ISSN 0169-1651

KONINKLIJK NEDERLANDS
METEOROLOGISCH INSTITUUT

WETENSCHAPPELIJK RAPPORT

SCIENTIFIC REPORT

W. R. 85 - 5

Some data reduction problems in meteorology solved by means of graph theoretical methods

by

Dr. H.M. de Jong

De Bilt, 1985

Contents

1. Abstract

It is observed that some data reduction problems in meteorology may be solved by graph theoretical methods.

The solution of a problem in a graph theoretical context corresponds to the solution of the original data reduction problem.

Three such problems come under review: a plotting problem, a super-observation problem and a profile problem.

All these problems being optimization problems admit a solution but computational algorithms generating these solutions are known for the plotting and profile problem only. As to the super-observation problem an algorithm is proposed which is believed to generate solutions close to the best possible solution. Super-observations are studied using the clustering observations provided by AIDS/ASDAR during three proximate flights.

With respect to the profile problem some applications are described using rawinsonde and aircraft meteorological observations.

The method proposed in this paper leads to a 30% reduction in the number of significant levels compared to the WMO method. Similar results for the plotting and super-observation problem are obtained.

2. Introduction

In meteorological practice when handling observational data it is often required to reduce the volume of data available in order to cope with some technical limitations and other externally prescribed conditions.

Data reduction may be required when 1) the frequency of data sampling in automatic operational observation systems becomes so high that the available data volume tends to surpass a critical limit of manageability, 2) when the data must be distributed over communication channels of limited capacity and 3) the data representation should meet certain conditions of display quality.

Advanced automatic systems of observation like satellite observations (SATOBs, SATEMs), automatic aircraft meteorological observations (ASDAR, ACARS) are in the foreground now and their pre-processing puts a high demand on methods and techniques of data reduction.

Even conventional systems like rawinsonde observations need to undergo such a data reduction when transmitting the available data in suitable code messages. Numerical weather prediction is another area requiring the introduction of reduction techniques.

The volume of observational data may be controlled mainly through two options, i.e. omitting observations or replacing a group of observations by a representative so-called super-observation.

In both cases the process of data reduction will entail a certain loss of information content. It will be a general goal to restrict the loss of information content to at least a practical minimum.

It occurs in practice that identical reports of observations arrive via separate communication channels, for example an aircraft report received via a ground satellite station through the ASDAR system and received via the normal WMO/ICAO channels as an AIREP. The elimination of such (semi)identical reports from the information flow is considered a trivial case of data reduction and will not be discussed further.

The purpose of this publication is to pose some problems of optimization in meteorological data processing schemes and to search for their solution by graph theoretical means.

The optimization of the data reduction presupposes that some normative measure is assigned to the information content of the data and in addition a suitable objective function is defined which needs to be optimized.

This is a difficult matter. It has not been explored so far in meteorology. Therefore, to simplify the discussion we will concentrate upon some realistic problems in which a maximum or minimum is sought for an objective function. The problems we will examine here are:

- the pictorial representation of information on meteorological charts (plotting problem).
- the fusion of locally dense observations (super-observation problem).
- the search for significant levels in wind, temperature and humidity profiles (profile problem).

The optimization problems here are of such a nature that in order to solve them, it will be necessary to make an excursion into the mathematical discipline of graph theory.

We will use only general concepts and elementary principles from this theory. For an introduction to graph theory the reader is referred to textbooks like BERGE [1], BUSACKER-SAATY [3], CHEN [4], CHRISTOFIDES [5], HARARY [7].

As to the problems to be described we may formulate these in graph theoretical sense as follows: given a collection of points or vertices (denoted by the set X) and a collection of lines or arcs (denoted by the set A) joining all or some of these points, a graph G is then fully described by the doublet (X, A).

The vertex set X usually involves the points of an observational network. The set of lines A links the points which are connected or related to each other. The lines A may have a direction which is shown by an arrow. The graph G then becomes a <u>directed</u> graph and may be described by the set X and a correspondence Γ called a mapping of the set X in X: $G = (X, \Gamma)$.

Solving an optimization problem of data reduction then is equivalent to a search for special subgraphs or paths in the graph G meeting an optimization criterium corresponding to that of the original problem setting.

To find a solution of the original data reduction problem the main task will be to associate the problem with a graph theoretical problem and subsequently to reformulate the problem in graph theoretical terminology and to seek for a graph solution.

Although at first sight the problems presented here are of a purely academic interest they certainly have a practical meaning because all solutions can be generated by means of suitable algorithms which have proven to be easily executable on today's fast and powerful computers.

3. Plotting problem

In the pictorial representation of meteorological data use is made of symbols. As to a surface observation all elements are plotted around the station circle in accordance with a standard plotting model. Furthermore, some rules are given concerning the symbols to be used for the plotting of various elements figuring in a graphic representation of the data (cf. Manual on the Global Data Processing, WMO No. 485).

The plotting of data on meteorological charts is presently done automatically by computer-driven plotters. This has made a lot of painstaking handwork obsolete now and it is generally agreed that the display quality is excellent.

However, there are also a few shortcomings. One complaint is the relatively low density of plotting due to lack of space in the chart. This may cause a considerable number of useful data to be discarded or to be placed elsewhere in an empty space of the chart.

In the discussion to follow it is assumed that the meteorological chart projection (stereographic, Lambert conformal conic or Mercator projection) is presented in an orthogonal coordinate system and that the necessary co-ordinate transformation has been applied in order to express the station locations in Cartesian co-ordinates.

Each plotting model requires a plotting space to be defined. For the plotting model of synoptic observations the plotting space practically is a circle (Fig. 1).

It is a rectangle when a single scalar value needs to be plotted like minimum temperature, total amount of precipitation, etc.

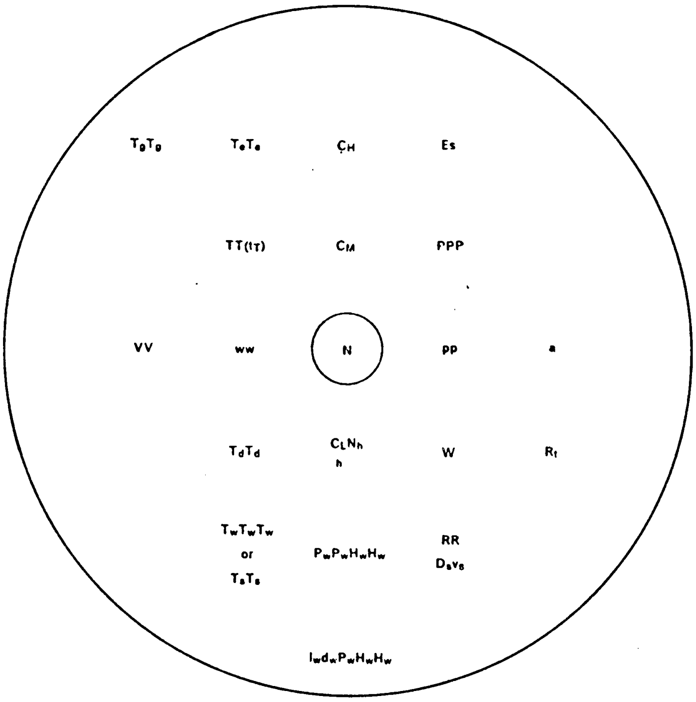

FIG 1. PLOTTING MODEL AND PLOTTING CIRCLE

For legibility of symbols plottings should not overlap, the stations should not be too close. When the dimensions, projection and scale of the meteorological charts are known this immediately confronts us with an interesting problem: <u>given a network of observation stations and a prescribed plotting space what can maximally be plotted on a meteorological chart?</u>

Obviously the plotting problem is a simple data reduction problem. It is also an optimization problem. Let us describe it in mathematical terms as follows: given a network of stations or set of points X and a set of not necessarily disjoint plotting spaces around these points (circles, rectangles), which subset $S \subset X$ yields a maximum number of stations with mutually disjoint plotting spaces?

