order of 50 ohm is required, to satisfy a typical specification for a fast storage system.
A Read-Out Circuit for High Speed Non-Destructively Read Stores

G. H. PERRY and E. W. SHALLOW
Royal Radar Establishment, Malvern, UK

Choosing the memory elements, selection and read-out circuits for a large capacity high speed store involves both technical and financial compromises. It is the authors’ opinion that a simplified read-out circuit would have a marked influence on these compromises, and this paper reports progress towards such a circuit.

Tunnel diodes following a diode limiter replace the amplifier, gate and store register of the usual read-out circuit. A voltage is applied across a pair of series-connected tunnel diodes, sufficient to maintain only one diode in its high voltage stable state. The input is applied to the junction of these two diodes, and its polarity determines which of the diodes switches to the high voltage stable state. This provides a sensitive amplitude discriminator. Inputs only influence switching during the applied voltage rise time, thus providing gating facilities. The circuit also functions as a register, since a diode pair once switched, remains so until the voltage across the pair is removed.

Inputs of $\pm 25 \, \mu A$ into an input impedance of approximately 150 ohm give reliable operation, while from the gated to the ungated condition, the sensitivity ratio of the circuit exceeds 1000 : 1. The gate width is less than 10 nsec. Temperature stability has proved a limitation and is discussed in detail.

Tunnel Diode High-Speed Memory

S. TAKAHASHI and O. ISHII
Electrotechnical Laboratory, Tokyo, Japan

K. NAKAZAWA and K. MURATA
Hitachi Ltd., Tokyo, Japan

Modern high-speed computers tend to have small capacity high-speed memories for storing a program segment, intermediate data and indices, besides their main core storage. For such a memory the authors propose a tunnel diode matrix, each element of which consists of a tunnel diode, a resistor and an ordinary diode. The first two components form a bistable circuit and the third serves for read/write selection.

The matrix is word-organised, namely all the tunnel diodes and resistors of the same word are connected to the same word driver, and all the ordinary diodes at the same bit position are connected to the same digit driver and read-out amplifier.

Information is non-destructively read out, and written in with coincident voltage selection. Both the matrix output signal and digit line driving power are not so dependent upon the size of memory as in other tunnel diode memories previously reported. All the word and digit lines of the actual matrix plane are constructed as asymmetric micro-strip lines printed on epoxy boards.

A memory of this type with a capacity of 272 bits (16 words of 17 bits each) was constructed for a 5 Mc test computer, and has been operating successfully with a cycle time of 200 nsec. A larger matrix of 128 words will be employed in the ETL Mk-6 computer.
Size and Speed of Thin-Magnetic-Film Computer Units

H. J. HARLOFF
Siemens & Halske A.G., Munich, Germany

Thin magnetic films permit the design of very fast computers. The minimum switching time of these elements compares with the signal transit time within a computer unit. Major problems arise in reconciling:
- the geometry of the layers,
- the signal waveforms and delay times,
- the switching time of the magnetic elements,
- the size of a functional unit and
- the timing of the flow of internal operations.

Sandwich-type units comprising conductive, insulating and magnetic layers are proposed. A calculation of the propagation of electro-magnetic waves in thin multilayer elements shows that attenuation and delay time impose strict limits on the diminution of overall unit size and also on the reduction of layer thickness. Coupling and screening effects lead to noise and may lower the switching speed of the magnetic films.

Moreover, the operating speed is affected by unavoidable imperfections of the magnetic films. The highest possible speed can only be achieved if stringent requirements are met in respect of the control field pulses.

The above considerations are applied to some proposed memory configurations and their limits of size and speed are detailed.

Ferrite Core Logic in All-Magnetic Techniques

U. HOLKEN
Max-Planck-Institut für Physik und Astrophysik
Munich, Germany

The circuits described are composed of wire wound ferrite cores only. Other elements, such as diodes or resistors, are omitted. Due to the lack of resistive elements each magnetizing core must find at least one counterpart to be magnetized at the same time.

