Methods of processing information from track detectors of high-energy charged particles

N. M. Nikityuk
Joint Institute for Nuclear Research, Dubna

Fiz. Élem. Chastits At. Yadra **26**, 719–779 (May–June 1995)

The review is devoted to methods of processing track information. The purpose and general properties of track detectors of charged particles of high and ultrahigh energies are briefly described. The methods and means used to reconstruct tracks with high curvature and multiplicity $\geqslant 100$ are considered: on-line methods, hardware track processors, and methods that combine these two approaches. The review also describes new methods proposed for processing track information using effective algorithms and hardware processors used in information theory for pattern recognition and data compression: neural networks, coordinate transformations of various kinds, associative methods of data processing, transputers, etc. Specific examples are given of the construction of track processors and multilevel multiprocessor systolic systems used to process track information. Questions of the architecture and requirements on the trigger systems proposed for use in future experiments on the Large Hadron Collider are discussed.

INTRODUCTION

The construction and improvement of colliding-beam accelerators fosters the rapid development of high-energy physics. In the colliders under development, the luminosity reaches 10^{33}–10^{34} $cm^{-2}\cdot s^{-1}$ at particle energies in the teva-electron-volt region. In the Large Hadron Collider under construction with proton energy 10 TeV, it is expected that the event frequency will be 100 MHz. Then the number of useful events may be 10–100 per second at a multiplicity of 100 or more. In addition, there is also rapid development of a new approach in the methods of high-energy physics associated with the study of short-lived particles (lifetime 10^{-13} s or less). The topology of events with such particles may contain one primary or several secondary decay vertices from which tens of particle tracks emanate, and it is necessary to determine the coordinates of the decay vertices rapidly and with micron accuracy.

As a rule, a typical high-energy physics facility consists of large spectrometers that contain tens of thousands of detection channels. The number of data obtained per event may be a few or tens of megabytes. Under competitive conditions, the problem is to process a vast number of data stored on magnetic tape with maximal reliability in the minimum possible time. Under these conditions, one not only needs effective on-line program methods for filtering useful events but also faces the acute and complicated problem of filtering tracks and event topologies on the background of appreciable disturbance during the shortest possible time (a few or tens of microseconds) by hardware methods. It has been noted in a number of studies that this part of the trigger system is currently the most vulnerable, and further success in the development of electronic methods of high-energy physics will to a large degree depend on the progress in the development of fast track processors capable of reconstructing during a short time and mainly by means of hardware processors the tracks of complicated events, so that the processing of the experimental data can be greatly accelerated.

The aim of this review is to consider, in a brief and accessible form, the present status and prospects for development of methods of processing coordinate information and reconstructing the tracks and topologies of complicated events that are studied by means of the electronic methods of high-energy physics. We also describe methods and algorithms used to construct fast track processors and the block diagrams of multiprocessor systems used for the final analysis of track information. We discuss the construction of multilevel trigger systems for future experiments.

In this Introduction, we establish the topicality of the subject and describe the content of the review section by section. Section 1 is devoted to a brief exposition of the properties of the track detectors used in electronic methods of high-energy physics. Section 2 of the review is devoted to methods of processing track information in the off-line regime. In Sec. 3 we describe a method of reconstructing events that combines both on-line and hardware methods. In particular, we describe in detail a method of constructing a track processor based on coordinate transformation in accordance with the Hough algorithm. Section 4 is devoted to the hardware realization of the "tree" algorithm. This method has great prospects for application in conjunction with associative (content-addressable) memories. Section 5 demonstrates the prospects for the use of neural networks and neuronlike processors for the processing of track information; they ensure a high degree of parallelism of the calculations and a good quality of recognition of complicated events with high multiplicity. Section 6 is devoted to a brief exposition of the elastic algorithm proposed for the processing of data with high track multiplicity in the presence of appreciable disturbance. The structure of an efficient processor in which a "contiguity mask" method is realized is considered in the Sec. 7. Many experiments often use specialized track processors designed for the efficient solution of the problems of a particular experiment. Such processors will also be used in the future. A characteristic specialized processor is described in Sec. 8 of the review. A method of parallel data compres-

 1063-7796/95/030302-27$10.00

TABLE I. Comparative characteristics of track detectors.

Detector	Spatial resolution, μm	Time resolution, ns	Load capacity, Hz	Two-track resolution, μm
Multiwire proportional chambers	200	100	10^6	600
Drift chambers	30–150	100–1000	10^7	120–600
Scintillators	5–10	1–10	10^7	20–50
Semiconductor detectors	3–10	1–5	10^6	40
Streamer chambers	200	10^3	10^5	40
Bubble chambers	100–200	10^6	10^4	2000

sion based on syndrome coding is discussed in Sec. 9. Section 10 gives a brief exposition of the structures of the employed multilevel multiprocessor systems that are used to process track information at high trigger hierarchy levels, at which the final processing of track data is done. In Sec. 11 of the review, we discuss aspects of the architecture of the multilevel trigger systems that it is proposed to use in future experiments at the Large Hadron Collider.

1. BRIEF CHARACTERIZATION OF TRACK DETECTORS

A large number of types of track detector used in electronic methods of high-energy physics are known. In part, this can be explained by the large number of problems that physicists solve, the diversity of the parameters of the elementary particles that must be measured by electronic methods, the complexity of the topologies of the studied events, the high accuracy that is required, and the rapid development of the methodology of detectors, which is based, in particular, on the achievements of semiconductor technology, optoelectronics, and the technology of gas-filled detectors. The typical characteristics of track detectors used in experiments are given in Table I. The wide spread of some of the parameters referring to detectors of the same kind can be explained by the fact that the detectors used in experiments are, as a rule, unique devices, and their characteristics often depend on the level of the technology of the manufacturers, the employed material, etc. It should be noted that the best parameters currently achieved are given. This applies above all to the detectors based on scintillating optic fiber waveguides. A parameter such as the resolution time determines the rate of event detection. However, the accumulation of statistics can be limited in a number of cases by the methods of information readout. This applies above all to detectors such as scintillating optic fiber waveguides and charge coupled devices, which belong to the class of semiconductor detectors. We consider briefly the properties of track detectors used in electronic methods of high-energy physics. In both current and planned experiments, the most popular are gas-filled detectors; in particular, multiwire proportional chambers have become very well known.[1] However, with increasing multiplicity of the detected events and increasing complexity of the topologies, the use of multiwire proportional chambers in large spectrometers became problematic during the eighties, above all on account on the difficulties that arise in the process of event reconstruction and the greater complexity of the spectrometers as a whole. Therefore, drift chambers were found to be more promising for the future. The parameters of the classical drift chambers that served as the prototypes for the development of multilayer drift chambers of cylindrical shape were considered in Ref. 2. The development of these detectors was based on the best achievements in the methodology of multiwire proportional chambers (high accuracy, the possibility of measuring a second coordinate by means of cathode strips). The development of drift chambers opened up the possibility of accurate measurement of one of the coordinates through measurement of the drift time of electrons from the interaction point to a sensitive wire. The achievements in the methodology of classical drift chambers with an account of the employed electronics are reflected in Ref. 3.

From the middle of the seventies, the methodology of drift chambers, above all chambers of cylindrical shape, progressed rapidly. A typical cylindrical drift chamber[4] has 15–20 layers (points per track). An individual drift cell has a simple structure—a signal wire and a triplet of potential electrodes—and the typical drift gap is 1–2 cm. The further development of multilayer drift chambers of cylindrical shape took the form of the development of three-coordinate gas detectors.[5]

Three types of such detectors are used in experiments: time-projection chambers,[6] jet chambers, and vector drift chambers. A time-projection chamber is actually a large ionization chamber at the ends of which there are multiwire proportional chambers of circular shape with cathode readout. Using multichannel detecting electronics, multistop time–digital converters, and the geometry of the detector, one can measure in space hundreds of coordinates r, z, and φ along one track (in a cylindrical coordinate system) at a multiplicity of 200 or more. A modern time-projection chamber is described in Ref. 7. Three-coordinate detectors of "jet" type contain in the volume of the chamber sensitive wires in the radial direction and, in a number of constructions, drift cells as well. This makes it possible to improve the two-track spatial resolution, especially in a region in which the track density is high (region of a jet).[8] Compared with jet chambers, vector chambers[9] are distinguished by the fact that their volume contains a large number of high-precision drift cells, which form concentric layers. The chamber itself consists of several such layers but with different sizes of the drift cells (the cell density is greatest close to the center). All these and

TABLE II. Parameters of some tested track detectors.

Type	Function	Commentary
Si pixel 30 μm × 300 μm $>10^7$ elements	Coordinates of secondary decay vertices	Much electronic readout is needed
Microstrip Si detectors Width: 50 μm Length: Several centimeters 10^6 strips	Measurement of momentum	Much electronics and accurate adjustment to within a few microns required
Microstrip gas avalanche chambers. 50 μm × 5 cm	Measurement of momentum	
Scintillating optic fiber waveguides 0.8 mm × 3 cm 10^6 elements	Measurement of momentum Trigger systems	Small-size, low-noise, and fast photodetectors needed
Mini drift (straw) tubes 4 mm × 3000 mm 10^6 elements	Measurement of momentum Trigger systems	Electronics and precision mechanics required

other refinements have the aim of simplifying the search for tracks and raising the detection accuracy as much as possible.

A bottleneck of the three-dimensional gas detectors considered above is the relatively slow response (tens of microseconds), since the typical electron drift time is 50 μm/ns. The desire to raise the speed of operation of such detectors in experiments in which the frequency of useful events may be hundreds of kilohertz or more led to the development of three-dimensional gas detectors based on multidrift tubes and tubes of straw type,[10] in which the maximum drift time does not exceed 100 ns, and the accuracy of coordinate detection is of order 20 μm. An individual tube consists of one anode wire and a plastic cylinder covered with aluminum. There is a detailed description of such detectors in the review of Ref. 11, and examples of their use in experiments are described.

In connection with the wide development of vertex detectors, by means of which the properties of short-lived particles are studied, there are great prospects for the development of high-precision track detectors such as multiplane silicon detectors,[12,13] charge coupled arrays,[14] semiconductor drift chambers,[15] and two-dimensional (pixel) silicon detectors.[16] The last two types of detector are in the stage of experimental development. The fast response inherent in scintillation counters and hodoscopes is guaranteed by track detectors based on scintillating optic fiber waveguides, which give micron accuracy.[17] It should be mentioned that the use of optical methods of detection and processing of signals in the experiments of high-energy physics is associated with a number of new properties inherent in light signals—the absence of a charge of the photon and compactness and considerable simplification of the systems for readout of the track information.[18] An optical track processor has been developed to process data detected in multiwire proportional chambers.[19]

Modern achievements in the field of track detectors that it is proposed to use in future experiments on the Large Hadron Collider are reviewed in Ref. 20. Some of these detectors, which it is proposed to use in the central detector, are given together with their parameters in Table II.

The main reasons why it is necessary to measure the tracks of charged particles by means of a central detector placed close to the beam collision region are as follows: determination of the electron charge; more accurate measurement of the muon momenta; the possibility of determining hadrons with large momentum p_t at the trigger level; it is possible to recognize b and τ decays by detecting secondary decay vertices.

Figure 1 shows schematically the detector system that it is proposed to use in the future experiments on the collider. It contains a vertex and central detector, a calorimeter with electromagnetic and hadronic sections, and an iron muon screen, behind which there follow muon track chambers, which cover a large part of the facility. In addition, this figure illustrates the reaction of the detectors to the four most characteristic particles and jets [B. Denby]. The beam is perpendicular to the plane of the page. It can be seen that the muons pass freely even through both sections of the calorimeters, releasing a small fraction of energy in each section, and also through the iron screen, and are ultimately detected by means of the muon chambers. The electrons completely give up all their energy in a local region of the electromagnetic calorimeter. In their turn, the pions give up their energy by interacting with both sections of the calorimeter. Jets, consisting of many different particles (mainly pions), give up their energy in the extensive regions of both sections of the calorimeter. As one would expect, the neutrinos pass freely through the complete detector system.

2. STAGES IN THE DEVELOPMENT OF HARDWARE METHODS FOR PROCESSING TRACK INFORMATION

The following stages may be noted in the development of the methods for searching for and reconstructing track information by means of specialized processors.

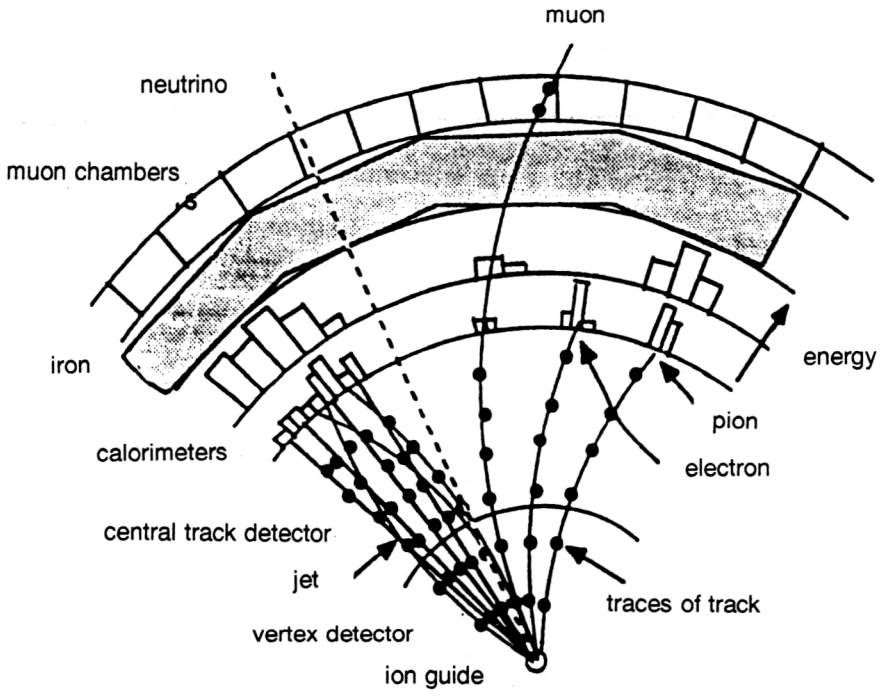

FIG. 1. Behavior of various particles in a detector system.

Specialized processors are used to study (1–2)-particle events. It is here necessary to mention the pioneering developments made at the beginning of the seventies. Both analog[21] and digital[22] processors were used. Scintillation hodoscopes and multiwire proportional chambers were used as sources of information. This stage in the development of specialized processors is considered in detail in the review of Ref. 23. The appearance of fast memory modules and programmable logic arrays in the middle of the seventies created the prerequisites for the development of hardware-programmed specialized processors with extensive functional possibilities. However, these processors were developed to solve narrowly specialized problems, above all for experiments with low detectable multiplicity, for scattering-angle selection of particles,[24] or for calculation of particle momenta. For finding the tracks of useful signals with high multiplicity, complicated topology, and appreciable track curvature, wide use was made of powerful microcomputers and a variety of emulators of large and small computers.[25] The use of such technology made it possible to perform in real time a preliminary selection of track information with subsequent analysis of the data in large computers. The processing time for a three-prong event was a few seconds. However, in connection with the rapid development of the physics of high and ultrahigh energies the problem is now that of guaranteeing selection of complicated events with a multiplicity of more than 100 during a few microseconds. To solve such a difficult problem, there is currently intensive development of new methods and approaches based on algorithms employed in information theory for pattern recognition and above all neural networks, cellular automata, associative processing methods, various forms of transformation of variables, etc.

3. ON-LINE METHODS OF PROCESSING TRACK INFORMATION

Two approaches in the processing of physical information have been developed traditionally in the electronic methods of high-energy physics: on-line methods and hardware methods. In some experiments, these approaches can be used simultaneously, and this makes it possible to raise appreciably the efficiency of the operation of specialized processors. Naturally, the on-line methods are, in their turn, based on mathematical algorithms and the methods of pattern recognition.

Equation of motion of a charged particle in a magnetic field. A track model

The basis for determining the trajectory of a charged particle in a static magnetic field is the Lorentz equation. Ignoring the energy loss and the effect of an electric field, and assuming that the absolute magnitude of the momentum $\mathbf{p}$ remains unchanged in vacuum, the equation of motion of a charged particle can be represented with allowance for multiple scattering in the form[26]

$$d^2\mathbf{r}/ds^2=(q/|\mathbf{p}|)(d\mathbf{r}/ds\mathbf{B}(\mathbf{r}))+\mathbf{R}(s),$$

where $\mathbf{r}$ gives the Cartesian coordinates (x, y, z) of a vector; s is the path length, and $ds^2=dx^2+dy^2+dz^2$; q is a constant proportional to the charge; $\mathbf{B}$ is the magnetic induction; and $\mathbf{R}(s)$ is white noise (a stochastic process) that describes the multiple scattering.

Instead of s, the equations of motion often contain a definitely known (independent) coordinate z, which is directed at right angles to the planes of the detectors or coincides with the cylinder axis if the detector has a cylindrical shape. Thus, to determine the trajectory of a charged particle

in a magnetic field, it is necessary to know the values of three parameters: two coordinates (x,y) for a fixed value of z near the target, direction cosines $(dx/ds,\ dy/ds)$, and some function of the momentum (for example, $1/p$). These five coordinates form a five-dimensional space, which is also called the phase space. Then the trajectory of the charged particle can be represented as a point in phase space. To obtain these five quantities by means of a computer, complicated calculations are performed on the track parameters by means of event-reconstruction programs. The problem is to calculate these quantities with the greatest possible accuracy and, ultimately, to reconstruct the topology of events with high resolution and in the shortest possible time, since otherwise the results of the experiment may lose their scientific novelty.