Fig. 2a and 3a show specimens of such point sets in case of a circle and a rectangle as plotting space ($n = 40$).

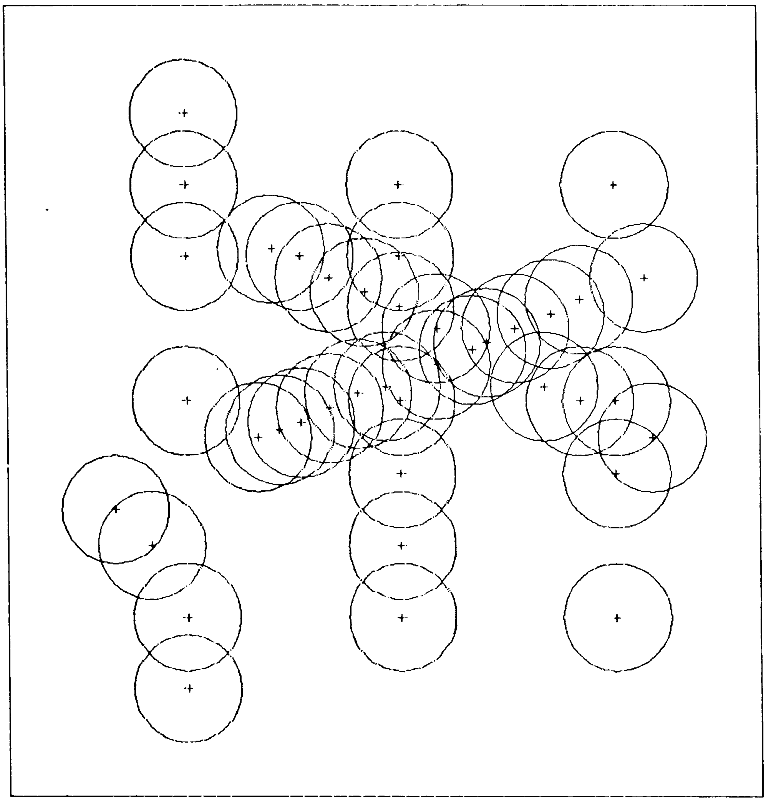

FIG.2A NETWORK OF POINTS WITH CIRCLE AS PLOTTING SPACE

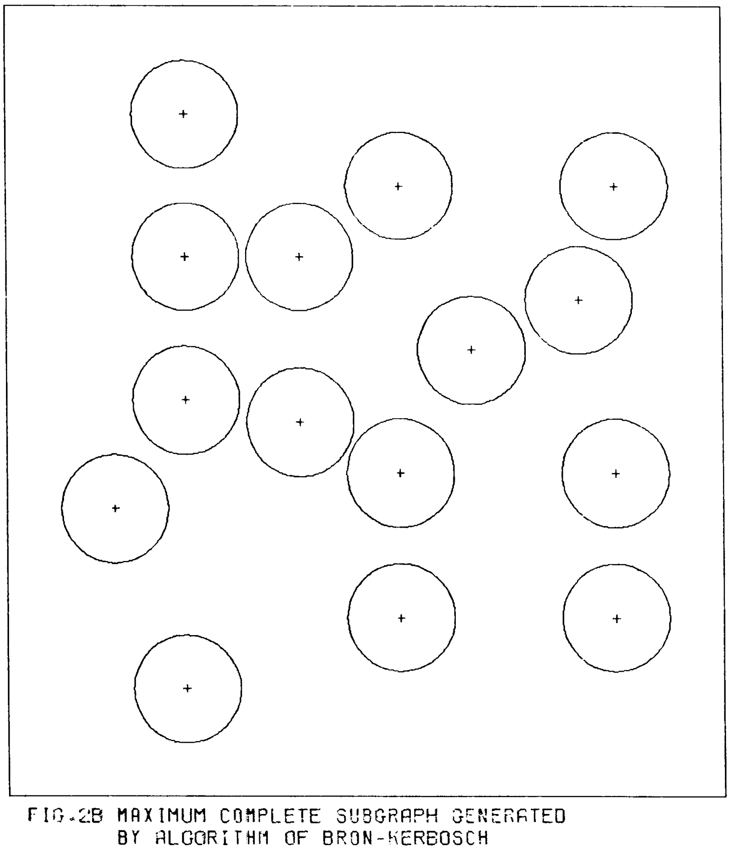

FIG.2B MAXIMUM COMPLETE SUBGRAPH GENERATED
BY ALGORITHM OF BRON-KERBOSCH

In case of a circular plotting space, the problem may also be visualized as a 'dime' problem: throw a handful of dimes on a table; some of them are disjoint, others partly cover each other. Take dimes away in such a way that all dimes left are disjoint and the total number of dimes left is a maximum or the number of dimes removed is a minimum.

Note: in case the dimes are replaced by rectangles and these rectangles are elongated to the length of little sticks of 1 or 2 decimeter pointing in an arbitrary direction then the game of the dimes shows much resemblance with the Japanese playgame known as Mikado.

3.1 The plotting problem viewed as a graph problem

The plotting problem is closely related to a graph problem. The graph G is composed of a set X of observing stations and a set of links A which express a connection of those stations which have disjoint plotting spaces. The graph G (X, A) is undirected.

In case the plotting space is a circle a link ($\in$A) exists when the distance between the stations i and j is greater than the diameter of the circle.

The graph G is a complete graph provided a link exists between all elements of X. The corresponding plotting spaces then are all mutually disjoint. Evidently the solution of the optimization problem here should be sought among the subgraphs which are both complete and maximal.