The fundamental circuit described is a register element with at least three cores combined in a ring-like structure. Input and output are isolated, i.e. a signal written into the input is not transferred to the output at the same time, and the reading of an output signal has no effect at the input terminals. In order to obtain an energy gain of \( \geq 1 \) and a unilateral information flow it is necessary to use a system of interlaced shift pulses with at least four pulses per cycle. By use of such elements, shift registers can be constructed which operate up to about 300 kc/s.

The realization of logical operations can be obtained by combining pairs of output terminals of such register elements. It is also possible to use more complex register elements with several inputs which are able to realize logical operations themselves.

Special circuits are described and experimental results stated.
New Components for Ferroresonant Circuits

M. ALIQUE, J. L. LLORET, I. SANTOS and M. A. ECED
Instituto de Electricidad y Automática, Madrid, Spain

As the non-linear components used at present in ferroresonant circuits show limitations in operation speed, mainly due to losses at high frequencies, a search for new components suitable for ferroresonance was made. Two new components allowing the use of higher frequencies in ferroresonant circuits are described—the variable capacitance diode and the thin magnetic film.

The non-linear behaviour of variable capacitance diodes, when operated in the large signal region, is first considered. A pair of such diodes connected in series-opposition shows non-linear resonance phenomena, when an appropriate linear inductor is used with the diodes. Bi-stable elements have been built to operate at carrier frequencies ranging from 2 to 32 Mc/sec and switching times of the order of 100 nsec have been obtained. Logical circuits performing some of the basic Boolean functions have been made using these bi-stable elements.

Magnetic thin films have attracted attention as a means of obtaining nonlinear inductors at high frequencies. A theoretical study of the thin film behaviour has been carried out, and an equivalent circuit for the film has been obtained.

Measurements on Permalloy thin films agree with the theoretical figures. The non-linear behaviour of thin film inductors is small and leads to difficulties in building bi-stable ferroresonant elements using this component.

Hydraulic and Pneumatic Switching Elements

H. H. GLAETTLI
International Business Machines Corp., Research Laboratory, Zurich, Switzerland

Valve elements and pure fluid amplifiers are compared. Valves are the only closed center device known at present. Their response time cannot be reduced substantially below one millisecond. Their logic power is between that of a transistor and of a relay with a single transfer contact. The volume of a single element, including connecting channels, is of the order of magnitude of 1 cm³. Pure fluid amplifiers behave quite differently. The requirement of a minimum Reynolds-number sets lower limits for supply pressure and power consumption, and both increase as the size decreases. The minimum useful size, therefore, depends not only on manufacturing difficulties and response time. The latter also decreases as the element becomes smaller.

Both types of element can be used with incompressible fluids as well as with gases. Gases, however, introduce capacitive elements into valve circuits. In pure fluid amplifiers, phenomena at higher Mach-numbers (when gases are used) are to a certain extent analogous to cavitation effects in elements working with incompressible fluids.

The circuit design makes profitable use of coplanar arrangements. Similar problems arise as in printed circuits, but in addition, inertia and finite signal propagation velocity must be taken into account.
A Very Small Electronic Digital Computer with Stored Program Control

H. GUMIN and F. K. KROOS
Siemens & Halske A.G., Munich, Germany

A very small electronic digital computer is described, for use with accounting and low-speed data transmission systems, the object being to keep the cost low, if necessary at the expense of operating speed.

Operation is controlled by a program stored on a perforated paper tape loop. During the operating cycle, the computer calls up the data to be processed from a keyboard or from external storage. A small number of internal registers permit the storage of interim results. The running of the program is controlled by 1-bit registers, which are set manually.

The instruction code contains 22 instructions including conditional and unconditional jumps. The word length in the computer is ten decimal digits, which are processed serially. Addition time is 1.3 msec.

Because of the moderate speed requirements in the computer, a low-cost magnetic-core/transistor circuitry has been developed. Building blocks with simple structures have been chosen. The basic block consists of a small magnetic core with several windings, a transistor and a resistor. This block can realize the functions OR, AND, NOR or INHIBIT. It is used, too, to form shift registers and fixed memories. The function of the block is determined by the external wiring only.