Fitting of tracks

The fitting of tracks and particle decay vertices is an important part of the analysis of experimental data. If the process of track fitting is considered, it basically consists of using statistical methods to process information obtained by means of track detectors. Additional data are also used, for example, the magnetic field strength, calibration constants, etc. As a result of the fitting, the position, direction, and curvature of the track are obtained. If the errors of the measurements and the interaction of the particle with matter are ignored, the measured vector $\mathbf{V}$ is a function of the five original quantities $\mathbf{P}$ (Ref. 27):

$$\mathbf{V}=\mathbf{f}(\mathbf{P}). \tag{1}$$

The function $\mathbf{f}$ is called the model of the track, and it is obtained by solving the equations of motion for the charged particle. The model of the track is uniquely determined by the magnetic field, the details of the track detectors, and the methods used to represent the track parameters. Equation (1) is interpreted as follows. If $\mathbf{V}$ is regarded as a point in an n-dimensional space, then the function $\mathbf{f}$ defines a so-called constraint surface, which consists of all points corresponding to exact solution of the equation of motion. Moreover, in the absence of a magnetic field, the particle trajectory is a straight line, and therefore the track model is a linear function. In a homogeneous magnetic field ($\mathbf{B}$=constant), the trajectory of the charged particle is a helical curve with axis parallel to the vector of the magnetic induction. The points of intersection of the particle trajectory with the planes (cylinders) of the detectors can be calculated analytically in this case. In an inhomogeneous magnetic field, the equations of motion are solved numerically, for example, by the Runge–Kutta method.

Methods of specifying the track model

By the choice of a suitable track model, it is possible to shorten significantly the computing time needed to fit tracks. To this end, the method of track-model parametrization is widely used. This is the name given to an algorithm that makes it possible to express quantities such as the coordinates as functions of data such as the geometrical sizes of the detectors and their positions.[28] Naturally, these functions also depend on the five kinematic parameters. As a result, from M measured coordinates on the track one obtains a much smaller number of parameters N. Such a approach to track fitting gives a gain in time if it is compared with the direct method of finding the track of an individual particle by calculating and analyzing individual coordinates stored on magnetic tape. For the purpose of parametrization, the function $\mathbf{V}$ is often approximated by means of a suitable analytic function. Various track models used in experiments are described in more detail in Ref. 29.

Estimates of track parameters

Tracks of three types are considered in the fitting process. 1. The ideal track, which corresponds to exact solution of the equation of motion for a charged particle. 2. The physical track, which corresponds to the particle trajectory with allowance for multiple Coulomb scattering. 3. The measured track, which is determined by means of coordinates measured by the track detectors. In contrast to the physical track, the measured track takes into account errors introduced in the process of coordinate measurement, and also as a result of multiple scattering. To determine the quality of the fit, i.e., the degree of approximation of the measured to the ideal track, the most common approach is to use a χ^2 estimation function (global method of estimation) and Kalman's algorithm. Both estimation methods are based on the least-squares method. The χ^2 function is calculated by finding the minimum of the expression[30]

$$S(\gamma)=\Sigma[x_i-x_i(\gamma)]W_{ij}[x_j-x_j(\gamma)],$$

where x_i and x_j are the measured (observed) and theoretically calculated coordinates, respectively; W_{ij} is a weighted matrix that is the inverse of the covariance matrix, or the error matrix. However, the χ^2 function gives good fitting results under the following conditions. 1. There must be a Gaussian distribution of the errors in the measured coordinates. 2. It is necessary to make a careful estimate of the weighted matrix. 3. The function $x_j(\gamma)$ must be linear in the parameter γ.

In the fitting process, the value of χ^2 is compared with the theoretical χ^2 distribution with $M-N$ degrees of freedom, where M is the set of measured coordinates x_i and N are the track parameters obtained after parametrization (transformation) of the data obtained by means of the track detectors. As is noted in Ref. 27, a serious shortcoming of the global method of estimation is that the track parameters are estimated at one point of the surface. In the presence of the effect of multiple scattering, the real track can differ appreciably from the ideal track in such a method of estimation. There are many experiments in which more accurate extrapolation and interpolation of the track is needed in the process of joining individual, above all, curved segments of the track.

In several studies it has been noted that the so-called Kalman's algorithm should be useful both for fitting track parameters and for fitting tracks with an estimate of the quality of the fit. This algorithm was originally used to analyze discrete linear dynamical systems. The fulfillment of three procedures is assumed. 1. Filtering (estimation) of the cur-

rent state of a vector on the basis of all previous measurements. 2. Using the current state of the vector, an estimate of the state in the following time interval is predicted. 3. Smoothing of the estimate of the state vector at a given instant in the "past" on the basis of measurements obtained in the "present."

Applied to the processing of track information, Kalman's algorithm has a number of advantages over other fitting methods (Ref. 31). 1. It is possible to perform simultaneously procedures such as track fitting and the search for and reconstruction of tracks. 2. It is not necessary to calculate the inverses of large matrices. Moreover, the computing time is proportional to the number of measurements on the track. 3. The estimated parameters of the track are close to those of the physical track. The essence of Kalman's algorithm for the processing of track information is as follows.[27] If there is an estimate of the state of the vector at the time t_{k-1}, then it is extrapolated to the time t_k by means of the system of equations. The estimate at the time t_{k-1} is then calculated as the weighted value of the predicted state of the vector and the value measured at the time t in accordance with the equations that relate to the measured values. The estimated data can then be applied again to the previous estimates, or the smoothing procedure can be performed on them. Kalman's algorithm has been used to reconstruct complicated events in the vertex detector ZEUS.[32] For track multiplicity 10–20, the track reconstruction can be performed in just 10–20 ms by means of the VAX8800 processor.

Search for and reconstruction of tracks

The fitting of track parameters is a more or less definite process, since one uses a well-developed mathematical formalism. The process of finding "candidates" for useful tracks is more complicated. In contrast to the fitting of track parameters, the various algorithms and methods for track fitting are generally most effectively applied to a specific experiment. We shall consider some of the most characteristic algorithms used for track fitting. These have been most fully described in Refs. 28 and 33.

1. Combinatorial method. The set of measured coordinates, including coordinates relating to background events, are divided into subsets on the basis of definite criteria. Then for each subset a fitting procedure is performed in order to establish whether the data of the fitted subset belong to the trajectory or not. If the solution is satisfactory, the data on the candidate for a useful track are stored. Otherwise, such information is stored in a file designated for background data. The shortcoming of this fitting method is the low speed, which necessitates a large number of samplings even in the case of a low multiplicity. Thus, in the case of five tracks in the presence of ten coordinate planes, up to 5^{10} combinations are needed.

2. Local method. Three modifications are used: a) Track following. This method can be used even in the cases in which processing is done with the participation of an operator who observes candidates for useful events on a display screen. A track segment consisting of three or four points is chosen, and the most suitable tracks are found. The calculation time is proportional to n^2, where n is the number of measured coordinates. b) The "window" method. A window is specified by means of two or three reference points, which are systematically chosen among the coordinates, and then a track model (a straight line or a circle) is interpolated between the points (Fig. 2). One then seeks coordinates that are close to this window. The width of the window is determined by the resolution of the chambers, their efficiency, etc. The calculation time is proportional to n^3. This method was found to be promising, since it was also used to reconstruct segments of tracks by hardware methods (see below). c) Choice of track elements. A candidate for a track is obtained in two steps—a short track segment is chosen, and interpolation or extrapolation along a straight line or a parabola is performed. In a second step, the segments are joined together. This method is convenient in that it simplifies the resolution of the problem of indeterminacies of the "left–right" type and, in

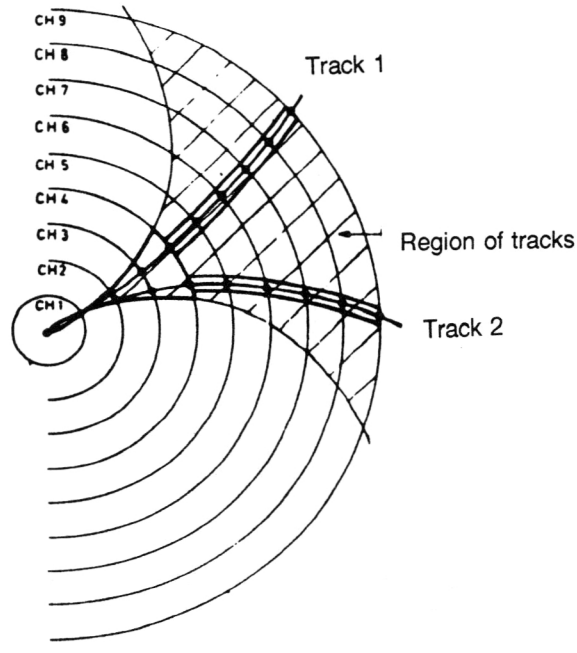

FIG. 2. The "window" method. CH1–CH9 are cylindrical drift chambers.

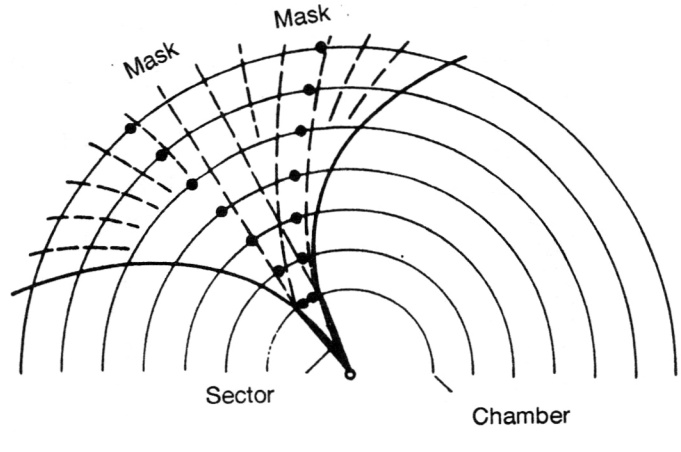

FIG. 3. The pattern (mask) method.

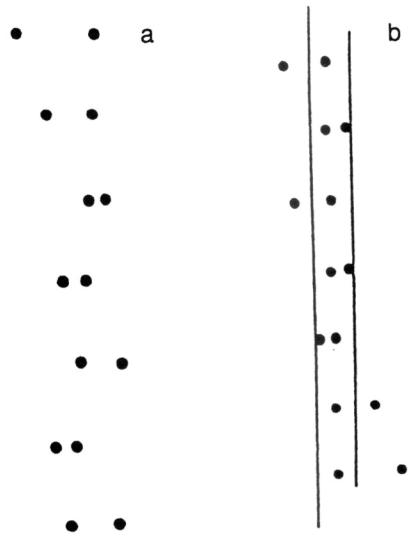

FIG. 4. (a) Straight track detected by means of a drift chamber; (b) the same track identified by the "window" method.

addition, it is well suited to a search for tracks in the case of a high density of measured coordinates; this is the case when three-coordinate detectors are used under conditions of a strong background and high luminosity.

3. Global method. This method gives good results in the analysis of track information with high density and in the process of the identification of events in which a single particle causes the successive firing of several position-sensitive sensors in a plane of a detector. The method is distinguished by a high speed, since there is no need to test a set of combinations of coordinates. Therefore, the global method has many prospects for realization by hardware methods. Here several modifications are also possible. a) Histogramming. The use of such a method presupposes the determination of different functions of the coordinates and their introduction into a histogram. The track positions are then determined from peaks or from clusters in the histogram. It is also noted that this method gives good results if the track model is represented in one or two projections. b) The pattern (mask) method. This method gives good results if cylindrical multilayer drift chambers are used. For realization of the method, it is necessary to have a dictionary (set of patterns) of the most probable useful events (Fig. 3). The method can be effectively realized by the use of content-addressable memories. c) The tree method (well-known from the theory of pattern recognition). This method can be effectively realized if multilayer drift chambers are used in the experiment. The essence of the method is best illustrated by means of a series of figures.[34] Figure 4a shows the trace of a straight track. For clarity, the horizontal scale is much greater than the vertical scale. If the "window" method is used, then, as can be seen from Fig. 4b, some of the coordinates by means of which the "left–right" type of indeterminacy is eliminated remain outside the corridor, since the top and bottom coordinates, situated on the vertical, are chosen to establish the track. It can be seen that tracks with large curvature are not reconstructed. Use of the "tree" method presupposes that the determination of the track is done in several steps by joining individual links (Fig. 5a) (two closest coordinates belong to each link). An individual link may belong to either a straight or a bent track. The connections between several neighboring links determine the "trunk of the elementary tree," which, in its turn, can be regarded as a branch of another tree. The elementary trees are combined in order to form a complete tree, which is a candidate for useful events (Fig. 5b). After suitable tracks have been found, the global method of statistical estimates is used. The results of the processing are illustrated by means of Figs. 6 and 7.

Most of the algorithms used to reconstruct track events require the performance of arithmetic operations, and they are therefore laborious and require much computing time for their realization. For this reason, searches are made for new hardware methods that make such operations unnecessary. A feature of the algorithm described in Ref. 35 is the possibility of reconstructing complicated topologies of events with large momenta fairly quickly and very qualitatively. In the computer memory, 12 832 different samples of track information that can be detected by means of a vector drift chamber are stored. Useful tracks are detected by comparing the data that

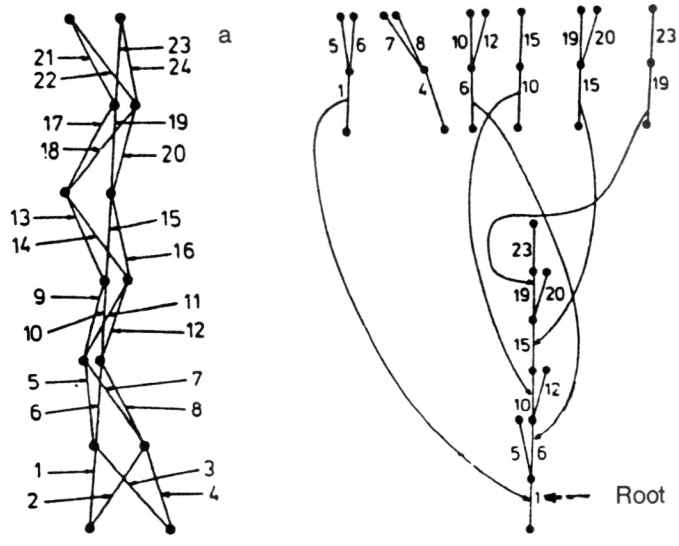

FIG. 5. Track identification by means of the tree method. The numbers give the indices of the links.

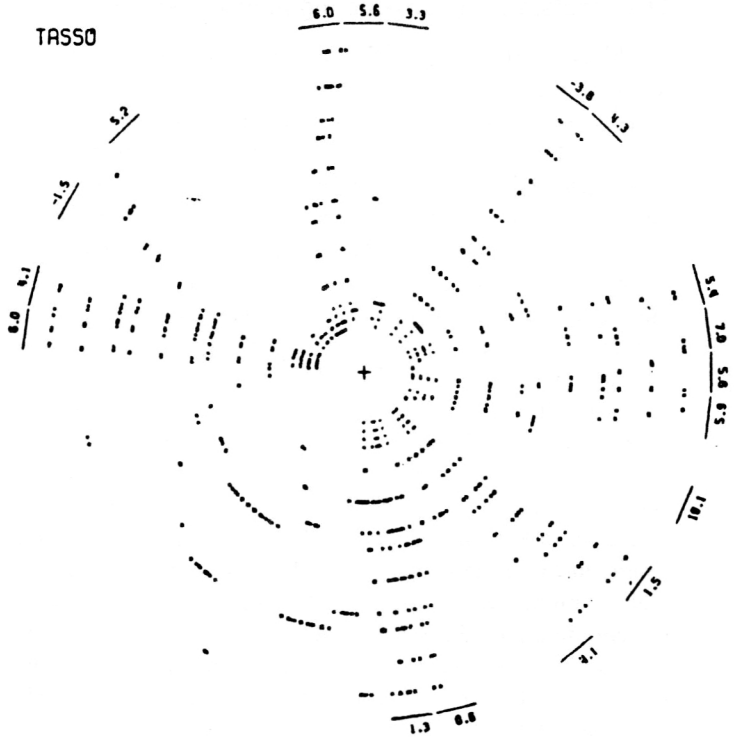

FIG. 6. Real event detected by means of a multilayer drift chamber.

arrive from the detector with the patterns stored in the dictionary (memory). The main advantages of this method of processing information are as follows. 1. Instead of arithmetic operations, simple procedures such as displacement and manipulation on individual data bits are used. 2. The algorithm is well suited to implementation on fast vector machines. 3. The detector geometry (cylindrical or symmetric) is not critical. 4. There is more effective recognition of tracks that pass through several common cells of the chamber (Fig. 8) or in the case of low firing efficiency. This is achieved by storing in the computer memory an additional bank of track segments that is used in the cases when regular situations are not recognized after the first run. However, such an approach is effective in the case of a low (of order 3–4) multiplicity of the events.