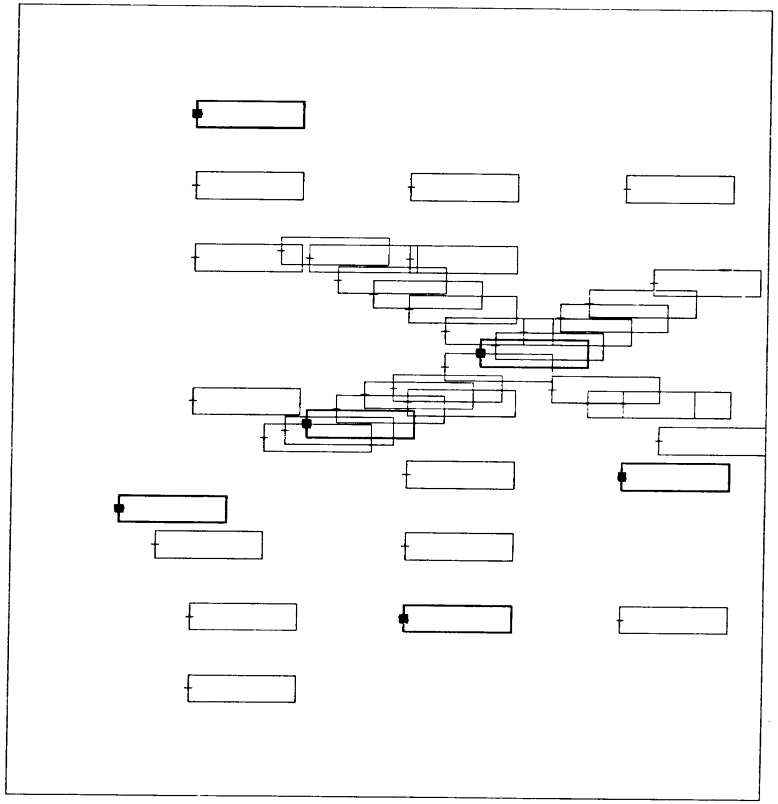

FIG-3A NETWORK OF POINTS WITH RECTANGLE AS PLOTTING SPACE

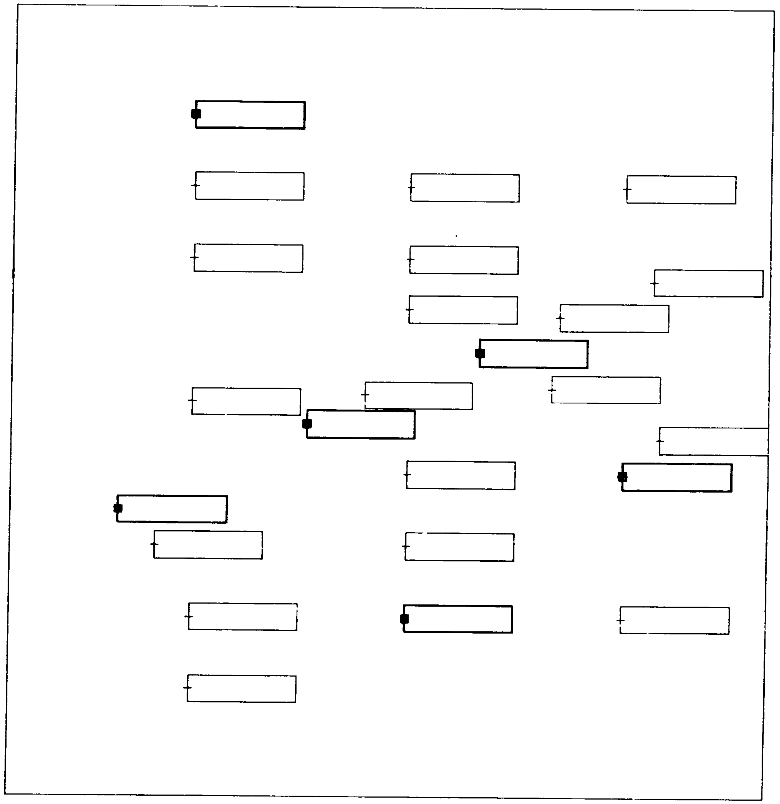

FIG-3B MAXIMUM COMPLETE SUBGRAPH GENERATED BY ALGORITHM OF BRON-KERBOSCH

A maximal complete subgraph of G is a subgraph based on the set S of vertices which is complete and which is maximal in the sense that any other subgraph of G based on a set $H \supset S$ of vertices is not complete. There is a whole family Q of maximal complete subgraphs or so-called cliques in a graph.

Of course we are most interested in the maximal complete subgraph with the highest number of elements or highest cardinality. The highest cardinality is called the clique number or density. The higher the clique number the more dense the chart is plotted.

The complete subgraph corresponding to the maximum cardinality is a maximum complete subgraph. The solution of our plotting problem is therefore to be found in the maximum complete subgraph of G with maximum cardinality or clique number.

Some concepts and definitions may be clarified by giving an example.

Fig. 4 shows a network of 4 stations and corresponding plotting circles.

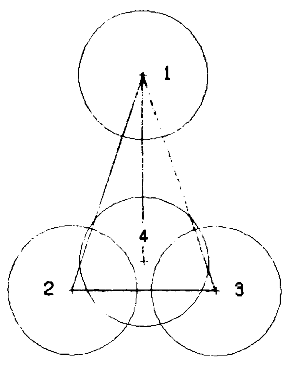

FIG.4 4-POINTS NETWERK

$X = \{1, 2, 3, 4\}$

$A = \overline{12}, \overline{23}, \overline{31}, \overline{14}$

The sets $\{1, 2\}$, $\{2, 3\}$ and $\{1, 3\}$ define complete subgraphs but these are not maximal.

$Q = \{X_1, X_2\}$, where $X_1 = \{1, 4\}$ and $X_2 = \{1, 2, 3\}$ defines the family of maximal complete subgraphs.

$X_2 = \{1, 2, 3\}$ is a maximum complete subgraph (in this case unique)

clique number $= \max |X_i| = 3$.

Is the solution of the plotting problem an unique solution, in other words is there only one configuration of stations whose cardinality equals the clique number or density? The answer is in the negative. In many optimization problems of graph theory the objective function takes on an integer value resulting not in one solution only but in a multiplicity of solutions.

There is in general no formula known for the multiplicity as a function of $|X|$, $|A|$ and other graph attributes.

How many cliques can a graph G have? The largest number of cliques (f(n) say) that a graph G with n vertices can have, as shown by MOON and MOSER [10], is:

$3^{n/3}$ if $n = 0 \pmod 3$

$4.3^{(n-4)/3}$ if $n = 1 \pmod 3$

$2.3^{(n-2)/3}$ if $n = 2 \pmod 3$

and the only graphs G which achieve f(n) are the following:

a) if $n = 0 \pmod 3$ then G consists of n/3 triples.

b) if $n = 1 \pmod 3$ there are two possible classes of graphs. Either G consists of one quadruple and (n-4)/3 triples or G consists of two pairs and (n-4)/3 triples.

c) if $n = 2 \pmod 3$ then G consists of one pair and (n-2)/3 triples.

It is to be understood that in these so-called Moon-Moser graphs triples, quadruples and pairs are groups of points per se.

Fig. 5 illustrates the Moon-Moser graph for n = 47. The graph consists of 15 triples and one pair and the set A of lines consists of 1035 links between points of disjoint groups. Hence the dark shade over the computer produced figure. The number of cliques is $f(n) = 2.3^{45/3} = 28697814$. It is remarkable that all cliques X_i here are not only maximal complete subgraphs but that they are all maximum complete subgraphs as well and $|X_i| = \max |X_i| = 16$.

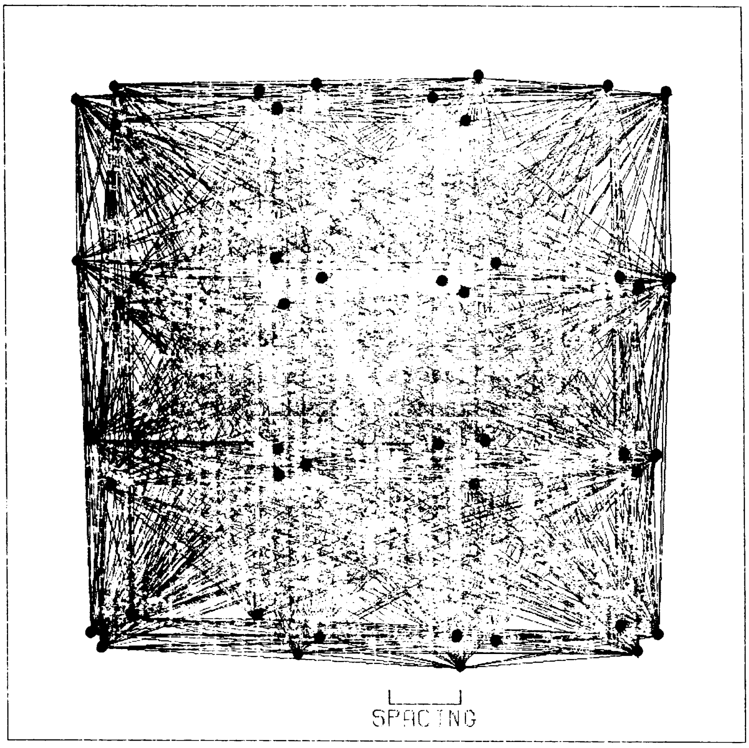

FIG.5 DEPICTING MOON-MOSER GRAPH.N=47.15 TRIPLES.1 PAIR

3.2 Computational method (Bron-Kerbosch algorithm)

The best method of computation of all cliques or maximal complete subgraphs of a graph G is probably that due to BRON and KERBOSCH [2].