The Central Control Unit of the ATLAS Computer

F. H. SUMNER
Manchester University, UK

G. HALEY and E. C. Y. CHEN
Ferranti Ltd., Manchester, UK

The paper describes the central control unit of the ATLAS computer. ATLAS is a high-speed parallel computer with a directly addressable core/drum store arranged to appear to the programmer as a large one-level store, eight magnetic tape channels and several different types of on-line peripheral equipments. Transfer of information between the different stores is autonomous after initiation by the central control.

All the peripherals may operate together, as they are controlled on a time-sharing basis by permanently stored programs which are entered automatically whenever a particular peripheral requires attention.

The complete operation of each individual instruction has been divided into several independent stages, thus permitting one instruction to be initiated before the previous instruction is completed. The logical structure of the central control unit has been simulated by a program on the MERCURY computer, and in this way many engineering designs have been investigated in order to approach an optimum system. It has been possible to arrange that as many as six instructions may be in progress at the same time without affecting the operation of the one-level store, the magnetic tapes or the peripheral equipments.
On a Flexible Implementation of Digital Computer Arithmetic

A. AVIZIENIS
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, Cal., USA

A method is presented for the implementation of arithmetical operations in a digital computer. Addition, subtraction, multiplication and division are performed with operands of variable multiple precision with respect to the length of the adder in an arithmetic unit. The result of the operation is generated sequentially, with the most significant digits appearing first.

A signed-digit number representation is employed to represent the operands and results. It is possible to add (or subtract) signed-digit numbers so that any digit of the sum (or difference) is the function of only two adjacent digits of the input operands. Consequently, carry-propagation chains of variable length do not occur during addition or subtraction.

When a special digit is employed to indicate the first non-significant position of each operand, an arithmetical operation can be terminated when all required significant digits of the result have been generated. It is expected that this implementation of arithmetic will be applicable in asynchronous computers, and in complex parallel systems, in which several arithmetic units are required to perform calculations asynchronously at their own maximum speeds.

A Comparative Study of Propagation Speed-up Circuits in Binary Arithmetic Units

M. LEHMAN
Ministry of Defence, Tel Aviv, Israel

Despite the fact that mismatch has always existed between the speeds of digital computer input/output and storage devices on the one hand, and of arithmetic units on the other, a considerable amount of thought and effort has been expended in order to achieve the highest possible arithmetic speeds. More particularly, various techniques and circuits have been proposed, and in some cases tested, which reduce the time spent in the process of carry propagation.

The minimisation of this time was considered by Burkes, Goldstine and von Neumann in their classic 1947 Princeton report, and since that time a seemingly endless stream of proposals has appeared. The present paper attempts to summarise and compare the various methods which have previously been described, on a combined basis of cost and speed.
An Algorithm for Division

A. SVOBODA

Research Institute for Mathematical Machines, Prague, CSSR

The algorithm for division presented in this paper can be used in automatic computers employing decimal representation of numbers, and possessing facilities for the computation of the expression \((ma + b)10^n\), where \(m\) is a decimal digit, \(n\) is an integer, and \(a, b\) are numbers represented in a decimal code. The generation of the integral multiples \(ma\) can be achieved by using a special multiplier or a table of multiples. The algorithm is based on the fact that when the divisor is of the form \(y = 1 + h\), where \(h\) is suitably small, the most significant digit of the quotient is given approximately by the most significant digit of the dividend.

The algorithm consists of two parts. The first part normalises both the dividend and the divisor to bring the latter near unity; this process employs a maximum of 4 steps. The second part then performs the division in as many steps as there are digits in the quotient. This portion of the algorithm is first presented formally for proof, and then modified for mechanization in an arithmetic unit. Speed of operation and simplicity of implementation are two of the advantages claimed for this means of division.

Design of a Data Processing System with Built-in Time-Sharing

J. OBLONSKY and A. SVOBODA

Research Institute for Mathematical Machines, Prague, CSSR

The logical design of the Czechoslovak EDP-system EPOS embodies both a modular structure and time-sharing facilities. The system consists of a central computer and a set of peripheral equipments, interconnected by a set of buses so that practically any combination of input/output units and tape units may be included.