Verification of candidates for useful tracks and fitting of decay vertices

The next stage after finding candidates for useful tracks is the procedure to test whether a track is useful. Many al-

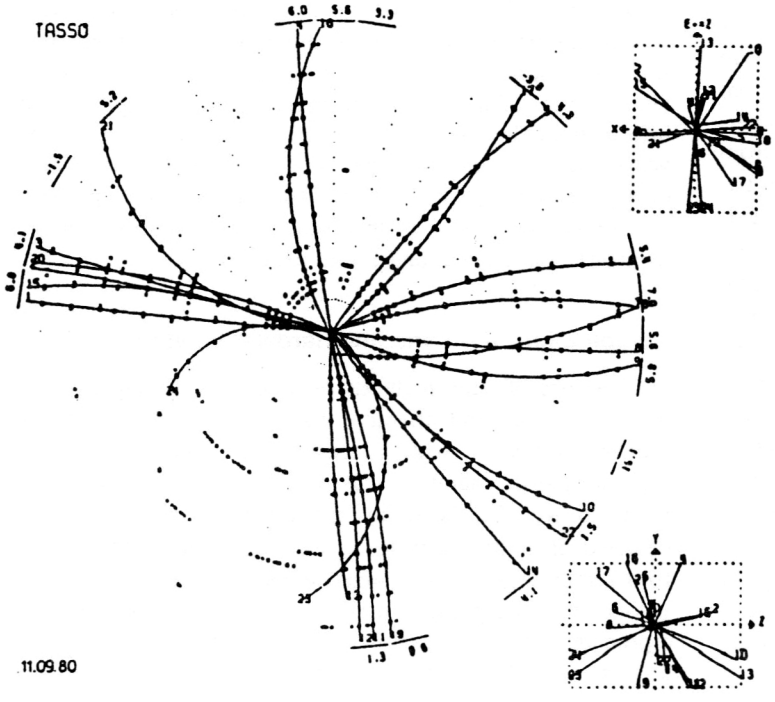

FIG. 7. The same event as in Fig. 6 after reconstruction.

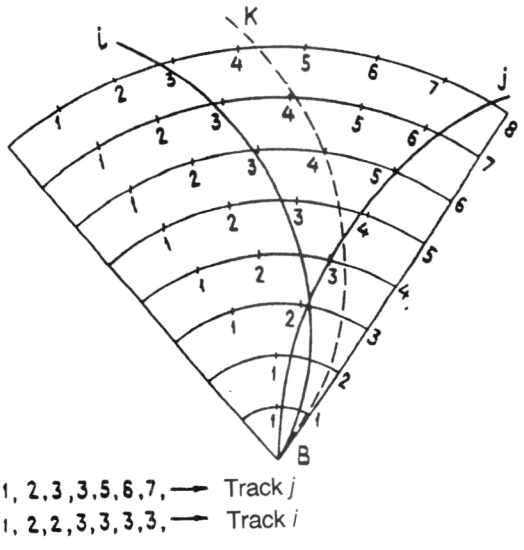

FIG. 8. Application of the method of generated patterns (masks) for the reconstruction of complicated tracks by means of a vector drift chamber. The identification of the track k requires additional verification.

gorithms have been developed to reconstruct useful tracks with allowance for the features of the detectors.[29] In particular, for this purpose one uses fitting to an analytic track model such as a straight line or a parabola, and tracks that have an inadmissibly large value of χ^2 are eliminated. In some algorithms, when it is necessary to recognize and join short curved segments,[36] local testing of the track curvature and its sign is used. If the coordinate of the decay vertex is known, as is the case in experiments with a fixed target, the direction or the shortest distance of the extrapolated track to the decay vertex is verified. In experiments at colliders, in which colliding particle beams are the target, even the coordinate of the primary decay vertex is unknown. Therefore, after the useful tracks have been found, the decay vertices are fitted. This procedure is described in detail in Ref. 37. We shall first briefly consider the process of analyzing an event with one common vertex. The following operations are performed to fit the common (primary) decay vertex, from which m tracks emanate. For each individual track, its parameters and covariance matrix with respect to the exact values are fitted. For m tracks, $5m$ fitted quantities are obtained, and these are regarded as virtual measurements with a known covariance matrix, in which only five parameters are correlated with the given track. The vertex from which the tracks emanate is assumed to be calculated if the values of three coordinates in space and three momenta (curvatures) for each track are known. This parameter is calculated in two steps. First, fitting is done in order to find the point closest to the m reconstructed tracks, and then the different weights of the track parameters are also taken into account. In the second step, the track parameters are modified in a definite manner with allowance for the value of the covariance matrix, subject to the condition that the vertex is a common point of the m tracks.

The problem of reconstructing multivertex events

Since short-lived particles have ranges of a few hundred microns, the problem of reconstructing events with many vertices is not simple even by means of on-line program methods. First, micron accuracy in the determination of the coordinates of the primary decay vertex and the secondary decay vertices is required. Second, the precise number of tracks that emanate from the decay vertices is not always known. In addition, there may also be neutral particles among those that decay. For the reconstruction of such events, it is also necessary to know the size of the beam and its exact position.[27] In addition, as the detected multiplicity increases, the time needed for the computer calculations rises sharply (to the third power of the multiplicity). Questions relating to the use of Kalman's algorithm to determine the coordinates of the primary and secondary decay vertices are discussed in Ref. 38. It is noted that this method gives good results in experiments with a fixed target with heavy ions at high multiplicities of order 100–150 for the determination of the coordinate of the primary decay vertex. The amount of computing time required is proportional to the track multiplicity. For the reconstruction of multivertex events, Kalman's algorithm is used to process data obtained by means of microvertex detectors in the DELPHI facility. Considered briefly, the reconstruction of the decay vertices is achieved by means of the following procedures. First, all the tracks are fitted, and the primary decay vertex is approximated. Each track is then estimated by means of the χ^2 function, and a sorting is performed on the basis of three criteria: "good," "indefinite," and "bad" (with respect to the primary decay vertex). After an additional test, some of the tracks of the second type may become "good." The remaining tracks together with the "bad" tracks are then fitted to find a secondary decay vertex, etc.

To conclude this section, it should be noted that there do not exist universal on-line program methods for processing physical information, and it is therefore difficult to recommend which of them is the most promising, since hardware processing methods can introduce their own corrections. It is for this reason that it is generally necessary to consider the effectiveness of the various methods of processing physical information. However, the experience accumulated during 25 years can be successfully used in conjunction with the details of a particular experiment as the basis for developing effective processing methods that satisfy the rigorous requirements of both current and future experiments.

Combination of on-line and hardware method

The use of such methods presupposes the implementation of original and effective algorithms. A promising approach associated with processing of data obtained in multilayer drift chambers was developed in Ref. 39. In the review of Ref. 22, the author noted that to reconstruct tracks detected in multiwire proportional chambers in the Rutherford Laboratory[40] a specialized algorithmic device was developed that is capable of performing rapidly (during 200 ns) a procedure often encountered in a track-fitting process such as $R=MX+C$, where M, C, and X are 16-bit numbers. In

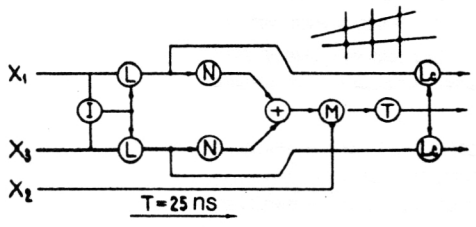

FIG. 9. Example of a computational module of pipeline type for three planes. x_1, x_2, x_3 are coordinates; I is a generator of binary indices; L: $x_i \rightarrow x_I$; N: $x \rightarrow Ax+B$; + is an adder; T: $x \rightarrow T(x)$ (operation with tables); L_c: strobing to memory; M: path mapping operation.

the considered study, the track-information processing is done by means of relatively simple processors by the pipeline method and with a high degree of parallelism. As computational blocks, one uses memory modules, shift registers, etc., by means of which the simplest subprograms, branching operations, indexing, and transmission to the next processing level are implemented. As an example, Fig. 9 shows part of the computational module intended for the processing of data obtained from three planes. It should be noted that the detection and processing of the track information are done in real time.

Investigations are made with the aim of using the effective algorithms employed in information theory for pattern recognition. The essence of the problem is that in the experiments that will be performed in beams of high-energy particles by means of high-precision detectors the physical information can be regarded as a discrete image of a complicated picture (Fig. 10). The possibility of using the method of coordinate transformation proposed by Hough to construct a track processor was demonstrated in Ref. 41. This method is also often called the method of histogramming with variable slope. The essence of the method is that any curve belonging to some pattern in a Cartesian coordinate system can be described by a set of points $[x_1, y_1, \ldots (x_n, y_n)]$ or by a parametric representation of the

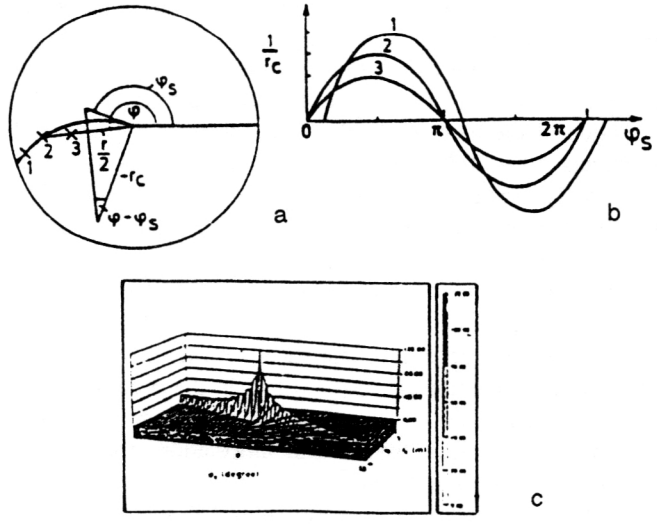

FIG. 10. (a) Mapping of the parameters (r, φ) and $(1/r_c, \varphi_s)$; (b) Hough space; (c) track in the $(1/r_c, \varphi_s)$ plane.

curve on a multiparameter surface. Different parametrizations of the curve can be used to establish the connection between the Cartesian coordinates and the parameters of the surface. It is important that the given parametrization for a given curve is unique. This means that a given curve in the Cartesian coordinate system is represented by a point on the parametric surface, and a point in the Cartesian coordinate system is represented by means of a certain curve (surface) on the parametric surface. This dual relationship has the consequence that the system of points of the curve in the Cartesian coordinate system can be mapped to a set of curves (surfaces) situated on the parametric surface with a common point of intersection, which represents the parameters of the curve in the Cartesian coordinate system. This property of the Hough transformation can be effectively used to recognize curved lines on the background of a complicated image. Data on the parametric surface are stored in a matrix of accumulators (Hough matrix), and the value of each point (x, y) of the original image is incremented in the system of accumulators in the Hough matrix in accordance with the chosen parametrization. The existence of a local minimum after the processing of the Hough matrix indicates the presence of a curve of interest to us in the original image. The practical realization of such a method presupposes fulfillment of the following procedures. 1. Choice of an appropriate parametrization of the curve. 2. Specification of the quantization value of the surface parameters. 3. For each point of the image (x, y), a system of parametric points on the multidimensional parametric surface is calculated in accordance with the chosen parametrization and is added to the contents of the matrix of accumulators. After the calculation of all the points, the Hough matrix is calculated in order to obtain data on the curves. The cited paper also gives a description of a processor by means of which the processing time for 1000 points of an image corresponding to 10 tracks (128×128 on the image) is 6.45 ms for the first track.

The Hough transform found actual application in the reconstruction of tracks of particles detected in the OPAL detector.[42] This detector is divided into 24 sections. Each track i is specified by a set of coordinates $(r, \varphi)_i$. Since the useful tracks emanate from a decay vertex, each of them is determined by a starting angle φ_s and a radius r_c. Thus, together with the coordinates of the decay vertex each pixel (r, φ) corresponding to an event detected in the detector determines a class of tracks with different φ_c and r_c. From Fig. 10a one can obtain the equation

$$\frac{r/2}{r_c} = \sin(\varphi - \varphi_s).$$

The histogram method used to identify particle tracks is based on accumulation of a number of pixels belonging to a given track (φ_s, r_c). As can be seen from the above relation, each pixel (r, φ) is mapped to a sinusoid with amplitude $2/r$ and phase with shift φ. Such a transformation is a specific type of Hough transform. Applying this transformation to each pixel of the detector, we obtain the curve corresponding to it in the $(1/r_c, \varphi_c)$ plane, as shown in Fig. 10b. Moreover, the pixels belonging to the given track with parameters $(1/r_c, \varphi_s)$ have a common point of intersection. Use of the

Hough transform gives a high probability of the presence of tracks. Figure 10c shows a representation of an ideal track in the $(1/r_c, \varphi_s)$ plane.

In connection with the development of optoelectronic methods of data processing, interest attaches to the method proposed in Ref. 43 for looking for track information based on Fourier transformation with a filter or correlation function. The use of such a method makes it possible to use vector or optical processors to reconstruct tracks.

4. HARDWARE REALIZATION OF THE TREE ALGORITHM

The tree algorithm, which is successfully used to reconstruct tracks by means of a computer, has also proved to be effective for the construction of parallel specialized processors. As was already noted above in concrete examples, the image of an event detected in three-coordinate gas and other detectors has a similarity to a tree or a neural network. With the development of semiconductor technology, the technique of integrated microcircuits has achieved great successes, and by means of such circuits it is possible to implement a high degree of parallelism in the execution of logical and arithmetic operations. This makes it possible to implement rapidly effective algorithms such as the tree method and the pattern (mask) method. These algorithms were successfully realized in Ref. 44 and in other studies to which references are given below. The method is distinguished by its universality and can be used to process track information obtained in multilayer drift chambers, microplane semiconductor detectors, etc. It is assumed that the detector consists of a definite number of layers, each of which, in its turn, is divided into a given number of "bins." A charged particle that passes through the detector initiates a signal in only one bin per layer. In such an approach, the problem of reconstructing the tracks of particles reduces to a search for coincidences between the tracks of an event and patterns stored in the computer memory. This data bank has the structure of a tree. The basic idea is to apply the method of successive approximation in conjunction with the tree algorithm to events that are "scanned" successively with different values of the spatial resolution. In the practical realization of the algorithm, such an approach makes it possible to reduce the amount of electronic equipment. The resolution is improved by combining neighboring bins in OR circuits. Figure 11 shows an event with four tracks passing through four layers of the detector. Between the top and the bottom, the resolution of the detector is improved by a factor 2. It can be seen that as the resolution is improved the efficiency of track recognition is, as one would expect, significantly raised. Figure 12a shows how a single track is scanned under the condition that each layer of the detector consists of two bins.

For the recognition of such events, it is sufficient if the memory contains eight patterns. If, however, the resolution is increased by a factor 2, one obtains the arrangement shown in Fig. 12b, in which pattern 3 shown in Fig. 12a is now represented by two subpatterns, etc. With increasing resolution of the detector, the number of patterns kept in the memory increases strongly. The question now arises: How should the memory bank be actually organized, and what

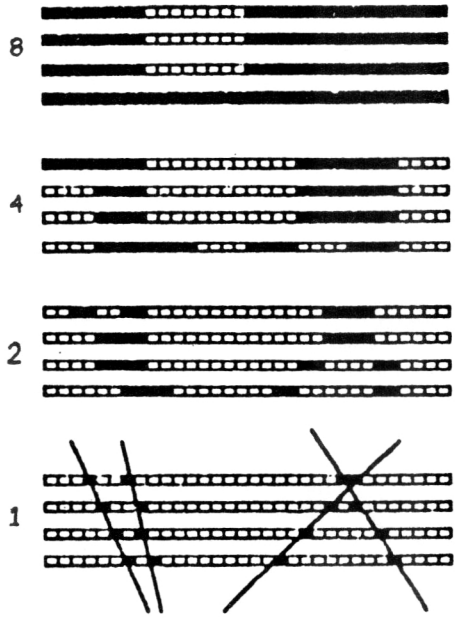

FIG. 11. Form of one four-prong event as a function of the detector resolution.

type of memory is it best to use? As is shown in the present paper, the process of comparison of a real event with patterns is most effectively realized by means of the tree algorithm, and the memory bank must have a hierarchical structure of a certain depth L, as shown in Fig. 13, in which $L=5$. The mean number of patterns is calculated by using the expression

$$N_m = k \log_2 n,$$

where n is the number of bins in a plane (layer) of the detector, and k is the mean number of patterns "ascribed" to a tree junction. As can be seen from Fig. 13, the number of patterns increases exponentially with increasing depth in the tree. To raise the efficiency of the search, it is proposed in Ref. 45 to divide the depth of the tree into two parts, and in the lower part of the tree, where the required memory capacity is not so great, to use an associative memory, since this makes it possible to shorten appreciably the event reconstruction time. This is possible not only on account of the parallel addressing of the memory but also by virtue of the selection of the more probable paths for searching for track segments in the following processing level, as is shown in Fig. 14. The realization of such an algorithm is facilitated by the fact that an associative-memory module has been developed for these purposes.

It should be noted that the use of associative memories in physics experiments began already at the beginning of the sixties. Methods of sorting statistical information by means of associative memories are described in the review of Ref. 46. A further review[47] considers methods of associative search for information by using memory address systems in the construction of multidimensional pulse analyzers to accumulate information in a wide range of indicators. The monograph of Ref. 48 is recommended to the reader with a more detailed interest in the principle of construction of as-

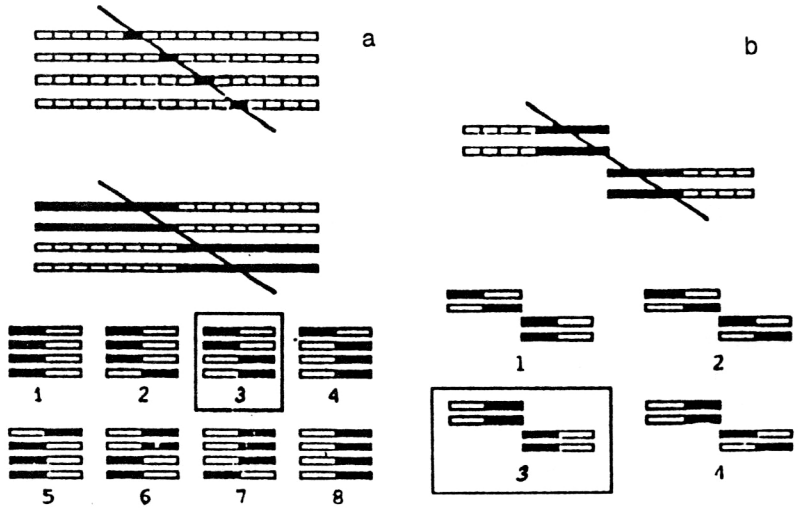

FIG. 12. (a) Isolated track detected in a plane containing only two bins. Eight matching patterns are needed for one straight track. (b) Picture of the event represented in Fig. 12a with twice the resolution.

sociative (content-addressable) memories. For the processing of track information, the main requirements on such a memory are as follows:[49] a) the capacity of a cell must be sufficient to store at least one pattern; b) there must be a logic device for comparing binary codes. Each row in the memory array represents one cell of an associative memory, and therefore an image of the event is contained in it. Each cell[50] contains several words (one word per plane or layer of the detector). In its turn, each word contains the address (coordinate) of a position that has fired in the plane (layer) of the detector. Thus, all the words stored in one cell determine a pattern corresponding to the firing of one coordinate in each plane (layer) of the detector. By means of a data bus, the data corresponding to a given plane are used at the start of operation to store the patterns. The search for and detection of an event are done as follows. All the coordinates measured in different planes are sent to data buses corresponding to them, and the comparison operation is then performed. In the case of a positive answer, the corresponding trigger is activated, and the addresses of all the candidates for useful tracks are successively read out to the output bus.