The method is a systematic enumerative tree search method which computes all cliques in almost constant computation time per clique, independent of the size of the graph. The algorithm is close to the best possible (see Appendix A). The algorithm generates also all maxi*mum* complete subgraphs associated with the maximum cardinality or density. In Appendix A the ALGOL program listing is given. An excellent tool to test the program listing is the use of the Moon-Moser graphs (section 3.1).

A special property of the method is that it tends to produce the larger cliques first but it is unknown whether the algorithm really generates a clique of maximum cardinality directly after the start.

Since there is no expression known for the multiplicity and maximum cardinality (clique number, density) of the solutions, their magnitudes can be determined only empirically by counting them during the execution of the algorithm. The method, when exhaustively continued to the end, soon becomes computationally unwieldy in view of the potentially gigantic numbers of cliques present in networks of moderate and big size. The algorithm of Bron-Kerbosch is therefore terminated as soon as the first (and most probably a largest) clique has been delivered, keeping the computational time within practical bounds especially in large networks.

When the Bron-Kerbosch algorithm is tried out on a Moon-Moser graph, such as depicted in Fig. 5, then the (first) clique produced, as expected, contains a representative vertex of each quadruple, triple or point pair. The clique is a maxim<u>um</u> complete subgraph. Recall that the total number of cliques is enormous, here f(47) = 28697814.

Let us turn now to some less exotic examples. Fig. 2b and 3b depict the results of runs of the Bron-Kerbosch computational algorithm when applied to the figures shown in Fig. 2a and 3a after generation of the first clique. In Fig. 2a the original network consists of 40 points. An upper bound for the number of cliques is here according to Moon-Moser : $f(40) = 4.3^{36/3} = 2125764$. Experience shows that in arbitrary graphs this upper bound is far from being reached.

The number of elements of the resulting subgraph is 15. When compared with the original subgraph (n = 40), the reduction of data amounts to about 60 per cent. In Fig. 3b this percentage is about 40 per cent.

3.3 Method based on lexicographic ordering

When a network of stations is fixed e.g. the WMO global network of synoptic stations, one can determine a maximum complete subgraph once and for all and apply this universally in a standard plotting program.

However, when a network appears to be dynamic i.e. the network contains observations of moving platforms (aircraft, satellites, drifting buoys) or the network is subject to mutations, then an optimum network configuration changes all the time making its re-computation most desirable.

In current practice the plotting routine seeks for one complete subgraph by applying (alphabetic/lexicographic) ordering. Upon reception of a report, it is tested whether the plotting space is disjoint relative to all those reports plotted earlier. If disjoint the report is plotted, if not, it is dropped or plotted elsewhere in the chart where space permits. This process is continued until the last report is received before an agreed cut-off time. The process does not guarantee that the subgraph is a maximal complete subgraph or clique let alone that it is a maximum complete subgraph. Reports received early and far apart from each other have the best chance to be plotted.

3.4 Key stations

When for some reason certain stations in the network should always be plotted on the chart, the Bron-Kerbosch algorithm takes care of this by assigning first a virtual link between these key stations and all other stations in the graph G(X, A).

As a matter of fact, the algorithm starts by absorbing first those elements of X which have the highest number of links in A. As a consequence the key stations will all be present in the solution, at least when the plotting spaces of these stations themselves are mutually disjoint. Fig. 3b shows how the 6 "priority" stations, indicated by darker rectangles, indeed participate in the optimal solution.

4. Super-observation problem

The amounts of observations from some global observing systems may be so overwhelming that the flow of information must be regulated carefully by the operator and the data received by the user be subjected to a process of data reduction in order to keep the volume of data manageable.

Fig. 6a is a section of a chart valid for 27 April 1982, 12 GMT, depicting the positions of stations from the low density W. European radiosonde network and the positions of the high density, over land, AIDS/ASDAR data series, collected from three transoceanic flights. The aircraft departed from international aerodromes in Switzerland. The flights were approximately 20 minutes apart, they followed the same airway and the cruising altitude was 31000 feet. Fig. 6b shows the reported observations plotted in the 300 hPa chart.

Over France the observations cluster strongly and the legibility of plottings in the chart is poor. It is conceivable that here the use of data reduction would be profitable. One could for instance group together observations which are so close in space-time that they allow the substitution by one representative observation. Groups of close data values are then compressed into so-called super-observations. The total information content altogether will be somewhat diminished but the fusion of observations promotes the data manageability and their pictorial representation.

The introduction of super-observations may also be described as follows: each observation is considered to represent the state of the atmosphere in the direct vicinity of the observation point, within a given action radius. It is assumed that a group of observations whose "influence" circles partly cover each other will be replaced by a super-observation.

Note: a "super obs." of a rawinsonde observation is obtained when the observation made during the ascent on the balloon would be pooled with the observation made during descent on parachute.