The internal time-sharing of the central computer permits the simultaneous performance of some operations occurring in the same program. The external time-sharing permits parallel processing of up to five independent programs in one EPOS-system. The central computer contains a special organizing unit, which automatically performs the switching between these programs, in accordance with the instantaneous priority of each program. A set of priorities is laid down which aims to ensure that the maximum utilization of the central computer, and other components of the system, is achieved.
An Experimental System for Logic Design Data Accumulation and Retrieval

R. J. PREISS
International Business Machines Corp., Poughkeepsie, NY, USA

To raise the productivity of the design engineer, and to eliminate the drafting stage in the production of documents for the design, manufacture and servicing of computers, an experimental logic design data accumulation and retrieval system was programmed.

The basic requirements of the language, to facilitate communication between the engineer and the computer, are described and also those of the programming system for a minimum of manual specifications and intervention. For retrieval, especially, a graduated reference to the data was needed. A solution was found in a listing-form modification of Polish (Lukasiewicz) notation. The unit of information is a line of print. The next larger unit is a sentence, followed by a paragraph, a page and a design.

The system has been programmed so that two passes are required to put data on the master tape. In the first pass, the data and the area to which it is to be added are inspected to determine whether the data are complete and whether they connect properly to the rest of the design, error messages being generated in the process. The actual updating of the master tape occurs in the second pass.

The paper concludes with experiences encountered in the operation of the experimental system.

The KT Pilot Computer — A Microprogrammed Computer with a Phototransistor Fixed Memory

H. HAGIWARA
Kyoto University, Kyoto, Japan

K. AMO, S. MATSUSHITA and H. YAMAUCHI
Tokyo Shibaura Electric Co. Ltd., Kawasaki, Japan

This paper describes the pilot model of the computer designed in co-operation between Kyoto University and Tokyo Shibaura Electric Co. Ltd.

The logical design incorporates micro-programming techniques into an asynchronous system. The circuitry of this computer has certain unique features: — a newly developed phototransistor fixed memory, and logical blocks using very fast mesa transistors and diodes as circuit components.

The computer is a binary, parallel, single indirectly addressed machine, and incorporates 32 macro-orders. A magnetic core memory, a high speed paper tape reader and a Flexewriter are used respectively as the main memory and as input and output devices.

The control system is embodied in a phototransistor fixed memory which is laid out in rectangular form. A punched card is laid on the phototransistor unit, and light is projected onto the unit through the holes of the card. This pattern of card holes determines the sequence and content of the micro-orders and thus the computer operator can easily alter the function code of the computer by merely replacing the punched card.

The registers consist of current switching type flip-flops which operate up to 50 Mc/sec. Mesa and gold-bonded diodes are used for the AND-OR circuits.
System Design of the ETL Mk-6 Computer

S. TAKAHASHI, H. NISHINO, K. YOSHIHIO and K. FUCHI

Electrotechnical Laboratory, Tokyo, Japan

This paper presents an outline of the system design of the ETL Mk-6, a high-speed computer under development at the Electrotechnical Laboratory. Employing 8 Mc/sec two-phase transistorised basic circuitry and a variant on Kilburn's adder, it adds two 48-bit fixed-point numbers in 250 nsec and multiplies in 4 μsec on the average, excluding access time.

To minimize the effect of access time, the Mk-6 employs a tunnel diode high-speed memory, and a capacitance type fixed memory, both with 250 nsec cycle time, as well as ferrite core stores and magnetic drums. However, it appear to the programmer as a one-level store of $2^{18}$ words.

A considerable amount of effort has also been made to ease compiling of programs written in ALGOL, and a compromise has been reached between this aim, and that of a high operational speed. An arithmetic stack which follows the last-in-first-out concept, and index registers which enable multiple modifications are introduced. The tunnel diode memory which is one of the characteristic features of this computer is allocated to these.

In order to realize its high-speed capabilities, the Mk-6 has flexible parallel programming features with provision to save the contents of the arithmetic stack and index and other registers, in the case of a program interruption.