In order to accelerate this operation, a priority coding register is used. Calculations show that a memory of capacity 100 kbyte is needed for a typical experiment. To obtain such a capacity, a computational network consisting of similar modules that each has 256 32-bit cells is constructed.[51] In Ref. 52 there is a description of a three-level trigger system intended for the selection of track information obtained in a microstrip vertex detector (Fig. 14). The use of associative memory makes it possible to detect candidates for useful events very rapidly and with high efficiency. It is sufficient to note that data are fed into the memory through the inputs at the frequency 25 MHz, and the comparison time is a few nanoseconds, so that information appears at the outputs with frequency 20 MHz. Each of the memory modules is connected to five detector planes (4096 strips). It is noted that the detection of one event consisting of 20 prongs does not require more than 2 μs!

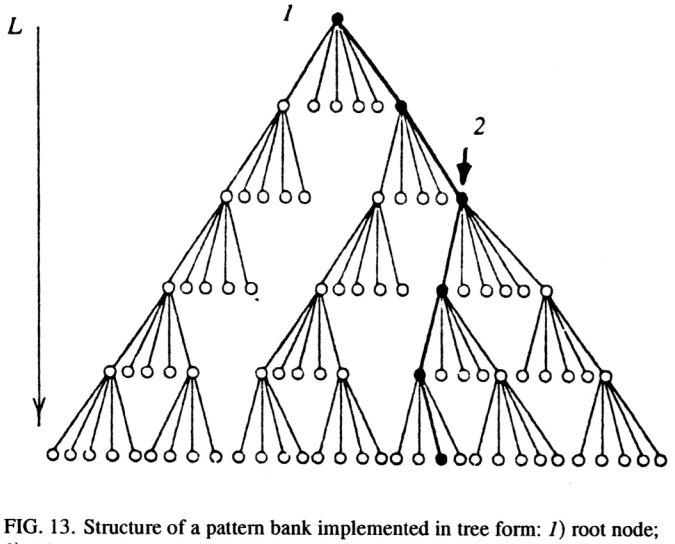

FIG. 13. Structure of a pattern bank implemented in tree form: *1*) root node; *2*) subtree; L is the depth of the tree.

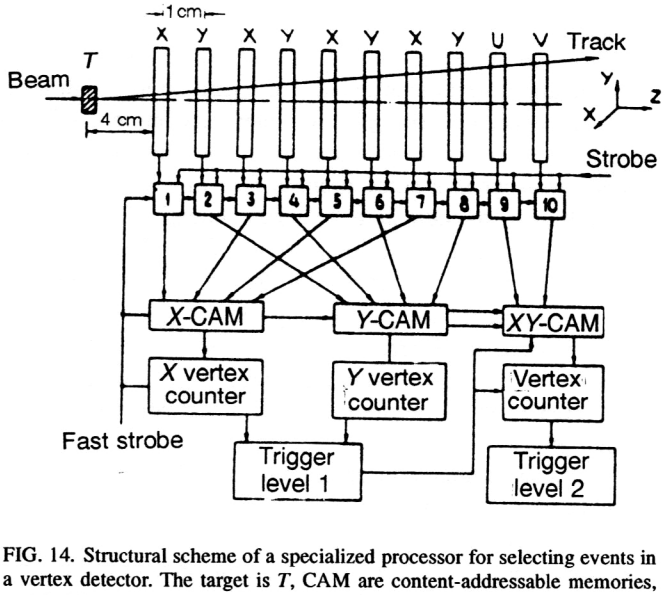

FIG. 14. Structural scheme of a specialized processor for selecting events in a vertex detector. The target is T, CAM are content-addressable memories, and 1–10 are discriminator amplifiers.

5. USE OF NEURAL NETWORKS

General issues

A shortcoming of most of the algorithms used to filter events is that they are adapted to the solution of specific problems dictated by the geometry of the detectors and the formulation of the physical problem; for this reason, it is difficult to modify them for the solution of other problems. In addition, the high accuracy inherent in modern track detectors and the existence of large amounts of digital data lead to the necessity to check a large number of combinations of numbers before a useful track segment or image of a complete event is found. In recent years, there has been a tendency to unification of both the algorithms and the hardware for the filtering of track information. The main attention is devoted to a high degree of parallelism in the search for tracks in both time and space, and also the representation of the initial data in analog and digital form. One of the promising approaches of this kind is the use of the theory and practice of cellular automata and neural networks, which are typical parallel computing systems. Several investigations have been devoted to these questions. Three parameters characterize the degree of parallelism of data processing: 1) the number of processors; 2) the computational power of the processor; and 3) the degree of linkage between them. To solve the problems of looking for and reconstructing particle tracks, it is sufficient if the processor elements are as simple as possible, are fast, and require a low power. Several investigations have been devoted to the possibility of using neural networks and cellular automata, which are known in computational technology for processing track information in experiments in high-energy physics. It has been shown that there is not only a certain superficial similarity between the simplest models of neurons and the topologies of typical events but also, in view of the achievements of semiconductor technology, a real possibility of creating effective processors for the recognition of complicated physical events. According to current ideas, the brain is a perfect computer, but unlike a computer it does not process numerical data, represented most often in the form of binary digits, but processes complete patterns. In other words, a "pictorial logic" is realized in the brain, and this increases greatly both the speed of calculations and the efficiency of operation at a high level of disturbance. For example, it "interprets imprecise information received from the senses at an incredible rate. It discerns a whisper in a noisy room, a face in a dimly lit alley, and a hidden agenda in a political statement" (Ref. 55). Readers interested in the foundations of the theory of neurons are recommended the book of Ref. 53. Some interesting papers devoted to brain psychology and the training of neural networks were published in the well-known journal *Scientific American*.[54,55] In the Russian literature, the review of Ref. 56 is devoted to the application of neural networks in experimental physics.

Cellular automata differ in a number of ways from neural networks. 1. In the first place, they are devices based on discrete operations, and therefore logic elements are used to construct them. 2. Each cell of an automaton is linked to only a limited number of neighboring cells. 3. As a rule, cellular automata are operated in a synchronous regime (timing pulses are used).

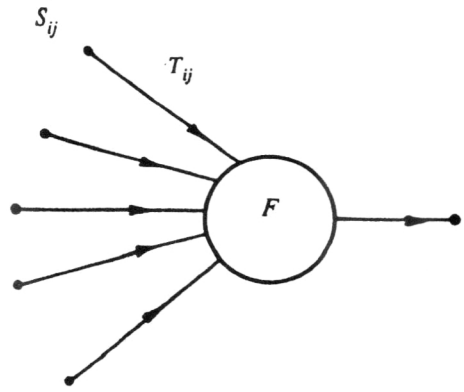

FIG. 15. Simplified scheme of a neuron. The inputs are S_{ij}, and T_{ij} are the input weights.

A more detailed description of homogeneous computational media can be found in the monograph of Ref. 57. The proceedings of Ref. 58 are devoted to the present status and development of an important trend in modern information theory—the use of the theory of neural networks to construct computers. Some theoretical questions are also considered in the papers cited below. We note that a neuron (Fig. 15) is an element of a computational technology that possesses the following properties: 1. It has n inputs S_j and one output. 2. Each input and the single output can be in only one of two states: excited or unexcited. 3. The nerve fibers from some input can bifurcate, but they cannot be joined to the fibers of a different input. 4. The fibers can be excited or inhibited. A neuron fiber can also block a signal traveling along a different fiber. 5. Signals can pass through a neuron only in one direction. 6. There is a certain delay in the transmission of a signal from an input to the output of a neuron. 7. In regular operation, a neuron is excited if the algebraic sum of the exciting and inhibiting signals exceeds a certain definite threshold. 8. Each neuron input has a weight specified by a connection matrix.

Neural networks are made from individual artificial neurons. For example, in a neural network that is operated at discrete instants of time the state S of the ith neuron at the time $t+1$ is determined by the equation

$$S_i(t+1)=g[\Sigma T_{ij}S_j(t)],$$

where g is a threshold function, and T_{ij} is the connection matrix. The solution of a particular problem is achieved by modifying the connection matrix. The main difficulty is that the adjustment of the neural network and its training is a laborious and lengthy process, since for N neurons there are N^2 optimal parameters.

We consider briefly the basic parameters and relations that characterize a neural network.[59] Such a network consists of N binary neurons $S_i=\pm 1$ and synapses T_{ij} (i, $j=1$, 2, ..., N), by means of which the neurons are connected to each other. The sign of a connection T_{ij} can be either positive or negative, and in most models it is assumed that $T_{ij}=T_{ji}$. The

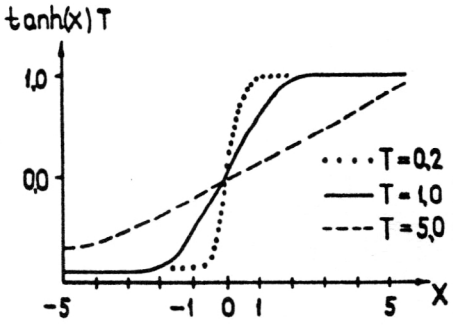

FIG. 16. Form of the sigmoid function at different temperatures.

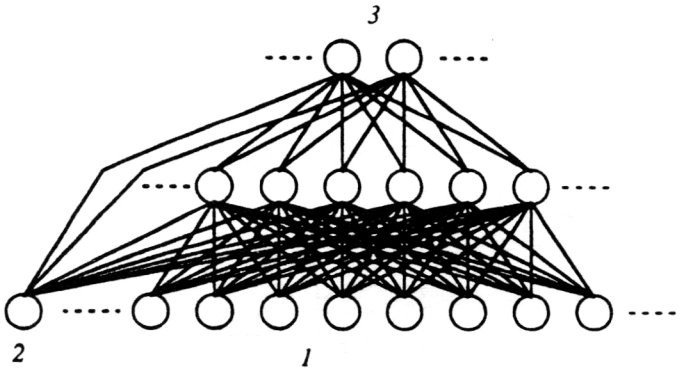

FIG. 17. Simplified scheme of a three-layer neural network without feedbacks: *1*) inputs; *2*) bias element; *3*) outputs.

dynamics of the processes in the neural network is estimated by means of a relation that has a local nature:

$$S=\sin(\Sigma T_{ij}S_j). \tag{1'}$$

A positive value of T_{ij} promotes signal action, while a negative value inhibits it. Further, at a given instant of time the state of the network is specified by means of a vector $S=(S_1, S_2, \ldots, S_N)$, and the dynamics of the system is then determined by means of the matrix T. It turns out that the rule (1′) actually describes gradient decrease of the "energy function"

$$E(S)=-\tfrac{1}{2}\sum_{ij}^{N} T_{ij}S_iS_j. \tag{2}$$

In other words, for initial conditions specified by (1′) there is a local minimum of the energy function. It was noted long ago that there are several analogies between neural networks, statistical mechanics, and thermodynamic systems, and therefore the state vector S satisfies a Boltzmann distribution

$$P(S)=(1/Z)e^{-E(S)/kT}, \tag{3}$$

where the partition function Z is obtained from

$$Z=\Sigma e^{-E(S)/kT}. \tag{4}$$

In Eqs. (3) and (4), T is the noise, and k is Boltzmann's constant. After various manipulations, one ultimately obtains an important equation describing the sigmoid function (Fig. 16):

$$V_i=\tanh\left(\sum_j T_{ij}V_j/T\right), \tag{5}$$

where V_i is the average "temperature" of the S_i. It should be noted that the behavior of an individual neuron is also described by a similar function. Thus, the behavior of the neural network at the "temperature" T can be emulated by means of the relation (5), and it is solved by iteration until the system being trained (by the choice of T_{ij}) becomes optimal and stable. For the processing of track information, one uses more complicated expressions for the energy and the sigmoid function, which are selected with allowance for the detector geometry, the length and curvature of the track segments, their density, etc. Some terms with negative sign are also introduced.

The use of neural networks

Irrespective of the problem that is to be solved by means of a neural network, the form of the sigmoid function is practically the same, and it is only the coefficients in the linear combination of the input variables and the type of connections that make it possible to solve the problem of recognizing particular patterns. The form of the sigmoid function is characterized by the presence of two plateaux at the beginning and end and a linear section in its middle. The complexity of a neural network is determined not only by the number of neurons but also by the number of layers. Three-layer neural networks (Fig. 17) have been used in practice, both without feedback and recursive networks with feedbacks (Fig. 18).[60] A three-layer network consists of an input layer containing an individual neuron to specify a bias voltage in the training process, an intermediate layer, and an output layer. The principle of associative memory is realized in a neural network. It is readily noted that in nuclear electronics there are several distant analogies with a formal neuron such as analog adders and discriminators, coincidence and anticoincidence circuits, and so on. A more complicated example can also be given. Figure 19 shows the circuit of a scintillation hodoscope with associative memory.[60] Since individual signals are developed at the outputs of the discrimi-

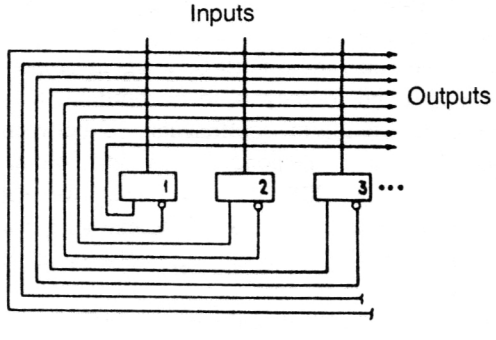

FIG. 18. Structural scheme of an artificial neural network with feedbacks. *1–3*) are amplifiers ($V_i \to U_i$), $V_i=\tanh(U_i/T)$, and the open points are resistors [T_{ij}^{+}, T_{ij}^{-}].

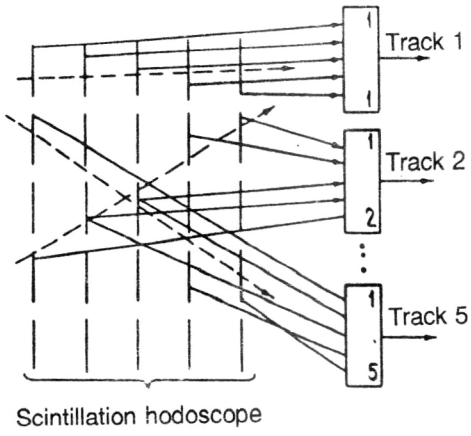

FIG. 19. Scheme of a neuronlike scintillation hodoscope. *1–5*) are OR elements.

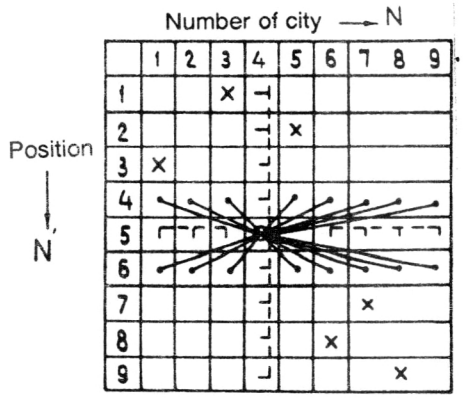

FIG. 20. Illustration of the problem of finding the optimum path between cities. The continuous lines are allowed connections; the dashed lines are forbidden connections. The connections of neurons (*4, 5*) are denoted by ●, and the other connected neurons are denoted by the symbol ×.

nators for each of the tracks, the task is to develop computational networks of elements containing tens or hundreds of thousands of formal neurons.

It should be mentioned that industry has begun to sell rather complicated neural-like integrated circuits. Two devices by means of which it is possible to model the operation of simple neural networks are described in Ref. 61. These devices consist of an array of thin-film resistive neurons with continuous readout and an array of programmable neurons (about 3000) manufactured using the CMOS technology. In connection with the large-scale development of optical methods of detection and recording of information, optical neural networks are also of interest. One such experimental network is described in Ref. 61.

Considering problems of the use of neural networks in experimental physics, we can identify two problems: the solution of combinatorial problems and the processing of track information[62] (although the second problem can also be reduced to combinatorics). An analog of the first problem is the problem of finding the shortest path between N cities (Fig. 20).[63] Along the horizontal in Fig. 20, the positions of the neurons in the array determine the number of the city, and along the vertical the numbers of the cities that are to be visited are plotted. The continuous lines show all possible shortest connections of just one neuron with coordinates (4, 5) and with connection coefficients proportional to $(A-d_{ij})$, where d_{ij} is the distance between cities i and j, and A is a positive constant. If neuron i has the value E_i at the output, then at the input of neuron j there must be a signal proportional to $(A-d)F_i$. If the journey is to be optimal, it is necessary to introduce prohibition coefficients $-B$ between each neuron placed in the corresponding row and column (the dashed lines in Fig. 20). If in Eq. (2) we set $T_{ij}=(A-d_{ij})$, then the minimum value of the energy E for the real journey will occur when the sum d_{ij} is also equal to the minimum value.