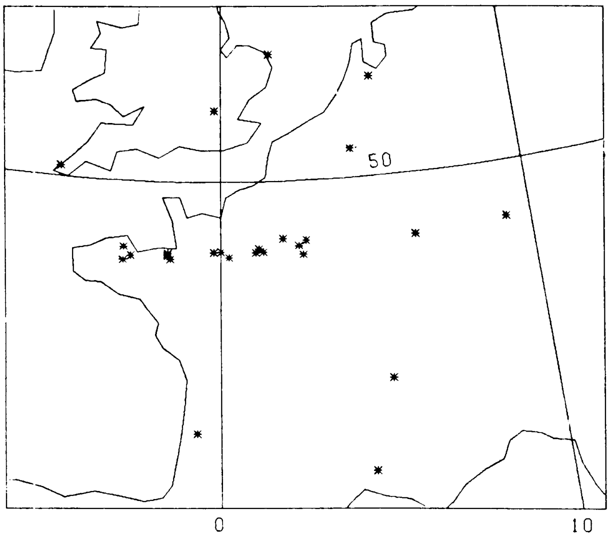

FIG.6A POSITIONS OF RAWIN STATIONS AND AIDS/ASDAR DATA SERIES ON 27 APRIL 1982 .12 Z

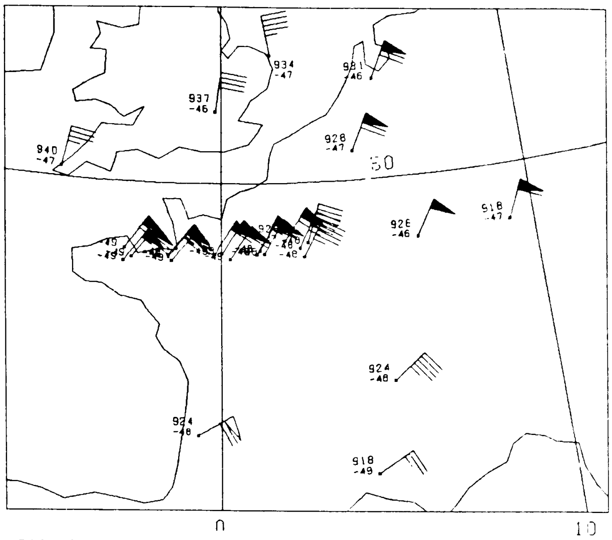

FIG.6B 300 H PA CHART OBSERVATIONS PLOTTED FOR DATA POINTS OF FIG.6A

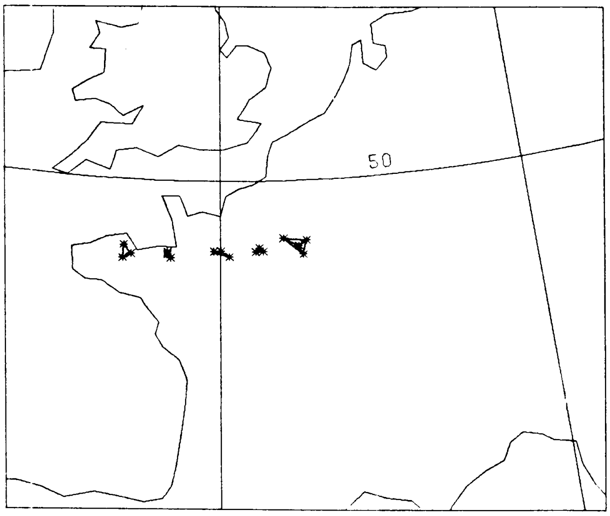

FIG.6C CLIQUES GENERATED BY THE ITERATIVE PROCESS USING THE ALGORITHM OF BRON-KERBOSCH

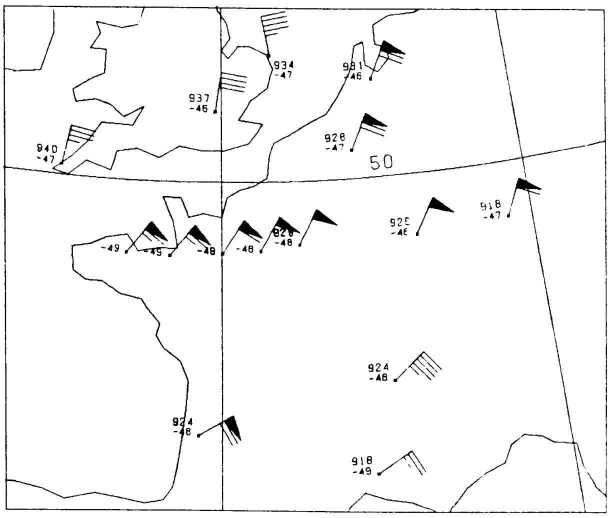

FIG.6D 300 H PA CHART .SUPER OBSERVATIONS INCLUDED

The problem of super-observations is then: given a measure of closeness or action-radius for "proximate" observations what is the maximal reduction of data when replacing proximate observations by super-observations.

The way in which the group of observations is replaced by a super-observation is not so relevant here. But it is quite apparent that one will determine (weighted) averages of all the elements and assign these to the representative super-observation in the gravity point of the group. Apparently the problem again is an optimization problem which hopefully may be solved when interpreted as a graph problem.

4.1 The super-observation problem viewed as a graph problem

The super-observation problem is related to the following graph problem: the graph is composed of the set X of observation stations and a set of links, A, expressing which points of X are adjacent in the sense that they are close within a given distance (action radius). The observations X_i which are mutually so close that they allow a super-observation to be substituted define a subgraph which needs to be both complete and maximal. Complete in the sense that all vertices in X_i are adjacent and maximal in the sense that any other subgraph containing X_i is not complete. It is obvious that the solution of the super-observation problem should be sought among these maximal complete subgraphs or cliques.

The graph problem corresponding to the super-observation problem is: find the family of cliques whose sets X_i have no vertices in common ($X_i \cap X_j = 0$, $i \neq j$) and for which the total number of vertices is maximal, $\sum_i |X_i|$ maximal.

The intersection condition $X_i \cap X_j = 0$, $i \neq j$, expresses that the cliques providing for the super-observations should not have vertices in common. This in general is not the case when viewing all cliques of a graph G(X, A), complicating the solution of the problem considerably. The problem is easily solved and quite trivially when all cliques are "disjoint" but the solution is unknown in the general case when the intersection condition is not fulfilled.

A nice example of disjoint cliques is encountered in complementary Moon-Moser graphs. Given a graph G(X, A) the complementary graph G(X, $\overline{A}$) is composed of the set X and the set $\overline{A}$ of links which are not in A. Fig. 7 represents such a complementary graph for n = 47. It is the graph complementary to the Moon-Moser graph of Fig. 5. In general the cliques in such a complementary graph comprise the complete graphs whose vertices are the quadruples, triples and pairs in the original Moon-Moser graph. They are all disjoint.

The solution of the super-observation problem in this case is trivially the set of all these cliques and $\sum_i |X_i| = |X|$ equals the number n of vertices in the original graph. In Fig. 7 the number of cliques and therefore the number of super-observations is 16.

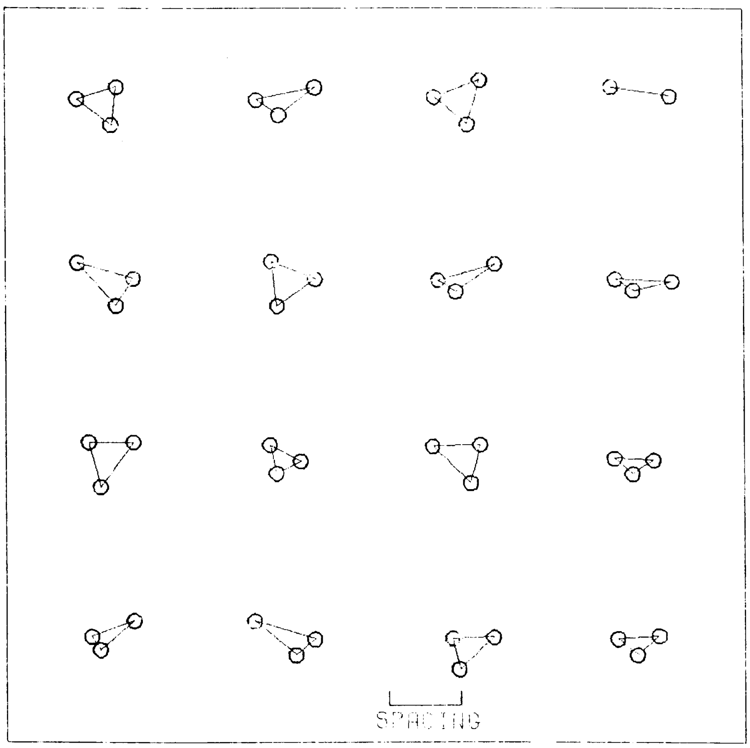

FIG.7 DEPICTING COMPLEMENTARY MOON-MOSER GRAPH OF FIG.5.

4.2 Iterative method for solving the problem

A practical approach which is believed to come close to the solution of the super-observation problem is the following.

Activate the algorithm of Bron-Kerbosch in the original graph G(X, A). As soon as the first clique is found, eliminate its set X_1 from X and re-activate the algorithm. Determine again the (first) clique and its set X_2 in $X - X_1$ and so on until after N steps X_N is obtained from the set $X - X_1 - X_2 - \dots X_{N-1}$ and no more disjoint cliques are found in the next step. The number N of disjoint cliques found and $\sum_{i=1}^{N} |X_i|$, the total number of vertices specify the total reduction of data. N is the number of super-observations to be substituted.

As the Bron-Kerbosch algorithmic computation generates the larger cliques first at each step (cf. section 3.2) there is a good chance that the sum total $\sum_{i=1}^{N} |X_i|$ is maximal or close to a maximum.

When this iterative process is tried out on a complementary Moon-Moser graph it is easy to see that the process generates indeed the known (trivial) solution: cliques related to the quadruples, triples and pairs. The sum total of number of vertices equals the number of vertices of the original graph. See Fig. 7 with 15 triples and one pair defining 16 disjoint cliques. Notice the small set $\overline{A}$ of lines here. $\overline{A}$ counts only 46 elements as against 1035 in the Moon-Moser graph of Fig. 5.

When applied to the network, displayed in Fig. 2a, where the circles are interpreted now as influence circles with action radius r, the iterative process generates 11 cliques in succession, all being appropriate, to replace their observations by super-observations (see Fig. 8).