Design of an Arithmetic Unit Incorporating a Nesting Store

R. H. ALLMARK and J. R. LUCKING

The English Electric Co. Ltd., Kidsgrove, UK

This paper describes the arithmetic unit of a computer whose order code is based on the Reverse Polish algebraic notation. The order code has been realised by causing the arithmetic unit to operate on data stored in the most accessible registers of a nesting store; these registers are of the transistor flip-flop type but are backed up by sixteen fast magnetic core registers. The functions are performed as micro-programmes of transfers between the registers in the arithmetic unit, and the necessary arrangement of transfer paths, logical gates and arithmetic circuits is described. The number system is binary, using the two's-complement representation of negative numbers. Automatic floating-point operations are included which use an autonomous unit to perform the shifts required.
Modern Programming Methods and Problems and their Influence on the Design of Computing Instruments

I. O. KERNER
Jena, Germany

The instruction systems of the first program-controlled automatic computers were so arranged as to permit, in principle, the solution of mainly arithmetical problems. Programming, however, soon influenced the design and instruction-structure of newly contemplated computers. At present the development of programming is in the direction of the so-called formulation languages. By building up on any primitive instruction language, it is possible so to design others that certain word-groups are defined as a new word of the master language, thus creating new semantics. Different languages inevitably bring about the problem of translation. Compilers of two types are already available: firstly as programmes, i.e. sequences of instructions for the automat, and secondly, as constructional units, i.e., as a number of operations of which the sequential control is given by the words to be translated. Programming and fixed circuits are equivalent but demands must be kept within reasonable limits. One can limit translation to a few new elementary processes. A machine with plug-in logical circuits and pertinent programming seems to be the preliminary solution to the problem.

Application of a Finite Set Covering Theorem to the Simplification of Boolean Function Expressions

M. B. WELLS
University of California, Los Alamos, N. Mex., USA

The problem of simplifying normal form expressions for Boolean functions is essentially a finite set covering problem. Given a finite set of points \( V \), \( P \) a set of subsets of \( V \), and a cost function with domain \( P \), one tries to find a cover \( C \), a subset of \( P \) with \( U C = V \), of minimum cost. A theorem is presented which yields a criterion for determining when a minimum cover has been obtained. Although the criterion itself is more theoretical than practical, it does suggest a useful simplification procedure. This algorithm systematically improves a cover by increasing its order of irredundancy. While the theorem insures that the process will eventually yield a minimum cover, the value of the algorithm springs from the empirical observation that low order irredundant covers very often have costs closely approximating to minimum costs. Thus, the method is helpful in obtaining approximate solutions for functions of many variables.
Digital Filters of Threshold Elements

G. HOTZ
Telefunken GMBH, Konstantz, Germany

Pyramid-like combinations of binary threshold elements to form more complex switching circuits with several inputs and one output, are considered. The inputs are connected to the outputs of the delay elements of a shift register, so that a sequential network is obtained with one binary input and one binary output; these electrical networks are our digital filters.

We investigate the mappings into itself of the set of binary sequences, infinite in both directions, defined by these filters. The filter — mapping relation is not one-to-one, and this leads to a classification of the filters, i.e. filters, which result in the same mapping, come within the same class. For these classes, normal forms are found which are not, however, unique.

Threshold Logic with one or more than one Threshold

P. ERCOLI and L. MERCURIO
Istituto Nazionale per le Applicazioni de Calcolo, Rome, Italy

This paper generalizes the concept of threshold, or majority logic and introduces an operator which represents the logical function of threshold elements with one or with more than one threshold (these latter elements being suitably defined). Some of the properties of this operator are demonstrated, and examples are given of their use for the synthesis of binary functions with threshold elements, the procedures and the formulae being essentially the same for all types of threshold element, due to the generalized approach to the problem.

It is also shown how to find, under certain restrictions, equivalent forms of the same function for the sake of circuit optimization.

Finally the implementation of threshold elements with two, three or more thresholds is outlined.
Some Theorems Useful in Threshold Logic for Enumerating Boolean Functions

E. GOTO
Massachusetts Institute of Technology Cambridge, Mass., USA

H. TAKAHASI
University of Tokyo, Japan

A new method of classifying Boolean functions, called self-dual classification, and specially suited for threshold logic, is presented. Besides permutation and negation of variables, two other operations, self-dualization and anti-self-dualization, are introduced to define the equivalence of functions within a class. These operations preserve the number of non-linear threshold elements in combinatorial switching circuits. The self-dual classification considerably reduces the number of different types of switching function in threshold logic, e.g., for up to 4 variables 83 instead of the 402 in the conventional classification.