In Ref. 64 it is shown how a neural network can be used to solve correctly the problem of identifying a pair of photons produced by particles such as π^0 or η under conditions of strong disturbance. Figure 21 shows combinations for hypothetical events with multiplicity $t=6$. In this example, three pairs of calculated invariant masses correspond to π^0 particles, and two to η particles. The boldface symbols identify true photon pairs, while the remaining pairs correspond to background events. The numbers in the squares give the relative probabilities indicating the extent to which the pair is true. In the neural network for each combination of a pair of photons a corresponding stimulation at the neuron input is ensured, and the neuron outputs are connected to the inputs of other neurons in such a way as to ensure that the necessary network parameters are obtained. A fundamental distinction of neural networks is that the data at the inputs and outputs are represented basically in a natural analog form. In addition, a neural network is capable of "learning." The results of such "learning" in the case of the recognition of tracks obtained in multidrift tubes are given in Ref. 65. A network without feedbacks was used as a model. The problem of finding inclined tracks and the points of their intersection was posed. It was shown that the employed model of a neural network permits the following. 1. It can be adjusted to look for individual inclined tracks. 2. It can find decay ver-

$\gamma_1=$ / $\gamma_2=$	1	2	3	4	5	6
1	\	—	—	—	—	—
2	**π^0 0,74**	\	—	—	—	—
3	0,1	0,1	\	—	—	—
4	**η^0 0,36**	0,1	**η^0 0,36**	\	—	—
5	0,1	0,1	**π^0 0,83**		\	—
6	0,1	0,1	0,1	0,1	**π^0 0,97**	\

FIG. 21. The possible $\gamma\gamma$ interactions for a hypothetical event with multiplicity $t=6$. The boldface denotes true photon pairs.

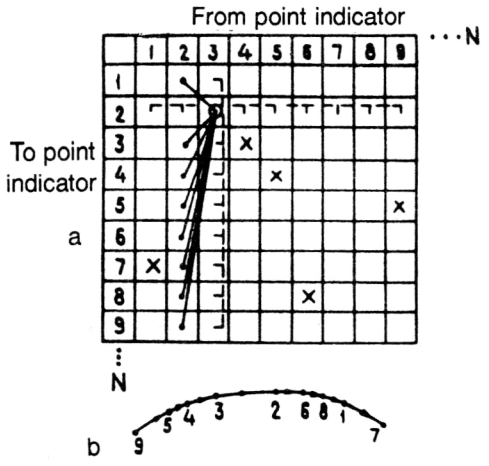

FIG. 22. Explanation of the principle for finding track segments by means of a neural network: (a) the 3→2 neuron connections are denoted by ●. The allowed connections are shown by continuous lines, and the forbidden connections by dashed lines; (b) the track segment 9→5→4→3→2→6→8→1→7 is shown in the upper figure by the symbol ●, while the other connections are denoted by the symbol ×.

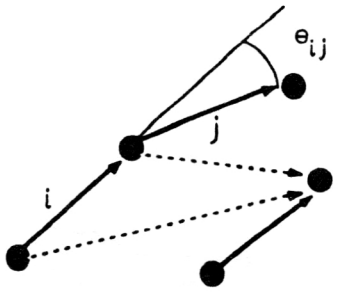

FIG. 23. Determination of segments and of the angle between them.

tices for two-track events on a noisy background. 3. It ensures the high accuracy inherent in multidrift tubes.

Results on the adjustment of a three-layer neural network for the detection of the coordinates of a primary decay vertex by means of a multilayer drift chamber are given in Ref. 66. The construction of modern three-coordinate gas-filled detectors containing many drift cells and amplification and signal-shaping channels connected to them resembles a neural network fairly closely, and it is therefore very natural to use such networks to recognize complicated physical events. The restricted number of inputs of artificial neural networks leads to the need to divide the chamber outputs into individual overlapping sections that each contain 18 outputs. For a chamber having resolution 500 μm with respect to a coordinate obtained by measuring the drift time, entirely satisfactory parameters were obtained for each subsection: single-track resolution 0.72 cm and two-track resolution 2.1 cm. By the combination of data from all subsections, the satisfactory accuracy 1.4 cm of detection of the decay-vertex coordinate was obtained. Figure 22 illustrates the methodology of the search for track segments by means of a neural network that is an analog of the method considered above for searching for the shortest path between N cities. In the first place, certain restrictions are imposed on the length of track segments: $L<R_c$. The value of R_c is several times smaller than the distances between two spatial points. As a result, each point can be assumed to lie on a track within a radius less than R_c. The solution of the problem then consists of joining all the track segments during the adjustment of the network. A candidate for a useful track is taken to be a track if it consists of a continuous chain of segments such that one can reach its ends from any point of the track. As additional restrictions that affect the choice of the coefficients T_{ij} there is the condition that only one point of space belong to the given track. The inhibition coefficients are chosen as in the example given in Fig. 20. The coefficients of the connection matrix have the form

$$T_{ij}=\alpha\,\frac{\cos^n\gamma_{ij}}{r_{ij}}\,, \tag{6}$$

where γ_{ij} is the angle between segments i and j, r_{ij} is the length of the vector equal to the sum of the segments i and j, α is a coefficient of proportionality, and n are small numbers (Fig. 23).

As yet, the use of neural networks is still very much at the exploratory stage. In fact, the most impressive results have been published in Ref. 67, which describes an algorithm for the reconstruction of complicated events with multiplicity greater than 100 detected by means of the time-projection chamber of the ALEPH detector. The most laborious process is still the process of training the network, which consists of a number of iterations. In the first place, it is necessary to impose some restrictions on the complexity of the tracks and the number of points on a track. Thus, if the number of coordinates per track is taken equal to 500 (an entirely realistic number), then one needs $2.5\cdot10^5$ neurons possessing more than $6\cdot10^{10}$ connections between them! Therefore, the following restrictions were introduced: 1. Coordinates situated in the corridor $\sigma_{XY}<0.3$ cm and $\sigma_Z<1.0$ cm, where X, Y, and Z are the coordinates of the given track, were taken into account. This made it possible to eliminate about 5% of the coordinates. 2. Restrictions were also imposed on the connections on the basis of the angles between the track segments, etc. The connection matrix (6) was used in the process of training the network model. The results of the training process are illustrated by means of Figs. 24 and 25. Figures 24a and 24b give display pictures of all generated lines for a real $Z^0\rightarrow$hadrons decay in the XY and Z planes, respectively, as obtained after the first training stage before the convergence process. It can be seen that well separated tracks are reconstructed fairly effectively, but there exist corridors of uncertainty, where there are track intersections and background points and segments. Figures 25a and 25b show the same event but after six iterations (convergences). The efficiency of track reconstruction by means of the neural network was compared with the efficiency of the ordinary program used earlier, and the values 99% and 99.7%, respectively, were obtained. It was calculated that the event reconstruction time was an order of magnitude shorter in the first case. The possibility was also investigated of recognizing considerably more complicated hypothetical events with high track multiplicity (more than a hundred), as will be

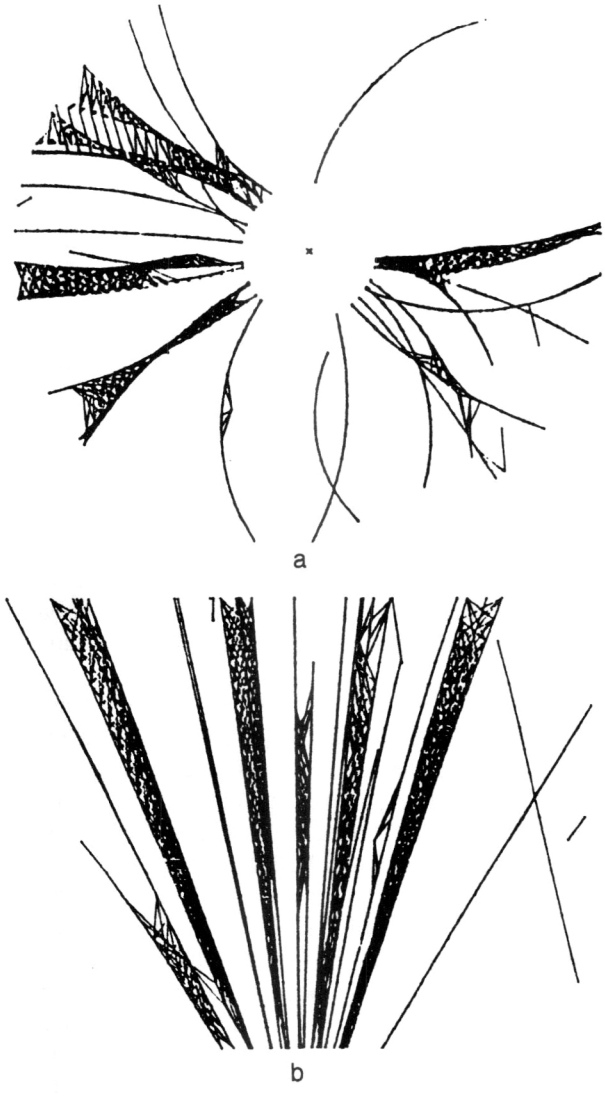

FIG. 24. Display pictures with generated lines for a real $Z^0 \rightarrow$hadrons event in the XY and RZ planes before convergence.

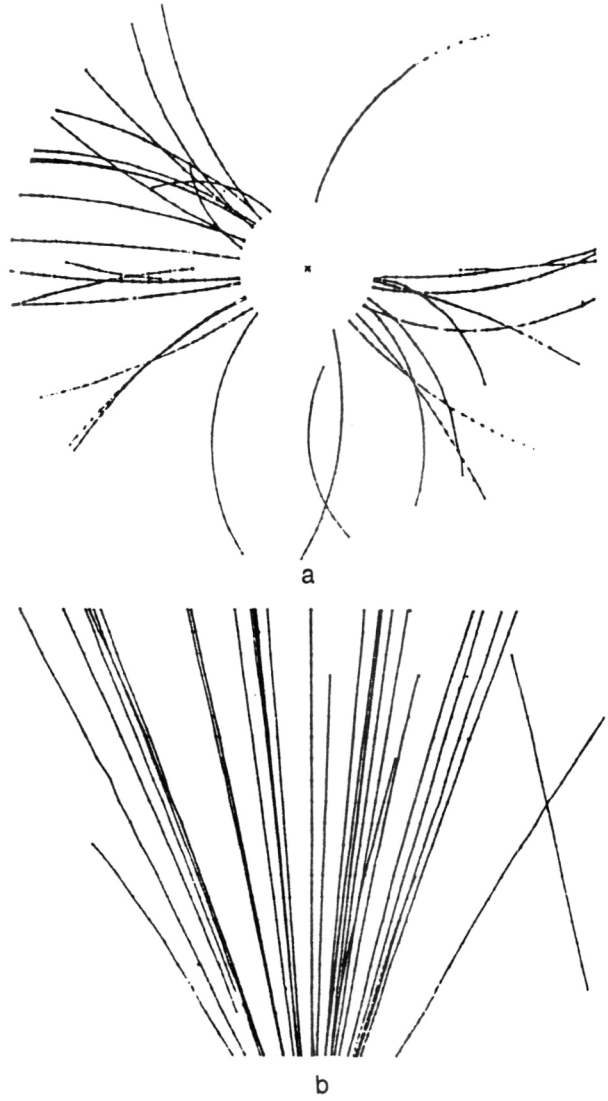

FIG. 25. The same event after convergence by the Monte Carlo method in the XY and RZ projections.

detected on the Large Hadron Collider that is under development.

The possibility of using neural networks in one further important area of elementary-particle physics—the identification of gluon and quark jets—was investigated in Ref. 68. The Monte Carlo method was used to generate the investigated $e^+e^- \rightarrow$hadrons events. Unity at the network output corresponded to a gluon, and zero to a quark. Two programs were used—one to train the network directly, and the other to identify a jet. The recognition efficiency was such that at first 90% of the jets previously not recognized by means of the ordinary program were observed. A similar problem of jet recognition was investigated in Ref. 69 but in $p\bar{p}$ interactions with the aim of identifying decays of Z and W bosons at the level of the QCD background. The information was obtained by means of a calorimeter. However, the problem of recognizing events by means of detectors that are not of the track type is outside the ambit of this review.

It should be noted that in the investigations of the use of neural networks considered above the employed algorithms were used effectively to recognize relatively simple events and were adapted to the use of data obtained by ordinary methods (without the use of neural networks). In other words, the known algorithms for processing track information no longer satisfy the growing requirements of the experimental physics of ultrahigh energies. One of the promising algorithms that, in the opinion of the authors, may give good results at a high multiplicity of events is considered in the next section.

6. ELASTIC ALGORITHM

It is noted in Ref. 70 that the algorithms and methods used to process track information, including the widely used local and global methods, give reasonable results at a relatively low multiplicity and density of the tracks (of order 5–10). There is undoubted interest in the development of an improved algorithm that will make possible the effective use of neural networks for the processing of track information. The algorithm described briefly below was called by the au-

thors the "elastic algorithm for processing complicated track information." It was shown that it can be used to recognize more efficiently complicated events on a background of appreciable disturbance in the case of a high track density and with allowance for errors in the measurement of the coordinates, and that it can also be used to identify the secondary decay vertices that will be observed in experiments in the B factory, in the large collider, and in heavy-ion experiments. The track density ρ_t is defined as the average ratio of the distance between measured points along a track to the distance between points belonging to other tracks or background points.

The elastic algorithm combines a dynamic approach to the elastic network algorithm with the aim of solving complicated problems of the optimization of geometrical problems with global methods of track reconstruction based on the Radon or Hough transform methods employed in the practice of pattern recognition. It is important that the fitting and reconstruction of tracks takes place simultaneously through the solution of the equations of motion of an analogous dynamical system. Mathematically, the Radon transform of the track density $\rho(x)$ is defined by a linear integral over a given class of trajectories:

$$R(\mathbf{x},\mathbf{p})=\int d\tau\rho(r_p(\tau)+\mathbf{x}), \tag{7}$$

where $r_p(\tau)$ is a trajectory defined by the parameter $\mathbf{p}$, so that $r_p(0)=0$, and $\mathbf{x}$ are the track coordinates. If (7) is used for the recognition of physical events in a homogeneous magnetic field, $r_p(\tau)$ describes a helical curve, while in the absence of a field we have straight lines in the direction $\mathbf{p}$. In addition, if the phase surface is represented in discrete form, the Radon transform goes over into the Hough transform considered above, and if each count (pixel) is a binary number, then the number of pixels belonging to the curve is counted. The main shortcoming of this transform is the significant growth in the number of data needed to find the maximum in the process of summing the discrete points as the required resolution and track density are increased. In the new algorithm, the Radon transform is generalized to a dynamical system. In it, R is interpreted as the interaction energy of an external measured (negative) charge density $\rho(\mathbf{x})$ with a positive density of the track T determined from the expression

$$\rho_T(\mathbf{x})=\int d\tau\delta(\mathbf{x}-r_{p_T}(\tau)-\mathbf{x}_T). \tag{8}$$

By restricting the limit of integration to $\tau>0$, $\mathbf{x}_T$ can be interpreted as the beginning of a trajectory with momentum p_T. The following equation for the track interaction energy is ultimately obtained in the considered study:

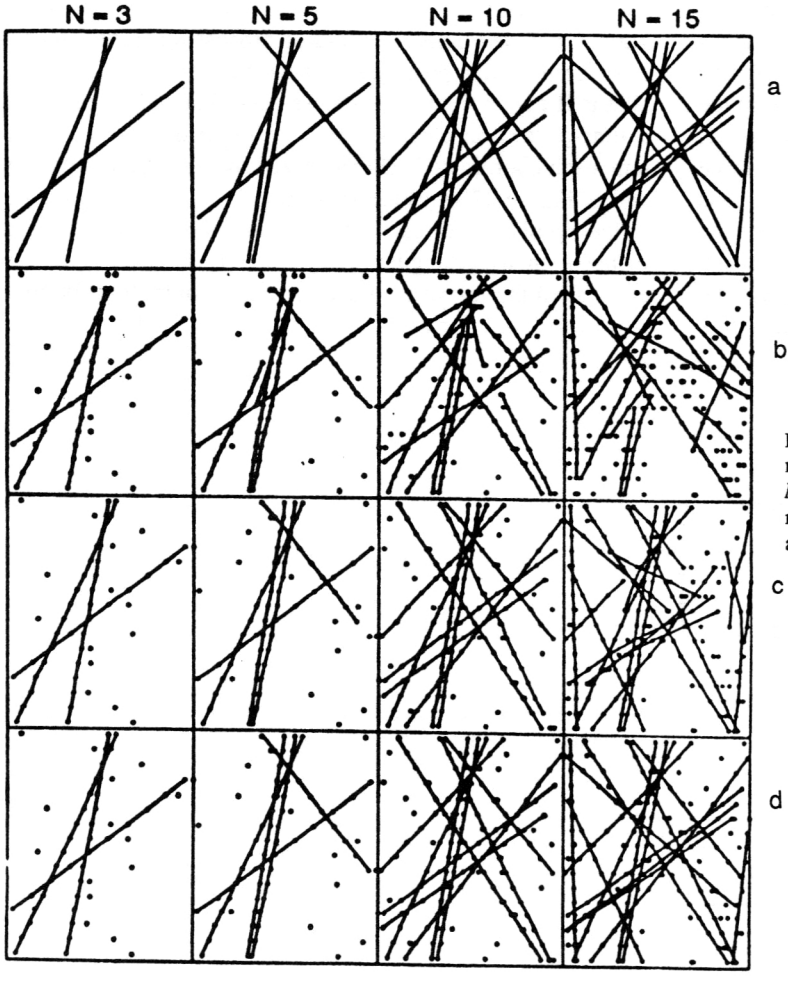

FIG. 26. Examples of efficiency of operation of three methods of reconstructing straight tracks as a function of the track multiplicity N. a) Nominal image of events without background; b) corridor method; c) Denby–Peterson method for neural networks; d) elastic algorithm.

$$R_V(\mathbf{x}_T,\mathbf{p}_T,t)=\int dxdx'\rho(\mathbf{x})V(\mathbf{x}-\mathbf{x}',t)\rho_T(\mathbf{x}'), \qquad (9)$$

where $V(\mathbf{x}, t)$ is a potential that depends on the time t. In the limit $V(\mathbf{x},t)=\delta(x)$, and Eq. (9) goes over into Eq. (7); for static states, Eq. (9) becomes the Hough transform. Thus, the essence of the elastic algorithm is that by the reduction of negative charge to an ionization charge density ρ and of the positive charge to the track density ρ_T the problem of finding useful tracks is transformed into the problem of finding the minimum of an effective interaction energy, which is found by solving the dynamical equations

$$d\mathbf{p}_T/dt=-\lambda\Delta_{p_T}R_V(\mathbf{x}_T,\mathbf{p}_T,t),$$

$$d\mathbf{x}_T/dt=-\lambda\Delta_{x_T}R_V(\mathbf{x}_T,\mathbf{p}_T,t), \qquad (10)$$

where λ is a small estimating parameter.