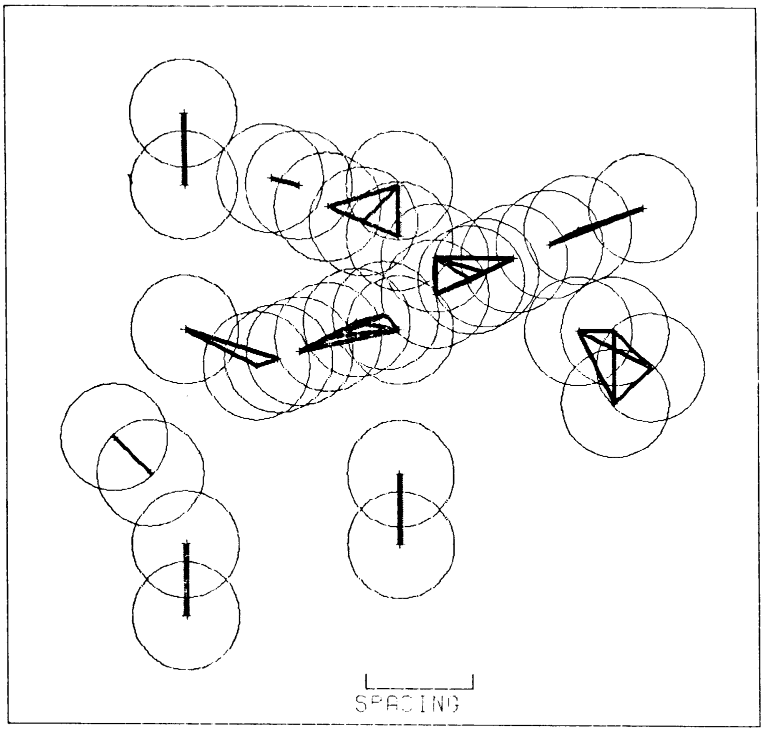

FIG.8 CLIQUES GENERATED BY THE ITERATION PROCESS USING THE ALGORITHM OF BRON-KERBOSCH IN THE NETWORK OF FIG.2A

The iterative method was also applied to the observational data presented in Fig. 6a and b. The cliques are generated successively in 5 steps involving all triples except for the first one which is a quadruple (see Fig. 6c). The projection of the charts in Fig. 6 is stereographic with standard latitude at 60°North and the scale is $1:12.5 \times 10^6$. The action radius is 75 km. Fig. 6d portrays the plottings of data in the 300 hPa chart including the super-observations which are attached to the gravity points of the 5 cliques and which are obtained by averaging the original observations of the cliques. The legibility of the plotted data has much improved after fusion of the clique observations into super-observations.

Notes: 1) The legibility of plotted observations depends of course on the action radius chosen, scale and projection of the chart, plotting model etc. When the action radius is adjusted to the radius of the plotting circle there is a correspondence between the plotting problem and the super-observation problem as can been deduced from their graph theoretical equivalences. But in general they need to be studied separately.

2) It is noted that in principle one can formulate a wide range of optimization problems with super-observations involved. All these problems have some meaning in a given context. For example, to find as few disjoint cliques as possible and fuse as many observations as possible, or to find as many disjoint cliques as possible and fuse as many observations as possible, etc. For all these problems solutions are unknown. This is still terra incognita in graph theory.

5. Profile problem

In meteorological practice various measurements take place, in situ, along trajectories of moving platforms (balloons, aircraft, parachutes) or remotely along scanning paths (satellites, radar). Vertical profile measurements of humidity, temperature and wind are standard. More and more horizontal, trajectory and scanning path profile measurements come into use (ASDAR, satellites). The sampling period is sometimes in the order of seconds. As a consequence the data volume puts a high demand on data handling, processing and reduction. Coding and distribution pose their own requirements.

The data reduction is partly accomplished by searching for significant levels in the data profiles which results in a minimum of reports to be compiled and encoded and a maximum of information to be distributed.

In the discussion to follow attention is paid to the techniques of data reduction applicable to trajectory profiles in general.

Using fast electronic computers effective methods of data reduction can be developed. These have experimentally been tried out on profiles provided by rawinsondes and ASDAR.

5.1 Polygonal approximation of profiles

The profile curves, displayed in diagrams, depict a physical parameter versus a monotonically increasing or decreasing function such as time, height, pressure altitude, geopotential or track distance along the trajectory. The parameters are single-valued functions over the whole range of measurements. In current practice the profiles are being approximated by polygons connecting the vertices of significant levels.

The significant levels alone should make it possible to re-construct the profiles with sufficient accuracy for practical use.

The polygonal approximation should meet certain aperture conditions. For example, the absolute or average profile departure from the polygon should stay within given bounds. The departures may be scalar or vectorial depending on the type of observed parameter.

In rawinsonde observations the following bounds are standard:

humidity : the departure of the humidity in "humidity polygons" obtained by linear interpolation between adjacent vertices shall not be more than 15 per cent relative humidity from the observed values.

temperature: up to 300 hPa or the first tropopause whichever is reached first, the departure should not be more than 1°C, not more than 2°C above that level.

wind : the departure with respect to the wind-direction profile should not be more than 10° and the departure with respect to the speed profile should not be more than 5 ms^{-1}.

The central problem of data reduction is here to approximate a profile curve by a polygon connecting a minimum number of vertices meeting prescribed aperture conditions.

A computational procedure to construct a polygonal approximation for wind profiles which is recommended for international use is to be found in the Manual on Codes, WMO No. 306. Details and weaknesses of the method are described in Appendix B. The WMO procedure is specially designed to facilitate the search for significant levels manually with the help of simple graphical representations. It is evident, however, that the proposed method is not optimal. Using the WMO-procedure the set of significant levels need not be minimal.

To develop a method which produces a minimum set of significant levels we again may formulate a specific optimization problem and try to solve this by finding a graph theoretical counterpart for it.

5.2 The profile problem viewed as a graph problem

The points X of a profile can be labeled by a ranking number. This implies that the profile problem is closely related to the following directed graph problem.

The graph $G(X, \Gamma)$ is composed of the set X and a correspondence Γ mapping every point x_i in points x_j $(j > i)$ for which along $x_i x_j$ one or more aperture conditions hold. The graph is also acyclic since the existence of a cycle would imply that the parameter under consideration would be multi-valued.

We extend the directed and acyclic graph by imposing to every link $x_i x_j$ an arclength equal to unity if the aperture conditions along $x_i x_j$ are satisfied. Then the optimization problem will be solved by finding in the graph $G(X, \Gamma)$ the shortest path from a starting vertex to an ending vertex. The shortest path corresponds to the desired polygonal solution whereas its total minimum length, because of the unity "length" along every arc, takes on an integer value, precisely equal to one less than the (minimum) number of vertices in the polygon solution.

In a somewhat cumbersome manner, the profile problem therefore proves to be associated with a shortest path problem in graph theory. In a directed acyclic graph there exists always a shortest path solution but in view of the integer value of the length of the shortest path the solution, as in so many graph problems, is not unique; there are multiple solutions.

In the original problem setting this simply means that there is a multitude of polygons solving the profile problem and having the same (minimum) number of vertices or significant levels.

5.3 Algorithm for the generation of the shortest path(s)

DIJKSTRA [6] developed an algorithm which solves the shortest path problem in general arc weighted graphs, imposed with positive arc lengths, very efficiently.

Here we will indicate the main features of this algorithm when applied to directed and acyclic graphs, arc weighted with positive arc lengths ($\equiv 1$). On the one hand the algorithmic steps become simpler in such a specific graph but on the other they just become more complicated, in view of the multiplicity of solutions. A detailed program listing used by the author is given in Appendix C.