Lower and upper bounds of the number \( N(n) \) of linear input functions, i.e., the function realizable with a single threshold element, of up to \( n \) variables are given. Upper bounds show that \( N(n) \) is smaller than \( 2^{n^2} \) and a lower bound shows that \( N(n) \) is larger than \( 2^{0.23n^2} \). From these bounds it is conjectured that for large \( n \), \( N(n) \) would behave asymptotically like \( 2^{kn^2} \) where \( k \) is a certain constant between \( \frac{1}{4} \) and 1.
<table>
<thead>
<tr>
<th>Name</th>
<th>Country</th>
<th>Page, Column</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gazzano, A</td>
<td>Italy</td>
<td>VII, 2</td>
</tr>
<tr>
<td>Genuys, F.</td>
<td>France</td>
<td>VI, 4</td>
</tr>
<tr>
<td>Glaettli, H. H.</td>
<td>Switzerland</td>
<td>XVII, 4</td>
</tr>
<tr>
<td>Goldberg, R.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>Gotlieb, C. C.</td>
<td>Canada</td>
<td>II, 3</td>
</tr>
<tr>
<td>Goto, E.</td>
<td>U.S.A.</td>
<td>XX, 4</td>
</tr>
<tr>
<td>Gray, H. J.</td>
<td>U.S.A.</td>
<td>VIII, 1</td>
</tr>
<tr>
<td>Green, C.</td>
<td>Italy</td>
<td>VII, 2</td>
</tr>
<tr>
<td>Guerré, L.</td>
<td>Italy</td>
<td>IV, 3</td>
</tr>
<tr>
<td>Gumin, H.</td>
<td>Germany</td>
<td>XVIII, 1</td>
</tr>
<tr>
<td>Habr, J.</td>
<td>Czechoslovakia</td>
<td>VI, 1</td>
</tr>
<tr>
<td>Hagiwara, H.</td>
<td>Japan</td>
<td>XIX, 3</td>
</tr>
<tr>
<td>Haley, G.</td>
<td>England</td>
<td>XVIII, 2</td>
</tr>
<tr>
<td>Harloff, H. J.</td>
<td>Germany</td>
<td>XVII, 1</td>
</tr>
<tr>
<td>Heit, M.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>Hindle, R.</td>
<td>England</td>
<td>I, 1</td>
</tr>
<tr>
<td>Hölsen, U.</td>
<td>Germany</td>
<td>XVII, 2</td>
</tr>
<tr>
<td>Holt, A. W.</td>
<td>U.S.A.</td>
<td>XV, 1</td>
</tr>
<tr>
<td>d'Hoop, H.</td>
<td>Italy</td>
<td>VII, 2</td>
</tr>
<tr>
<td>Hotz, G.</td>
<td>Germany</td>
<td>XX, 2</td>
</tr>
<tr>
<td>Howland, J. L.</td>
<td>Canada</td>
<td>III, 6</td>
</tr>
<tr>
<td>Huskey, H. D.</td>
<td>U.S.A.</td>
<td>XIV, 2</td>
</tr>
<tr>
<td>Ishii, O.</td>
<td>Japan</td>
<td>XVI, 4</td>
</tr>
<tr>
<td>Jeanniot, J. P.</td>
<td>Canada</td>
<td>II, 2</td>
</tr>
<tr>
<td>Julesz, B.</td>
<td>U.S.A.</td>
<td>XIII, 2</td>
</tr>
<tr>
<td>Kay, M.</td>
<td>U.S.A.</td>
<td>IX, 3</td>
</tr>
<tr>
<td>Keese Jr, W. M.</td>
<td>U.S.A.</td>
<td>XIV, 2</td>
</tr>
<tr>
<td>Kerner, I. O.</td>
<td>Germany</td>
<td>XIX, 6</td>
</tr>
<tr>
<td>King, W.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>Korganoff, A.</td>