Concrete examples demonstrate the advantage of this algorithm over the corridor method and the algorithm for processing track information developed on the basis of neural networks and considered above. The elastic algorithm provides an adaptive nonlinear fit to disturbances with a large number of tracks. Unlike Hough transforms, there is no need here for discretization (binning) of the phase surface and laborious searches for a minimum.

Figure 26 gives pictures that illustrate the efficiency of the three algorithms for different track multiplicities with allowance for a 20% background and errors in the measurement of the coordinates with accuracy 3%. The tracks without background disturbance are shown nominally in the upper part of the figure. As the multiplicity N is increased in the limits 3, 5, 10, 15, the mean track density increases as 0.5, 1, 2, 3. It can be seen from Fig. 26 that for $N=3$ all three methods give the correct result. When the track density was doubled, the corridor method for $N=5$ identified one of the tracks as two tracks with different slopes. As N is increased, the efficiency of the corridor method drops sharply. The Peterson–Denby algorithm developed for neural networks and considered above had a failure at $N=15$. The question of realization of the elastic algorithm by means of neural networks is also discussed in the cited study.

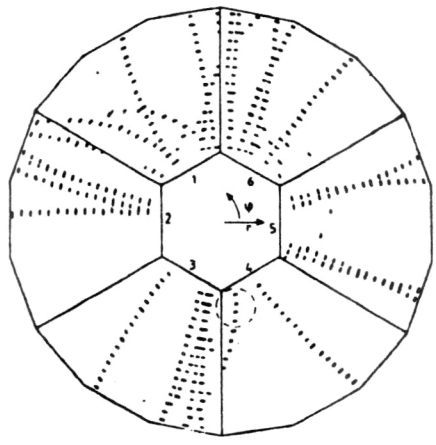

FIG. 27. Projection of an $e^+e^-\to q\bar{q}$ event in the $r\varphi$ plane of the time-projection chamber of the DELPHI facility.

7. THE CONTIGUITY-MASK METHOD

This method is interesting above all because it was developed and used in an actual experiment. Proposed in Ref. 71, the contiguity-mask method and the hardware developed for its realization[72] can, in a certain approximation, be regarded as a neuronlike method of processing track information. A neuronlike processor was used for the first time in the DELPHI experiment[73] to process data obtained by means of a time-projection chamber. In a cylindrical coordinate system, each coordinate is characterized by three quantities: z, r, and φ. Figure 27 gives the projection of an $e^+e^-\to qq$ event in the $r\varphi$ plane. For each of 10 z bins, the r and φ data are mapped on a rectangular program-controlled array that is also called the image memory. As an example, Fig. 28 gives part of the contiguity mask, which has the following proper-

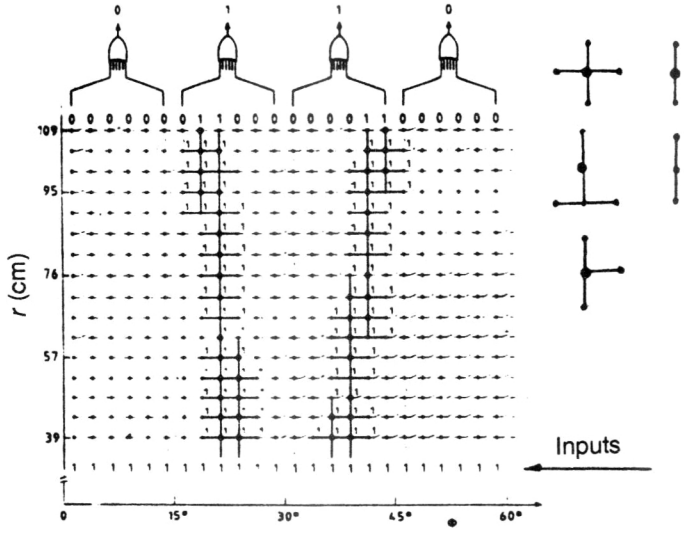

FIG. 28. The contiguity-mask method. The possible connections between neighboring cells are shown at the top right.

ties: Each point has a connection with the neighbor situated below and above and, in addition, on the right side each of the points is similarly joined to the neighboring points. Near each of the points there are vertical and horizontal switches. As can be seen from Fig. 28, some of the switches situated along the track are switched on, since at these points the measured track coordinates are stored. The track is found by sending control signals simultaneously to all inputs of the memory. Figure 28 shows a case in which two tracks were observed, one of which (on the right) was taken but the other was not, since there were no contiguous connections in the mask. Besides the image memory, the processor also has two trigger memories of the same capacity—a vertical commuting memory and a horizontal commuting memory. In conjunction with the image memory, these ensure commutation of the switches in the contiguity mask. The commuting memories are loaded with the contents of the image memory by means of special commands. In addition, there are shift commands, by means of which the contents of the commuting memories can be shifted not only in all four directions but also, if necessary, a cyclic shift can be performed, including a shift around one cell. All these features of the processor mean that it can be adjusted to fulfill many different tasks, for example, search only in the vertical direction, reconstruct tracks of a definite curvature, and guarantee track search even in the cases in which "breaks" in the track arise because of inefficient operation of the detector. The track-search operation is performed in 1 μs. The programmable contiguity mask has a three-step structure. All the blocks except the contiguity-mask block are $n \times m$ arrays (16×24), the elements of which are shift registers.

8. SPECIAL METHODS OF TRACK SEARCH AND RECONSTRUCTION

The fast track processor described above is practically universal and therefore is not tied to a particular detector geometry or physics problem that must be solved. However, universal algorithms require, as a rule, much apparatus for their realization. So far, in current and planned experiments, specialized track processors are widely used. Processors developed specially for a particular experiment and with allowance for the geometry of the employed detectors can solve the posed problems more effectively, and they are therefore of interest for designers of apparatus. As we have already noted, most experiments are divided into two classes: fixed-target experiments and colliding-beam experiments. The first class of experiments can be performed by means of spectrometers based on hodoscope planes (multiwire proportional chambers, microstrip semiconductor detectors, scintillation hodoscopes, and others). For the second type of experiment, more complicated central detectors are built. These are facilities of cylindrical form, through the center of which the ion pipe passes. We shall consider a characteristic example.

The track processor of the OPAL detector[74]

The OPAL facility is a central detector that consists of several detector spectrometers. Around the ion pipe there is a high-precision vertex detector consisting of drift chambers. It has a length of 1 m and is divided into 36 azimuthal sectors. The accuracy of coordinate measurement in the $r\varphi$ plane is 50 μm. The vertex detector is surrounded by a system of large drift chambers, which form a "jet" detector of length 4 m and is divided azimuthally into 24 sectors, each of which contains 159 sensitive wires. The accuracy of detection in the $r\varphi$ plane is 120 μm. The third concentric layer consists of z chambers, which are essentially ordinary drift chambers. The z coordinate is determined by the charge-division method. All these detectors are inside a magnet, outside of which there are time-of-flight counters, an electromagnetic calorimeter, a muon detector, and more. Since the time between bunches is 22 μs, this parameter determines in the first place the operating rate of the trigger system. To look for tracks in the rz plane, the histogram method, realized by hardware, is first used. If a track emanates from the primary decay vertex,

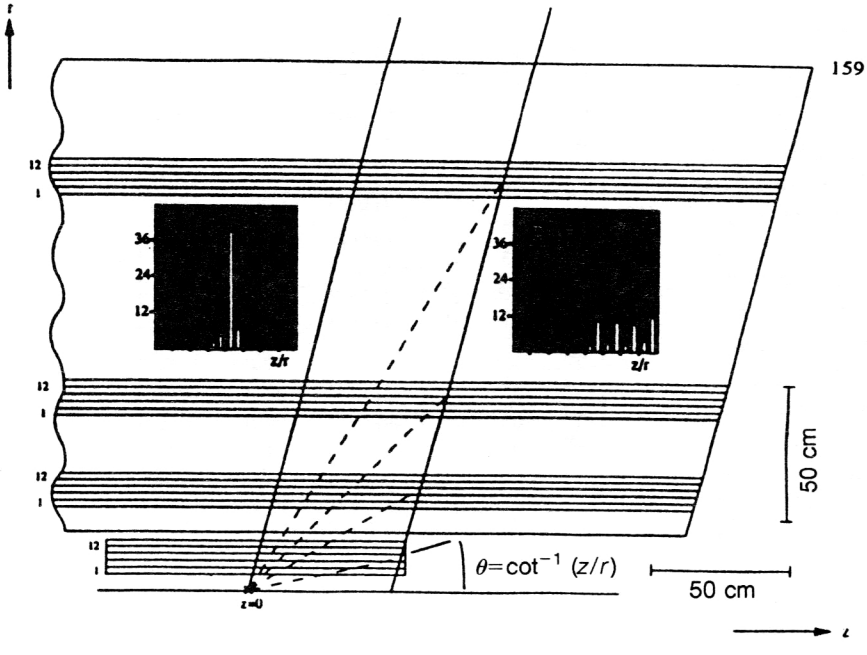

FIG. 29. Method of finding the coordinate of the primary interaction vertex in the OPAL experiment. In the window on the left one can see a peak in the histogram for a track emanating from the interaction vertex. The picture for the background tracks is shown in the window on the right. The numbers 1–12 identify detectors of the "jet" chamber.

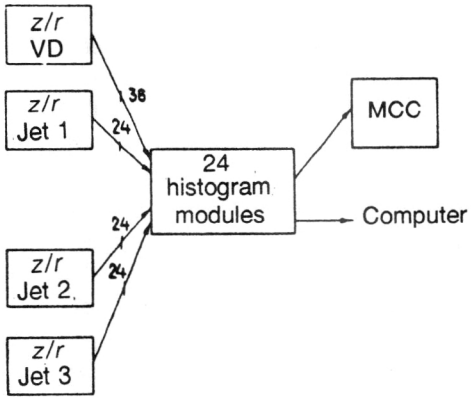

FIG. 30. Block diagram of a histogram processor. MCC is a majority coincidence circuit; jet 1–jet 3 are drift chambers; VD are the modules for the vertex detector.

the ratio z/r for all points belonging to the given track must be the same. It is assumed here that in the rz plane the tracks are straight lines. Thus, if for a given narrow angle φ a peak is obtained in the histogram, then the track emanates from the interaction point, as is shown in Fig. 29 (Ref. 75). To obtain a fast response, each of the sectors has its own processor. Figure 30 shows the block diagram of the histogram processor. First of all, the ratio z/r is calculated. After the maximum drift time, which is 7 μs for the "jet" chamber, has elapsed, the processor reads out the data from the φ sector. In total, 36 modules are needed to obtain the ratio z/r, which is calculated by using tabulated data. Then each of the 24 histogram modules, receiving the ratio z/r, constructs its histograms. After four histograms have been constructed, a test is made for the presence of a peak. The histogram processor is based on a memory module with random sampling. Of interest in this connection is the development in integrated circuitry of a plotter of 512×512 cell histograms with the possibility of cascading.[76]

9. THE METHOD OF SYNDROME CODING

The essence of the method of syndrome coding was first proposed in Ref. 77. Information read out from a detector plane containing n detection channels is first compressed to an amount $N \leqslant t \log_2 n$, and it is only after this that the data are to be processed (t is the multiplicity). The essence of the method is explained by means of Fig. 31 (Ref. 78). The multichannel charged-particle detector is regarded as a system that consists of n position-sensitive sensors. In the absence of an event, an n-bit unitary position code is read out. Event detection is represented as an error vector, which is added to the null vector. Then the syndrome N is rapidly calculated in accordance with the methods of the theory of error-correcting codes. The use of the method of syndrome coding gives a gain both in time and in hardware, provided the number t of position-sensitive sensors that fire simultaneously is relatively small compared with their total number n in the hodoscope plane. As a rule, this condition is satisfied in practice. The methods for constructing data-compression devices are described by means of the methods of algebraic coding theory. It is important here that the code of syndrome

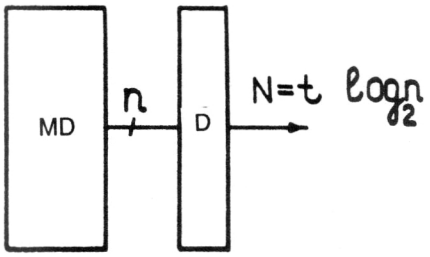

FIG. 31. The method of syndrome coding. MD is a multichannel detector; D is the decoding device.

N contains, within given limits, information both about the number t of position-sensitive sensors that have fired and about their coordinates. Since the number of bits of the syndrome code satisfies $N<n$, it is possible to use more effectively tabulation methods to determine the multiplicity and coordinates of events. We shall explain this by means of a concrete example. Suppose that a word read out from a detector plane has 15 bits. In the absence of an event, the 15-bit word

000000000000000

is read out from the detector. Suppose that signals from three particles have arisen at the 4th, 9th, and 13th positions (positions are counted from left to right). We then obtain the word

000100001000100.

An important step is the coding process, which is done by means of a coding array for a code that corrects three errors. Figure 32 shows two equivalent arrays. The first array (on the left) gives the elements of the Galois field GF(2^4).

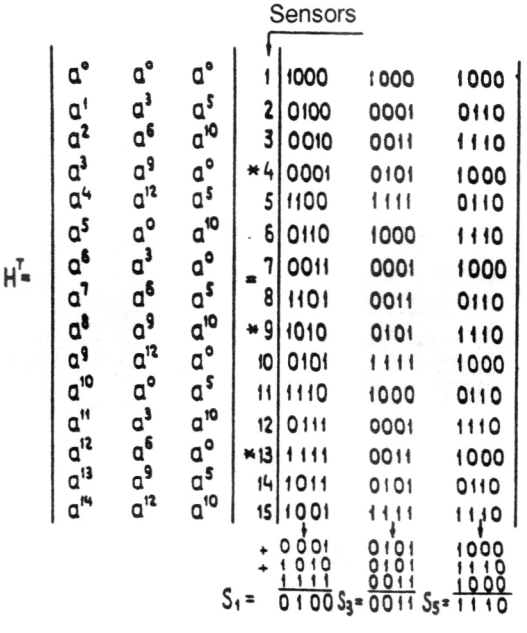

FIG. 32. Coding array for $n=15$ and $t=3$. The asterisk identifies sensors that have fired; + denotes the sum modulo 2.

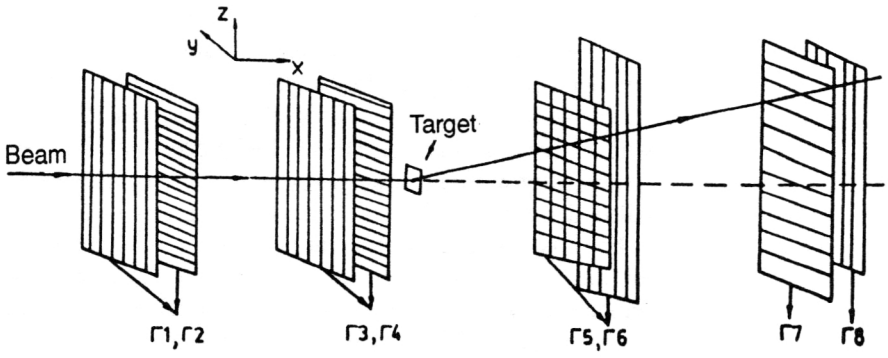

FIG. 33. Schematic representation of the hodoscope system of the experiment NA28 (CERN). Г1, Г2, Г3, and Г4 each contain 60 scintillation strips; Г5 contains 144 cells; Г1 and Г6 contain 13+2 strips, and Г7 and Г8 contain 24+2 strips.