The process of Dijkstra is iterative and based on vertex labelling where at the end of the k-th iteration the labels represent the lengths of the shortest paths from the starting point, containing k+1 or fewer arcs. In order to label vertex x_j with the length $\ell(x_j)$ of the shortest path from the starting point to x_j the following operation is performed:

$$\ell(x_j) = \min_{x_i \in \Gamma(x_j^{-1})} \{\ell(x_i)+1\}$$

At the same time the vertex x_j is provided with a label $m(x_j)$ representing the multiplicity of shortest paths between the starting point and x_j. During the operation the label $m(x_j)$ is obtained according to the rules:

if the path length $\ell(x_i)+1$ is increased: $m(x_j) := m(x_j)$

if the path length is decreased : $m(x_j) := m(x_i)$

if the path length remains unchanged : $m(x_j) := m(x_j)+m(x_i)$

Once the final values of the vertex labels $\ell(x_j)$ and $m(x_j)$ are known one can find the paths themselves as follows:

If the shortest path is unique this path can be obtained by a back tracking method based on the label $\ell(x_j)$ and a recursive application of the condition

$$\ell(x_i) = \ell(x_j) - 1$$

If the solution is not unique the information of the totality of shortest paths can be summarized using both labels $\ell(x_j)$ and $m(x_j)$, (for details cf. Appendix C)

This can concurrently be done by inserting all path information in a tableau. The columns of this tableau provide for the vertex sequences of the (multiple) path solutions. The number of columns equals the multiplicity of solutions. Ultimately, the whole family of path solutions is present in this tableau.

Such a tableau is a simple tool to use when deciding to select a particular member out of the family of solutions such as the most regular spaced solution (cf. Appendix C).

5.4 Profile curves produced by rawinsonde observations

5.4.1 Wind profiles

The polygon approximation of wind profiles by means of the computational procedure as proposed in the WMO Manual on Codes, WMO No. 306 is not optimal (cf. Appendix B). This can be remedied now by utilizing the computational algorithm based on the shortest path method using a computer (appendix C).

Fig. 9a is a plot of minute-points of the wind speed versus time derived from radar after release of the sonde on 30 November 1984, 12 Z and a plot in the same diagram of the minute-points of wind direction. The speed and direction profile curves serve as input material for the algorithmic computation, outlined in Appendix C, using precisely the WMO limits (wind direction 10°, wind speed 5 ms^{-1}). The algorithm was not activated for the wind speed and wind

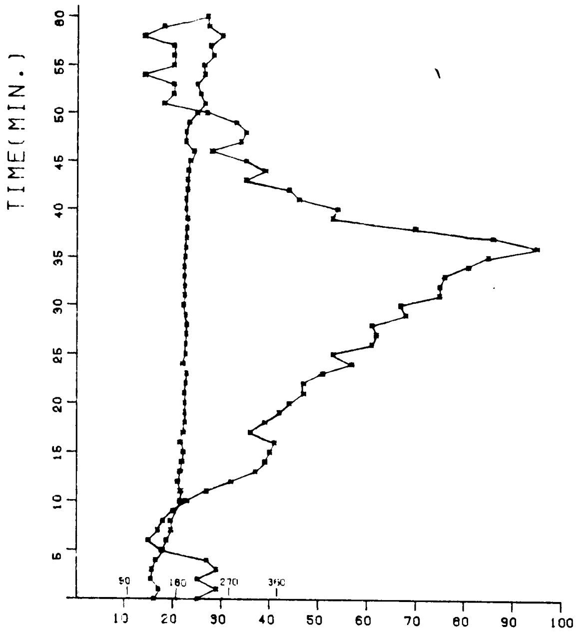

FIG.9A WIND SPEED AND WIND DIRECTION PROFILES
FOR DE BILT .30 NOVEMBER 1984.00 Z.

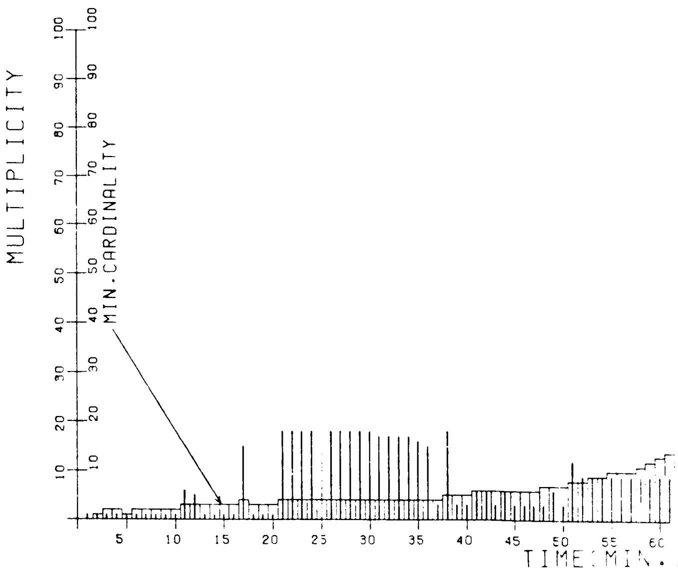

FIG.9B (MINIMUM) CARDINALITY AND MULTIPLICITY FOR
SOLUTIONS BETWEEN THE SURFACE AND TIME
ELAPSED AFTER RELEASE

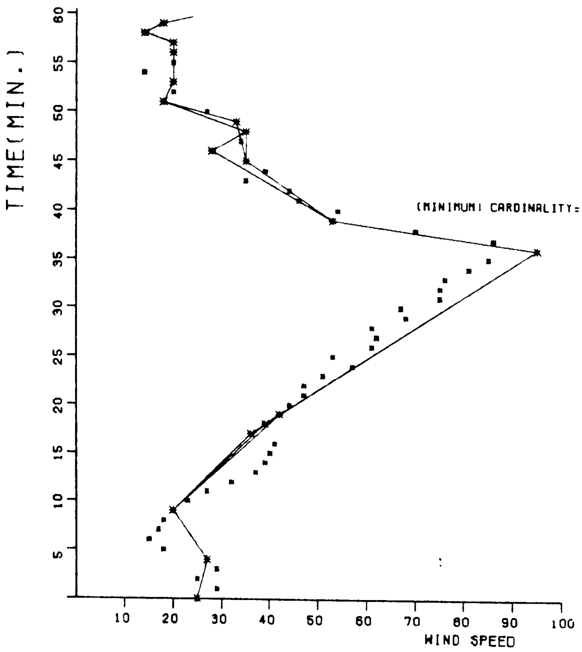

FIG.9C TOTALITY OF POLYGON SOLUTIONS
USING SHORTEST PATH ALGORITHM

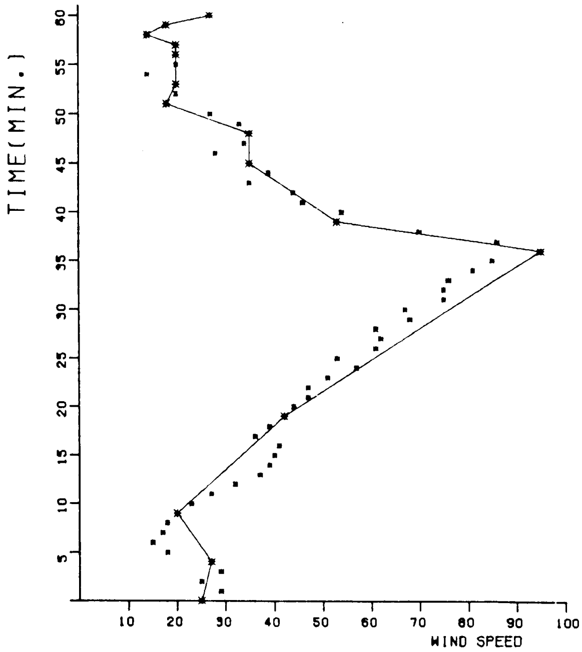

FIG.9D MOST REGULARLY SPACED POLYGON SOLUTION direction profiles separately but instead was run meeting both limits simultaneously.