<td>France</td>
<td>III, 3</td>
</tr>
<tr>
<td>Kotkin, B.</td>
<td>U.S.A.</td>
<td>IV, 4</td>
</tr>
<tr>
<td>Kroos, F. K.</td>
<td>Germany</td>
<td>XVIII, 1</td>
</tr>
<tr>
<td>Kudielka, V.</td>
<td>Austria</td>
<td>XII, 2</td>
</tr>
<tr>
<td>Kuno, S.</td>
<td>U.S.A.</td>
<td>IX, 1</td>
</tr>
<tr>
<td>Kuntzman, J.</td>
<td>France</td>
<td>V, 1</td>
</tr>
<tr>
<td>Landow, H.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>Läuchli, P.</td>
<td>Switzerland</td>
<td>IV, 1</td>
</tr>
<tr>
<td>Lehman, M.</td>
<td>Israel</td>
<td>XVIII, 1</td>
</tr>
<tr>
<td>Leigh, A.</td>
<td>Canada</td>
<td>I, 2</td>
</tr>
<tr>
<td>Lenk, F.</td>
<td>Austria</td>
<td>X, 2</td>
</tr>
<tr>
<td>Lewis, T. H.</td>
<td>U.S.A.</td>
<td>VII, 3</td>
</tr>
<tr>
<td>Lloret, J. L.</td>
<td>Spain</td>
<td>XVII, 3</td>
</tr>
<tr>
<td>Lombardi, L.</td>
<td>Italy</td>
<td>XIV, 3</td>
</tr>
<tr>
<td>Lucas, P.</td>
<td>Austria</td>
<td>XV, 4</td>
</tr>
<tr>
<td>Lucking, J. R.</td>
<td>England</td>
<td>XIX, 5</td>
</tr>
<tr>
<td>Matsushita, S.</td>
<td>Japan</td>
<td>XIX, 3</td>
</tr>
<tr>
<td>McGee, W. C.</td>
<td>U.S.A.</td>
<td>XV, 2</td>
</tr>
<tr>
<td>McQuillan, J. D. R.</td>
<td>England</td>
<td>XX, 3</td>
</tr>
<tr>
<td>Mercurio, L.</td>
<td>U.S.A.</td>
<td>X, 4</td>
</tr>
<tr>
<td>Miller, A. E.</td>
<td>Japan</td>
<td>XVI, 4</td>
</tr>
<tr>
<td>Murata, K.</td>
<td>France</td>
<td>XIII, 5</td>
</tr>
<tr>
<td>Nadler, M.</td>
<td>Japan</td>
<td>XVI, 4</td>
</tr>
<tr>
<td>Nakazawa, K.</td>
<td>England</td>
<td>VIII, 3</td>
</tr>
<tr>
<td>Needham, R. M.</td>
<td>Japan</td>
<td>XIX, 4</td>
</tr>
<tr>
<td>Nishino, H.</td>
<td>Czechoslovakia</td>
<td>XIX, 1</td>
</tr>
<tr>
<td>Obloinsky, J.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>O'Connor, E.</td>
<td>U.S.A.</td>
<td>IX, 1</td>
</tr>
<tr>
<td>Oettinger, A. G.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>Ogle, J.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>Opier, A.</td>
<td>Germany</td>
<td>XIV, 1</td>
</tr>
<tr>
<td>Paul, M.</td>
<td>U.S.A.</td>
<td>XV, 2</td>
</tr>
<tr>
<td>Perkins, R.</td>
<td>England</td>
<td>XVI, 3</td>
</tr>
<tr>
<td>Perry, G. H.</td>
<td>Germany</td>
<td>XI, 1</td>
</tr>
<tr>
<td>Petri, C. A.</td>
<td>Poland</td>
<td>VI, 2</td>
</tr>
<tr>
<td>Pietrzykowski, T.</td>
<td>U.S.A.</td>
<td>XIX, 2</td>
</tr>
<tr>
<td>Preiss, R. J.</td>
<td>U.S.A.</td>
<td>VIII, 1</td>
</tr>
<tr>
<td>Prywes, N. S.</td>
<td>Canada</td>
<td>XII, 3</td>
</tr>