The first column of the array contains all the nonvanishing elements of the field in the ascending order of their powers. The second column consists of the cubes of the corresponding elements of the field of the first column, while the third column contains the fifth powers of the field elements placed in the first column. On the right, Fig. 32 shows the array consisting of the binary equivalents of the Galois field GF(2^4). The last two columns in the array H^T can be omitted, since one of them is equal to zero, and two others are identical. The number of columns in the coding array in the general case can be $t \leqslant n/2$. By means of such an array, it is possible to construct the circuit of a parallel three-signal coder, since the positions of the units in the array determine couplings between the inputs of the parity-testing circuits, which perform the functions of adders modulo 2, and the outputs of the signal shapers. For given positions of the fired sensors, the coding process is illustrated by means of Fig. 32 (at the bottom), in which the symbol + denotes the sum modulo 2. It is important that the syndrome code contains data not only about the multiplicity $t \geqslant 3$ of the signals but also about their coordinates X_1, X_2, and X_3. As a result, after coding one obtains in place of a 15-bit code, on which it is difficult to perform logical and arithmetic operations, a 10-bit syndrome code consisting of elements of the Galois field. In its turn, the syndrome code can be processed in two ways. 1. Direct decoding by means of a charged coupled device makes it possible to determine the values of t and X_i rapidly. 2. If a word is long (more than 20 bits), it is possible to use tabulation methods to solve the coordinate equations, exploiting the fact that, as follows from algebraic coding theory,[79] relations hold between the values of S_1, S_3, and S_5:

$$X_1+X_2+X_3=S_1, \quad X_1^3+X_2^3+X_3^3=S_3, \text{ and}$$
$$X_3^5+X_3^5+X_3^5=S_5. \tag{11}$$

The tabulation methods of solution of the nonlinear equations (11) and the block diagrams of the fast coordinate processors are given in Refs. 78 and 80. A method for constructing majority coincidence circuits for a large number of inputs ($n>30$) and implemented by the method of syndrome coding is described in Ref. 81. A high speed and economy is achieved by virtue of the fact that the logical operations are performed on the syndrome code, and not on the n-bit word.

The choice of a particular coding scheme is determined by the formulation of the physical problem and, in the first place, by the value of the detected multiplicity t, on which the length N of the syndrome code depends. The use of the iterative Hamming–OR code in conjunction with fast programmable logic arrays made it possible to design an effective single-track processor for selection of particles on the basis of the scattering angle. The detectors that were used were scintillation hodoscopes. Figure 33 gives a schematic representation of the hodoscope system. To achieve reliable resolution of "ghosts" that arise in the process of detection of more than one particle, one of the hodoscopes has a cellular structure. The decision time does not exceed 80 ns.[82] The code distance of the Hamming–OR code is $d=3\times2=6$. If the simple relation $t=(d-1)/2$ known from coding theory is used, it is found that by means of such a coding scheme two-track events can be unambiguously detected. For the construction of a track processor with higher multiplicity by the method of syndrome coding, it is necessary to use more powerful correcting codes. In Ref. 83 there is a description of a track processor designed to select straight tracks of particles detected in a multiwire proportional chamber. The number of tracks identified by means of such a processor may reach eight. For the data compression, the rule for calculating the syndrome N in accordance with the Boys–Chaudhuri–Wokingham codes was used.

10. USE OF MULTILEVEL MULTIPROCESSOR SYSTEMS

The use of transputers

Transputers are powerful single-crystal microprocessors with RISC architecture that have, in addition to a parallel interface, four fast sequential-action interfaces, by means of which it is possible to construct both one-dimensional and multidimensional networks for processing track information containing several hundreds or thousands of transputers. Figure 34 gives three different topologies of computational systems designed on the basis of transputers. Modern transputers, which are described in the review of Ref. 84, have a 32-bit processor and four independent duplex interfaces. The use of a network consisting of transputers to process physical data is described in Ref. 85. The presence of a sequential interface in such a system does not affect the speed of the network as a whole because of the independent operation of each transputer. Therefore, transputer networks are very

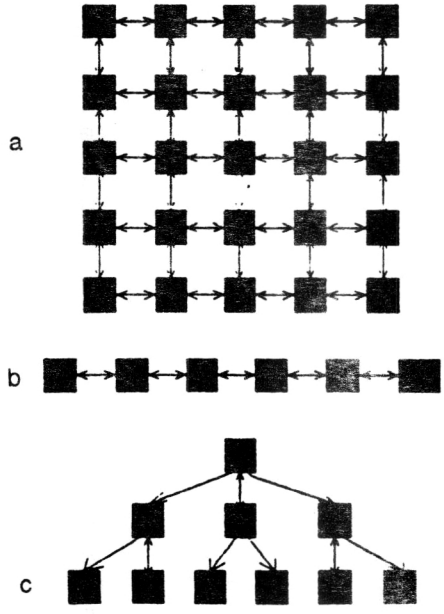

FIG. 34. Different configurations of computational systems based on transputers; a) array system; b) linear system; c) network of "tree" type.

promising for the processing of track information. Transputers and transputer networks are particularly widely used in the ZEUS detector (see below).

The use of powerful microcomputers

Track information is usually processed in two stages. After track segments have been determined or candidates for useful tracks have been identified, the data are stored on magnetic tape to permit further processing of them on powerful computers, which often contain emulators. However, for well-known reasons, the need for more powerful computing systems designed for the processing of track information increases all the time. The appearance of the 32-bit 68020 microprocessors of the firm Motorola in conjunction with the standard BME made it possible to design in 1986 the first version of a microcomputer.[86] Currently, an even more powerful system is being constructed for future experiments; its structure differs appreciably from that of the first version. The differences are as follows. 1. The process of calculation and exchange of data has a multilevel nature. 2. The 68020 microprocessor is replaced by a more powerful microprocessor of the new generation with cycle shorter than 10 ns and with a large number of resistors (RISC processors); they perform tens of millions of single-cycle operations. A more detailed description of such processors is given in Ref. 87. 3. For commutation of different branches, special commutator methods are used. In other words, such a system resembles in its organization a computational network with corresponding software. 4. Instead of magnetic-tape storage, it is possible to use optical disks or video tapes with capacity up to 2 Mbyte. 5. The system includes a set of random-access memories with direct access to the processors.

Figure 35 shows the structure of a multibranch multiprocessor system for processing track information. The individual links of the system consist of computing modules joined by means of a BME bus controlled by an MIV controller (BME multiprocessor interface). In its turn, each module is connected through the multiprocessor of the branch controller to a bus that is organized in such a way that each module can operate in the driving–driven regime or vice versa. In addition, magnetic-tape memories (M) are connected to one of the modules through a magnetic-tape interface for inputing and outputing data.[88]

Parallel event construction

As we have already noted, modern data-processing systems have a similarity with computer networks. The need for such organization is dictated by the fact that the information detected in large physics facilities arrives in parallel from different detectors. At the same time, the data processing is done by the pipeline method, and after processing the data are passed from one step to another. Moreover, information about events may be in different transmission channels, but ultimately they must be stored in prearranged memories or processors. Furthermore, all channels are simultaneously active, and the links between the detectors and processors change dynamically with each timing pulse.[89] To ensure a high rate of event switching, the timing frequency in future experiments must be of order 100 MHz. The main logical elements in event builders are commutators. The simplest of

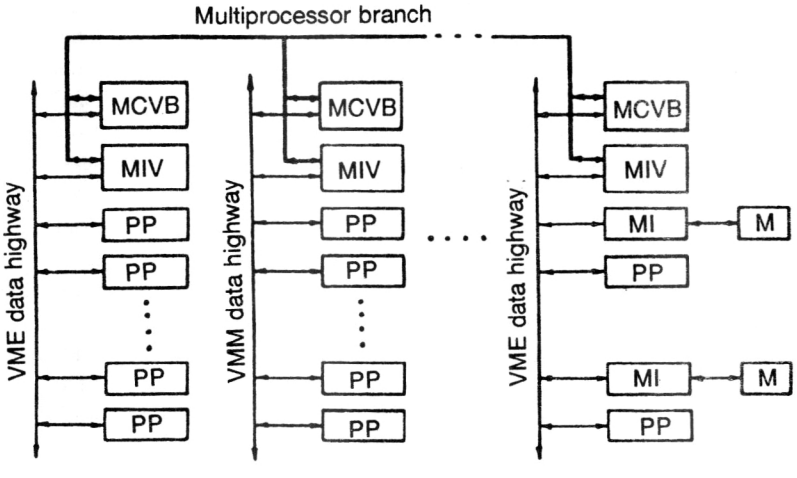

FIG. 35. Block diagram of a computational multiprocessor module. MCBB is the multiprocessor controller of the VME branch; MIV is the multiprocessor VME interface; IM is the magnetic-tape interface; and M is the magnetic- tape unit.

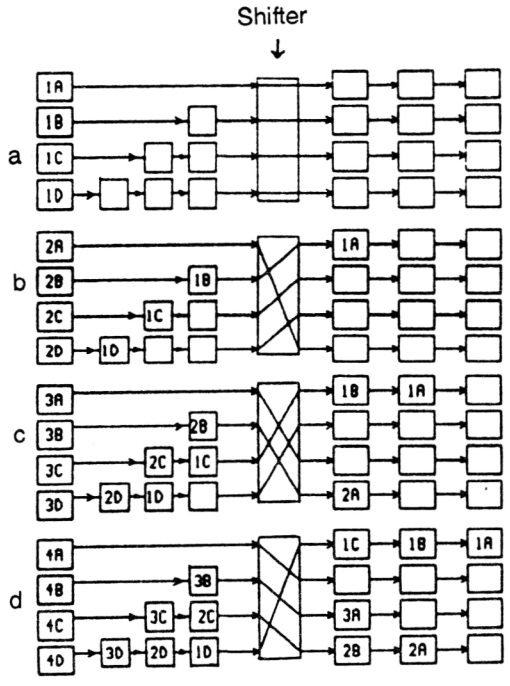

FIG. 36. Operation of a barrel shifter. 1A–1D are data relating to events of the same kind.

them, which has been widely used, is the barrel shifter. The operation of the barrel shifter is illustrated by means of Fig. 36, in which the letters A, B, C, and D denote event fragments. To ensure that these fragments arrive at the necessary processor, a definite number of digital delays is foreseen in each channel, the delays being implemented by memories of FIFO type. In Ref. 90 there is a discussion of the possibility of realizing event construction by fast commutation networks, by means of which it is possible to arrange multiplexing of data from the sources to the processors. There now exist several commercial devices of this kind with optical transmission channels and a throughput of 100 Gbyte/s. However, the cost of such networks is high, and, in addition, the data protocol does not take into account the specific features of experiments in high-energy physics. Finally, conflict situations are possible in networks. There are also proposals to use two-port memories as conflict-free parallel commutators. However, when a large number of commutation channels are used, arrays consisting of expensive memories are required. Therefore, parallel 2×2 commutators have been used as the commutators, and by means of them multichannel parallel commutators and barrel shifters have been constructed by cascading methods.[91] In Ref. 92 there is a description of a barrel shifter with optical transmission channels realized in a standard VME based on a 2×2 commutator in the form of an integrated circuit. Using such commutators, it is possible to construct parallel commutators with a large number of inputs, as shown in Fig. 37. In the ZEUS detector,[93] the event construction is done on the basis of a transputer network. Events are transmitted from the second-level to the third-level trigger system at the rate 24 Mbyte/s.

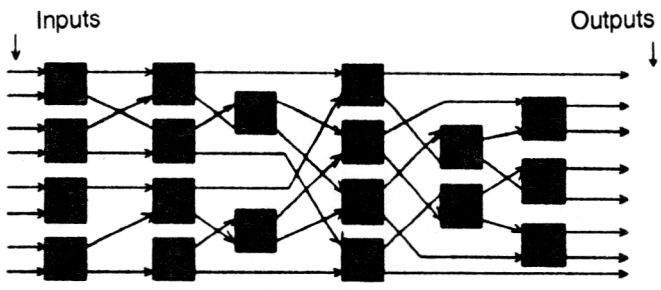

FIG. 37. Block diagram of a parallel commutator based on a 2×2 commutator.

11. PROCESSING OF TRACK INFORMATION AT THE LARGE HADRON COLLIDER

Detailed investigations have shown that the system of detection and signal processing in future experiments may consist of three trigger levels. We consider briefly their characteristics.[94] The generalized scheme of a trigger system is shown in Fig. 38.[95]

First level

In view of the short interval between collisions of bunches (15 ns), the logic devices and simplest processors must operate at frequency 67 MHz and with a decision time of not more than 15 ns. In practice, this means that if a detection channel contains several pipeline stages, the intermediate result of a decision must be transmitted to the next step of the pipeline at each timing pulse. It is therefore necessary to arrange for analog and, when necessary, digital delays in each detection channel. These questions are considered in detail in the review of Ref. 96. In the calorimetric channel, fast signal coding is also needed. The maximum decision time must not exceed 1 μs. This time is determined by the number of steps of the pipeline processors. In addition, the first trigger level may consist of a definite number of pretriggers, each of which solves its own specific problems on the basis of data obtained from detectors "assigned" to them. The frequency at which events reach the second level must be lowered by a factor 100–1000.

The muon trigger systems must be fast, since it is necessary to measure with high accuracy isolated tracks with large momentum (small curvature). In Ref. 97, it is proposed to use a pipeline track processor based on a scintillating optic-fiber hodoscope of cylindrical shape as a first-level trigger, by means of which it is proposed to look for tracks by the "window" method in the plane of r and ϕ. The hodoscope must be situated within the central detector. In addition, the fast response of the system can also ensure detection of bunch collision events. Signals received from the waveguides are detected by means of multianode photomultipliers. In the ATLAS project,[98] it is proposed to detect high-momentum muons by means of cheaper high-precision drift tubes. Algorithms and the structure of a computational medium based on systolic arrays in a homogeneous computational medium are developed in Ref. 99. They realize fast pipeline processing of track information detected by means of drift tubes.

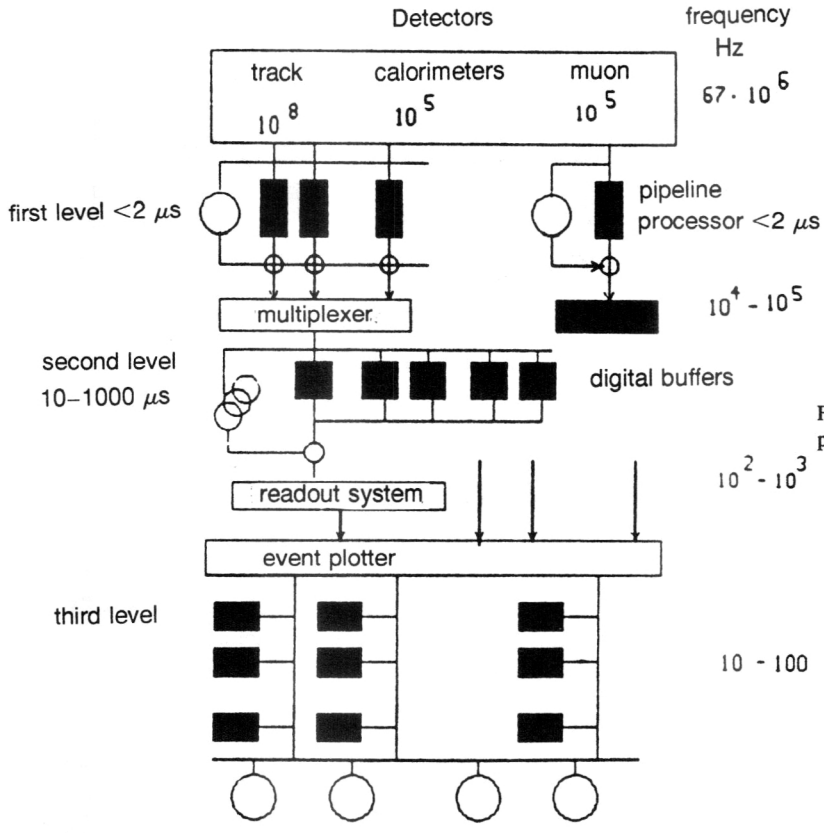

FIG. 38. Generalized block diagram of a trigger system proposed for experiments on the LHC.

Second level

Complete programmability and decision time 10–20 μs are the main requirements on the second-level trigger system. However, the programmability must refer mainly to the change of the parameters. The event frequency is reduced from 10^4–10^5 Hz to 10^2–10^3 Hz. At the same time, the track information is analyzed approximately by means of rough topological criteria and data obtained in the calorimetric trigger. To construct the second-level trigger system, it is proposed to use the best computing techniques and semiconductor technology. Wide use will be made of programmable logic arrays, digital delays in the form of memories of FIFO type in order to lengthen the allowed decision time, commercial array processors based on transputers, digital signal processors, and specialized processors developed for these purposes. It should be noted that the functions of the second-level trigger system have a dual nature: Specialized event selection and a global selection process. In the specialized selection, the data on candidates for detection such as electrons, photons, muons, or jets are parametrized by using data from one or several detectors. In the global calculations, a combination of obtained solutions is used in order to produce a signal indicating a useful event.

Third level

At the outputs of the third level of the trigger system, data arrive that are to be stored at a frequency of order 1–10 Hz in a large memory for final analysis. The data arrive at the inputs from the event-building outputs. In addition to general-purpose multiprocessor systems, it is also proposed to use specialized computing devices that contain 1000–5000 fast microprocessors on one standard board. To reduce the cost and increase the speed, these microprocessors have a specialized architecture oriented toward the processing of track information. In this connection, the project MPPC (Massively Parallel Collaboration), which was already proposed in 1988 (Ref. 100), is of interest. The project is based on an associative processor element, the structural arrangement of which is shown in Fig. 39. It contains two registers, one of which is a 64-bit data register and is connected to a data bus in order to store and read track and other information about events, while the other is an activity register having a capacity of 6 bits. This register is connected only to the memory bus. It is important that the contents of these registers can be compared with data that appear at definite times in the corresponding buses. The results of comparison or noncomparison are stored in two triggers TR1 and TR2. In addition, each associative processor element has its own local memory, which is needed to keep previously recorded data and also to implement a small number of simple commands. An array of similar associative processor elements, mounted on one standard board, belongs to the class of so-called SIMD computers (Single Instruction Many Data) as regards its principle of operation. This means that one and the same instruction is simultaneously implemented by all processors containing different data. For example, this could be information received from the second-level trigger and also data previously stored in local memories. Thus, in contrast to an ordinary computational process with sequential addressing, the associative processor elements are used to realize parallel accessing of data in accordance with their contents.

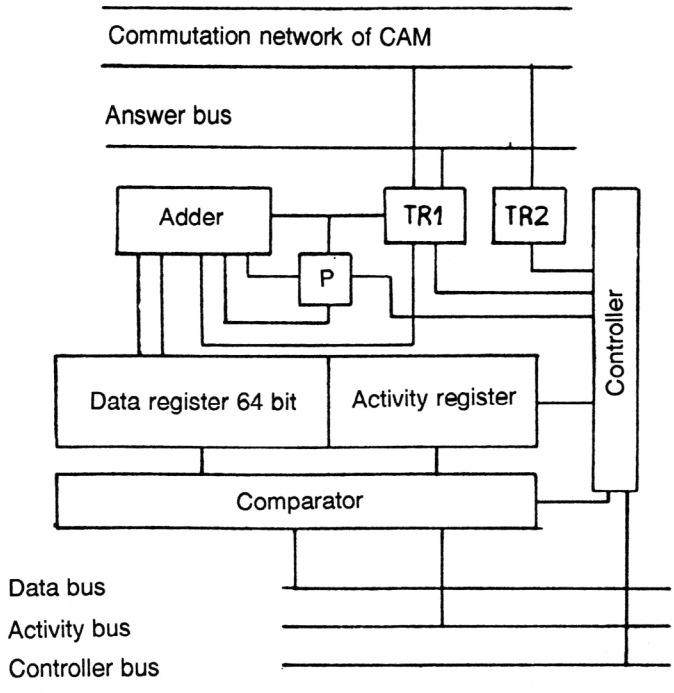

FIG. 39. Block diagram of an associative processor element. TR1 and TR2 are triggers.

CONCLUSIONS

The great outlay of material and intellectual resources in the development of the physics of high and ultrahigh energies necessitate a careful investigation of effective methods of processing track information in real time. Several experimentalists have noted that the successes in obtaining useful information detected in multichannel charged-particle detectors in colliders with luminosity 10^{33}–10^{34} will depend above all on the possibilities of the systems of detection, selection, and reconstruction of the topologies of complicated events under conditions of an appreciable background and a high multiplicity of the detected particles. This need can be met by the development and improvement of the following approaches used in experimental physics of high and ultrahigh energies.

1. The improvement and construction of new types of two- and three-coordinate high-precision detectors with high spatial resolution and a large number of sensitive cells in the region of detection. 2. Wide introduction of optical methods of detection and processing of physical information. 3. Development and introduction of specialized microcircuits. 4. The development of new on-line methods of joining tracks and the combination of on-line processing methods with hardware methods in real time. 5. The design and use of fast specialized processors based on programmable logic arrays, content-addressable memories, etc., that make it possible to process simultaneously track information with respect to three coordinates. 6. Research studies aimed at the introduction into the practice of physics experiments of various methods and hardware means used for pattern recognition: various types of transformation of variables, associative methods of data processing, cellular automata and neural (synergetic) networks, syndrome coding, etc. 7. The creation of powerful computational systems and networks for the final analysis of track information in both the on-line and off-line regimes.

If we compare the various hardware methods of reconstructing track information, then in the near future promising developments will include neuronlike devices that realize efficient mathematical algorithms and combine hardware means based on content-addressable memories and programmable logic arrays. In conjunction with on-line methods of data processing, such processors can solve difficult recognition problems of not only track information but also a relatively new phenomenon, namely, particle jets, which are difficult to identify by traditional methods.

In conclusion, it should be noted that various problems that in one way or another relate to the subject of this review have been described in more detail in reviews published earlier in this journal. In Ref. 101 there is a detailed exposition of the physical properties of charge-transfer devices and their characteristics. The uses of charged-coupled devices in systems of image readout, memories, etc., are described. In Ref. 102 there is an analysis of the characteristics of radiation detectors with measurement of the coordinates of the tracks of particles based on the centroid of the distributions that arise after passage of a particle through the sensitive volume of the detector. The application of cylindrical multiwire proportional chambers is considered in Ref. 103. The review of Ref. 104 considers the present status and prospects for the development of electronic methods in the physics of high and ultrahigh energies. The essence of the method of syndrome coding is described in more detail in Ref. 105. The review of Ref. 106 describes the present status of the method of detection and processing of electrical and light signals detected in calorimeters. Finally, the review of Ref. 107 considers the computer and on-line implementation of the systems of data analysis of electronic experiments in high-energy physics.

[1] Yu. V. Zanevskiĭ *et al.*, *Multiwire Proportional Chambers* [in Russian] (Atmoizdat, Moscow, 1975).
[2] F. Sauli, Preprint CERN 77-09, Geneva (1977).
[3] J. Hientze, Nucl. Instrum. Methods **156**, 228 (1978).
[4] H. Boerner, H. M. Fischer, H. Hartmann *et al.*, Nucl. Instrum. Methods **176**, 151 (1978).
[5] D. H. Saxon, Preprint Ral-87-022, Chilton (1987).
[6] D. Fancher, J. H. Hilke, S. Loken *et al.*, Nucl. Instrum. Methods **161**, 383 (1979).
[7] G. Rai, A. Arthur, F. Bieser *et al.*, Preprint LBL-28141, Berkeley (1990).
[8] A. A. Carter, J. R. Carter, and J. C. Hill, Nucl. Instrum. Methods **A286**, 107 (1990).
[9] D. Bernstein, J. Berstein, and K. Bunnell, Nucl. Instrum. Methods **226**, 30 (1984).
[10] R. Bouclier, G. Charpak, and G. A. Erskine, Preprint CERN-EP/87-89, Geneva (1987).
[11] W. H. Toki, Preprint SLAC-Pub-5232, Stanford (1990).
[12] P. F. Manfredi and F. Rogusa, Nucl. Instrum. Methods **A252**, 208 (1986).
[13] A. G. Chilingarov, Preprint 90-113, Institute of Nuclear Physics, Novosibirsk (1990) [in Russian].
[14] M. Bocciolini, A. Conti, and G. D. Caporiacco, Nucl. Instrum. Methods **A240**, 36 (1985).
[15] G. Hall, Nucl. Instrum. Methods **A273**, 559 (1988).
[16] B. Diericx, Nucl. Instrum. Methods **A275**, 542 (1989).
[17] C. Angelini, W. Beusch, and I. J. Bloodworth, Nucl. Instrum. Methods **A277**, 132 (1989).
[18] M. N. Atkinson, D. J. Crennel, C. M. Fisher *et al.*, Nucl. Instrum. Methods **A263**, 333 (1988).
[19] W. E. Clealand, D. E. Krans, and J. A. Thomson, Nucl. Instrum. Methods **216**, 405 (1983).

[20] C. W. Fabjan, CERN-PPE, 193-124.
[21] Z. Guzik, Z. Hajduk, R. Krasowski *et al.*, Nucl. Instrum. Methods **104**, 337 (1972).
[22] Z. Guzik and S. G. Basiladze, Nucl. Instrum. Methods **114**, 83 (1974).
[23] N. M. Nikityuk, Prib. Tekh. Éksp. No. 2, 7 (1983).
[24] A. P. Kashchuk and N. Madjar, CERN CAMAC News, No. 13 (1978), p. 13.
[25] C. Verkerk, in *Proceeding of the 1980 CERN School of Computing*, CERN 81-03, Geneva (1981); SLAC-PUB-4611 (1988).
[26] P. Billoir, R. Frühwirth, and M. Regler, Nucl. Instrum. Methods **A241**, 115 (1985).
[27] R. Frühwirth, 1989 CERN School of Computing, 1990, CERN 90-06, 217.
[28] H. Crote, Rep. Prog. Phys. **50**, 475 (1987).
[29] H. Crote and P. Zanella, Nucl. Instrum. Methods **176**, 29 (1980).
[30] L. Bugge, Nucl. Instrum. Methods **179**, 365 (1981).
[31] R. Frühwirth, Nucl. Instrum. Methods **A262**, 444 (1987).
[32] P. Billoir and S. Qian, Nucl. Instrum. Methods **A294**, 219 (1990).
[33] H. Eichinger, Nucl. Instrum. Methods **176**, 417 (1980).
[34] D. G. Gassel and H. Kowalski, Nucl. Instrum. Methods **185**, 235 (1981).
[35] J. J. Becker, J. S. Brown, D. Coffman *et al.*, Nucl. Instrum. Methods **A235**, 502 (1985).
[36] H. Crote, in *Proceedings of the 1980 CERN School of Computing*, CERN 81-03, Geneva (1981).
[37] H. Eichinger, Preprint CERN 81-06, Geneva (1981).
[38] R. Frühwirth, D. Liko, M. Mitaroff, and M. Regler, Preprint HEPHY-PUB-532/90, Vienna (1990).
[39] B. Knapp, Nucl. Instrum. Methods **A289**, 561 (1990).
[40] C. Maclean, G. McPherson, and P. Wilde, Preprint RAL-74-049, Chilton (1974).
[41] Y. Ermolin and C. Ljuslin, Nucl. Instrum. Methods **A289**, 592 (1990).
[42] F. Klefenz, W. Cohen, R. Zos *et al.*, in *Proc. of the Int. Conf. "Computing in High Energy Physics 92,"* CERN 92-07, Geneva (1992), p. 251.
[43] D. Underwood, Preprint ANL-HEP-CP-86-98, Illinois (1986).
[44] M. Dell'Orso and L. Ristori, Nucl. Instrum. Methods **A287**, 436 (1990).
[45] P. Battaiotto, M. Budinich, and M. Dell'Orso, Nucl. Instrum. Methods **A287**, 431 (1990).
[46] R. G. Ofengenden, Prib. Tekh. Éksp. No. 4, 84 (1966).
[47] R. G. Ofengenden and F. N. Berezin, Prib. Tekh. Éksp. No. 2, 5 (1967).
[48] T. K. Kohonen, *Content-Addressable Memories* (Springer-Verlag, Berlin, 1980).
[49] P. Battaiotto, M. Budinich, and M. Dell'Orso, Nucl. Instrum. Methods **A293**, 531 (1990).
[50] M. Dell'Orso, in *Proc. of the Int. Conf. on the Impact of Digital Microelectronics and Microprocessors in Particle Physics*, Trieste (1988).
[51] M. Dell'Orso, Nucl. Instrum. Methods **A278**, 436 (1989).
[52] S. R. Amendolia, F. Bedeschi, G. Bellettini *et al.*, Nucl. Instrum. Methods **A289**, 539 (1990).
[53] *Principles of Self-Organization* [Russian translations], edited by A. Ya. Lerner (Mir, Moscow, 1966).
[54] G. D. Fischbach, Sci. Am. **267**, No. 3, 24 (1992).
[55] G. E. Hinton, Sci. Am. **267**, No. 3, 105 (1992).
[56] I. V. Kisel', V. N. Neskromnyĭ, and G. A. Ososkov, Fiz. Élem. Chastits At. Yadra **24**, 1551 (1993) [Phys. Part. Nucl. **24**, 657 (1993)].
[57] É. V. Evreĭnov, *Homogeneous Computational Structures and Media* [in Russian] (Radio i Svyaz', Moscow, 1981).
[58] *Neural and Synergetic Computers, Proc. of the Int. Symposium*, edited by H. Haken (Springer-Verlag, Berlin, 1988).
[59] C. Peterson, Nucl. Instrum. Methods **A279**, 537 (1989).
[60] B. Denby, Fermilab, Preprint 90/94, Batavia (1990).
[61] L. D. Jackel, H. P. Graf, and R. E. Howard, Appl. Opt. **26**, 5077 (1987).
[62] B. Denby, M. Campbell, F. Bedeschi *et al.*, Fermilab, Preprint 90/20, Batavia (1990).
[63] B. Denby, Comput. Phys. Commun. **49**, No. 3, 429 (1988).
[64] C. T. Awes, Nucl. Instrum. Methods **A276**, 468 (1989).
[65] B. Denby, F. Lessner, and C. S. Lindsey, Fermilab, Preprint 90/68, Batavia (1990).
[66] C. S. Lindsay and B. Denby, Preprint Fermilab-Pub-90/n92, Batavia (1990).
[67] G. Stimpfl-Abele and L. Garrido, Comput. Phys. Commun. **64**, No. 1, 46 (1991).
[68] L. Lönnblad, C. Peterson, and T. Rögnvaldsonn, Phys. Rev. Lett. **65**, 1321 (1990).
[69] P. Bhat, L. Lönnblad, K. Meier *et al.*, Preprint DESY 90 144, Hamburg (1990).
[70] M. Gyulassy and M. Harlander, Comput. Phys. Commun. **66**, 31 (1991).
[71] G. Darbo and S. Vitale, Nucl. Instrum. Methods **190**, 81 (1981).
[72] G. Darbo and B. W. Heck, Nucl. Instrum. Methods **A257**, 567 (1987).
[73] G. Darbo and B. W. Heck, IEEE Trans. Nucl. Sci. NS-34, No. 1, 227 (1987).
[74] A. A. Carter, J. R. Carter, R. D. Heuer *et al.*, Nucl. Instrum. Methods **A250**, 503 (1986).
[75] M. Arignon, A. H. Ball, K. W. Bell *et al.*, Nucl. Instrum. Methods **A313**, 103 (1992).
[76] F. Slorach and J. R. Alsford, IEEE Trans. Nucl. Sci. NS-35, No. 1, 209 (1988).
[77] N. M. Nikityuk, R. S. Radzhabov, and M. D. Schafranov, Nucl. Instrum. Methods **155**, 485 (1977).
[78] N. M. Nikityuk, in *Proceedings of the 6th Int. Conf*, AAECC-6, Rome, 1988, Vol. 357, Springer-Verlag, p. 324.
[79] W. W. Peterson and E. J. Weldon, Jr., *Error-Correcting Codes* (MIT Press, 1961).
[80] N. M. Nikityuk, Preprint E10-89-362, JINR, Dubna (1989).
[81] N. M. Nikityuk and A. V. Selikov, Prib. Tekh. Éksp. No. 6, 55 (1987).
[82] L. Gustafsson and E. Hagberg, Nucl. Instrum. Methods **A265**, 521 (1988).
[83] V. A. Kalinnikov, V. R. Krastev, E. K. Chudakov, Prib. Tekh. Éksp. No. 3, 105 (1986).
[84] T. Woeniger, Preprint 90-024, DESY, Hamburg (1990).
[85] R. C. E. Devenish, D. M. Gingrich, P. M. Hallam-Backer *et al.*, Preprint OUNP-99-5, Oxford University, Oxford (1990).
[86] I. Gaines, H. Areti, R. Atac *et al.*, Comput. Phys. Commun. **45**, 331 (1987).
[87] R. Wilson, Comput. Des. 68 (1988).
[88] I. Gaines, H. Areti, R. Atac *et al.*, Fermilab Preprint 88/211, Batavia (1988).
[89] A. J. Lankford, E. Barsotti, and I. Gaines, Nucl. Instrum. Methods **A289**, 597 (1990).
[90] W. H. Graiman, S. C. Loken, and C. P. McPorland, in *Computing in High Energy Physics 92*, CERN 92-07, Geneva (1992), p. 184.
[91] M. Nomashi, O. Sasaki, H. Fujii, and T. K. Oshka, in *Computing in High Energy Physics 92*, CERN 92-07, Geneva (1992), p. 188.
[92] O. Sasaki, N. Nomachi, T. K. Oshka, and H. Fujii, KEK Preprint 93-9, Ibaraki-ken (1993).
[93] H. Kowalski, in *Computing in High Energy Physics 92*, CERN 92-07, Geneva (1992), p. 33.
[94] N. Elis, in *Computing in High Energy Physics 92*, CERN 92-07, Geneva (1992), p. 51.
[95] L. Mapelli, in *Computing in High Energy Physics 92*, CERN 92-07, Geneva (1992), p. 237.
[96] N. M. Nikityuk, Prib. Tekh. Éksp. No. 6, 8 (1993).
[97] M. Curatolo, B. Esposito, G. Franchi, and A. Teodoli, Preprint CERN-PPE/92-168, CERN, Geneva (1992).
[98] ATLAS, CERN/LHCC/92-4, LHCC/12 (1992).
[99] V. Kotov, I. Aleksandrov, R. Pose, and Yu. Yatsunenko, E10-93-191, JINR, Dubna (1993).
[100] S. Anvar, E. Auge, A. Basso *et al.*, CERN 93-07, Geneva (1993).
[101] L. M. Soroko, Fiz. Élem. Chastits At. Yadra **10**, 1038 (1979) [Sov. J. Part. Nucl. **10**, 412 (1979)].
[102] L. S. Barabash, Fiz. Élem. Chastits At. Yadra **22**, 716 (1991) [Sov. J. Part. Nucl. **22**, 345 (1991)].
[103] N. P. Kravchuk, Fiz. Élem. Chastits At. Yadra **25**, 1244 (1994) [Phys. Part. Nucl. **25**, 526 (1994)].
[104] N. M. Nikityuk, Fiz. Élem. Chastits At. Yadra **23**, 1470 (1992) [Sov. J. Part. Nucl. **23**, 641 (1992)].
[105] N. M. Nikityuk, Fiz. Élem. Chastits At. Yadra **24**, 180 (1993) [Phys. Part. Nucl. **24**, 77 (1993)].
[106] N. M. Nikityuk, Fiz. Élem. Chastits At. Yadra **25**, 1004 (1994) [Phys. Part. Nucl. **25**, 428 (1994)].
[107] V. G. Ivanov, Fiz. Élem. Chastits At. Yadra **25**, 1279 (1994) [Phys. Part. Nucl. **25**, 542 (1994)].

Translated by Julian B. Barbour