<tr>
<td>Ratz, H. C.</td>
<td>Germany</td>
<td>XII, 4</td>
</tr>
<tr>
<td>Remus, H.</td>
<td>Italy</td>
<td>VII, 2</td>
</tr>
<tr>
<td>Riotte, A.</td>
<td>Austria</td>
<td>X, 2</td>
</tr>
<tr>
<td>Rothauser, E.</td>
<td>Switzerland</td>
<td>III, 1</td>
</tr>
<tr>
<td>Name</td>
<td>Country</td>
<td>Page</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>Sakai, T.</td>
<td>Japan</td>
<td>XIII, 3</td>
</tr>
<tr>
<td>Sandiford, P. J.</td>
<td>Canada</td>
<td>II, 2</td>
</tr>
<tr>
<td>Santos, I.</td>
<td>Spain</td>
<td>XVII, 3</td>
</tr>
<tr>
<td>Saunders, M. G.</td>
<td>Canada</td>
<td>XIII, 1</td>
</tr>
<tr>
<td>Scharbert, J.</td>
<td>Germany</td>
<td>XVI, 1</td>
</tr>
<tr>
<td>Sellin, K. G.</td>
<td>Denmark</td>
<td>IX, 4</td>
</tr>
<tr>
<td>Shafritz, A. B.</td>
<td>U.S.A.</td>
<td>X, 4</td>
</tr>
<tr>
<td>Shallow, E. W.</td>
<td>England</td>
<td>XVI, 3</td>
</tr>
<tr>
<td>Simon, H. A.</td>
<td>U.S.A.</td>
<td>XI, 4</td>
</tr>
<tr>
<td>Slawson, A. W.</td>
<td>U.S.A.</td>
<td>XIII, 4</td>
</tr>
<tr>
<td>Slesinger, D.</td>
<td>U.S.A.</td>
<td>XV, 3</td>
</tr>
<tr>
<td>Smith, J. R.</td>
<td>U.S.A.</td>
<td>X, 4</td>
</tr>
<tr>
<td>Stauffer, R. B.</td>
<td>U.S.A.</td>
<td>VII, 3</td>
</tr>
<tr>
<td>Steinbuch, K.</td>
<td>Germany</td>
<td>X, 3</td>
</tr>
<tr>
<td>Sumner, F. H.</td>
<td>England</td>
<td>XVIII, 2</td>
</tr>
<tr>
<td>Svoboda, A.</td>
<td>Czecho-Slovakia</td>
<td>XVIII, 5</td>
</tr>
<tr>
<td>Svoboda, A.</td>
<td>Czecho-Slovakia</td>
<td>XIX, 1</td>
</tr>
<tr>
<td>Swanson, D. R.</td>
<td>U.S.A.</td>
<td>VIII, 4</td>
</tr>
<tr>
<td>Takahashi, S.</td>
<td>Japan</td>
<td>XIX, 4</td>
</tr>
<tr>
<td>Takahashi, S.</td>
<td>Japan</td>
<td>XVI, 4</td>
</tr>
<tr>
<td>Takahasi, H.</td>
<td>Japan</td>
<td>XX, 4</td>
</tr>
<tr>
<td>Thomas, G. H. M.</td>
<td>Canada</td>
<td>XII, 3</td>
</tr>
<tr>
<td>Uhr, L.</td>
<td>U.S.A.</td>
<td>XII, 1</td>
</tr>
<tr>
<td>Urich, W.</td>
<td>Germany</td>
<td>V, 3</td>
</tr>
<tr>
<td>Vazsonyi, A.</td>
<td>U.S.A.</td>
<td>II, 4</td>
</tr>
<tr>
<td>Vossler, C.</td>
<td>U.S.A.</td>
<td>XII, 1</td>
</tr>
<tr>
<td>Wachspress, E. L.</td>
<td>U.S.A.</td>
<td>III, 5</td>
</tr>
<tr>
<td>van Wauwe, A.</td>
<td>Italy</td>
<td>VII, 2</td>
</tr>
<tr>
<td>Wells, M. B.</td>
<td>U.S.A.</td>
<td>XX, 1</td>
</tr>
<tr>
<td>Yamauchi, H.</td>
<td>Japan</td>
<td>XIX, 3</td>
</tr>
<tr>
<td>Yoshihiro, K.</td>
<td>Japan</td>
<td>XIX, 4</td>
</tr>
<tr>
<td>Zendeh, F.</td>
<td>Germany</td>
<td>X, 3</td>
</tr>
</tbody>
</table>