To that aim the graph $G(X,\Gamma)$ was defined consisting of the set X of profile point(pairs) and a correspondence Γ which to each point(pair) x_i links those points $x_j(j>i)$ for which along an arc $(x_i x_j)$ both the 5 ms^{-1} and 10 degree aperture limits are satisfied. The results are shown in Fig. 9c. Between the surface and the top (61 minutes) there are 9 optimal polygon solutions each having 15 significant levels. Not less than 12 points are pivot-points, being common points in all solutions. The multiplicity = 9 and cardinality = 15.

The result can be read from the tableau below, summarizing the information of the totality of 9 solutions. The columns give the ranking numbers of the vertices of the solutions.

1	1	1	1	1	1	1	1	1
5	5	5	5	5	5	5	5	5
10	10	10	10	10	10	10	10	10
18	19	20	18	19	20	18	19	20
37	37	37	37	37	37	37	37	37
40	40	40	40	40	40	40	40	40
46	46	46	47	47	47	46	46	46
49	49	49	49	49	49	50	50	50
52	52	52	52	52	52	52	52	52
54	54	54	54	54	54	54	54	54
57	57	57	57	57	57	57	57	57
58	58	58	58	58	58	58	58	58
59	59	59	59	59	59	59	59	59
60	60	60	60	60	60	60	60	60
61	61	61	61	61	61	61	61	61

Which of the 9 solutions to report, encode and distribute is a matter of choice. In order to select a member of the family of 9 solutions one can subject these solutions to an additional constraint. For example, find the solution with most regular spacing of the vertices. As the vertex x_i is specified here by a ranking number or co-ordinate x_i along an axis the most regular spacing is found when analytically:

$$\sum_{i=1}^{\text{cardinality}-1} (x_{i+1} - x_i)^2 \text{ is a minimum.}$$

This special profile solution is represented in Fig. 9d. Fig. 9b depicts a graph showing the (minimum) cardinality or minimum number of elements of the optimal vertex sets of the polygon solutions between the surface and the time elapsed after release. The graph also depicts the multiplicity of these solutions. It can be observed that the cardinality is a non regular step function. The multiplicity is sensitive to the end-point chosen and is very erratic.

When not resolved in its components the profile for the wind may be conceived as a vector profile and the shortest path algorithm be generated based on a vectorial departure. Such an approach is feasable only by computer. We propose to take an upper bound for the magnitude of the difference vector between the linear interpolated wind vector and the actual profile wind. As upper bound for the vectorial departure we chose 7.5 knots.

Then subjecting the data of Fig. 9a to the shortest path algorithm results in the generation of 36 solutions through 8 vertices between the surface and the top. Fig. 10a depicts all these solutions in a normal wind speed curve. There are 4 pivot points, the maximum wind is one of them. Within the family of 36 path solutions we selected the one with most regular spacing, see Fig. 10b.

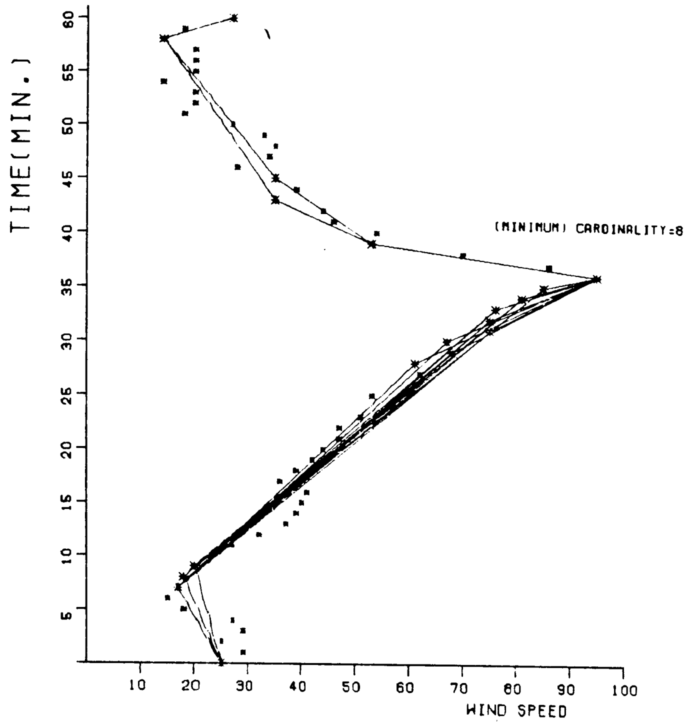

FIG.10A TOTALITY OF POLYGON SOLUTIONS GENERATED BY THE SHORTEST PATH ALGORITHM BASED ON A VECTORIAL DEPARTURE OF 7.5 KNOTS.

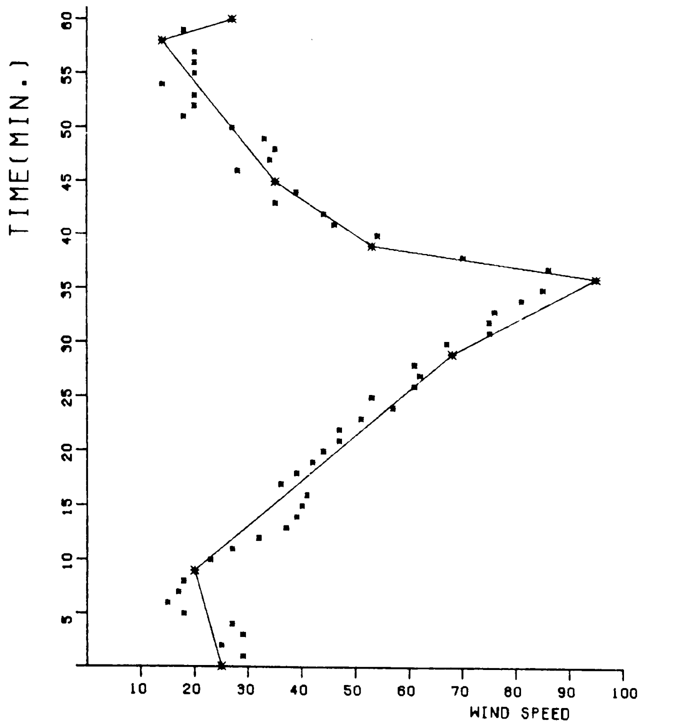

FIG.10B MOST REGU_ARLY SPACED SOLUTION

5.4.2 Temperature and humidity profiles

In modern fully automated rawinsonde observations special process computers take over the convential graphical and manual techniques in search of the set of significant levels in temperature and humidity profiles. The physical parameter values are recorded at intervals of a few seconds as a function of time elapsed after release. By eliminating time as an intermediary the data may also be given as a function of log pressure, height or geopotential.

When the data during the observation are stored on a suitable medium they may be searched for the set of significant levels by applying a suitable path finding algorithm. The utilization of a shortest path algorithm is feasible but given the large volume of data the process time may become quite uneconomical.

Fig. 11 is a simple computer plot of a hypothetical temperature profile diagram, T versus height, involving a sampling interval of 6 seconds (~ 35 meter). The profile has been obtained by introducing a vertex set of 17 significant levels adding data through interpolation and using a random generator. The resulting graph comprises 453 profile points. Next the shortest path algorithm was applied to see to what extent the original vertex set was reproduced given an aperture of 0.2°C. In Fig. 11 the reproduced levels of an optimal solution are indicated to the left of the original levels. The agreement of both vertex sets is excellent but the process time on the computer appeared to be impractical (on the Burroughs B6800: 30 minutes).

To avoid inadmissable computer process times in practical operations a simple pathfinding algorithm is used. This algorithm is based on the lexicographic ordering of the profile points. Directly after the release of the sonde the process computer examines the correspondence Γ to the effect that with each incoming signal it is

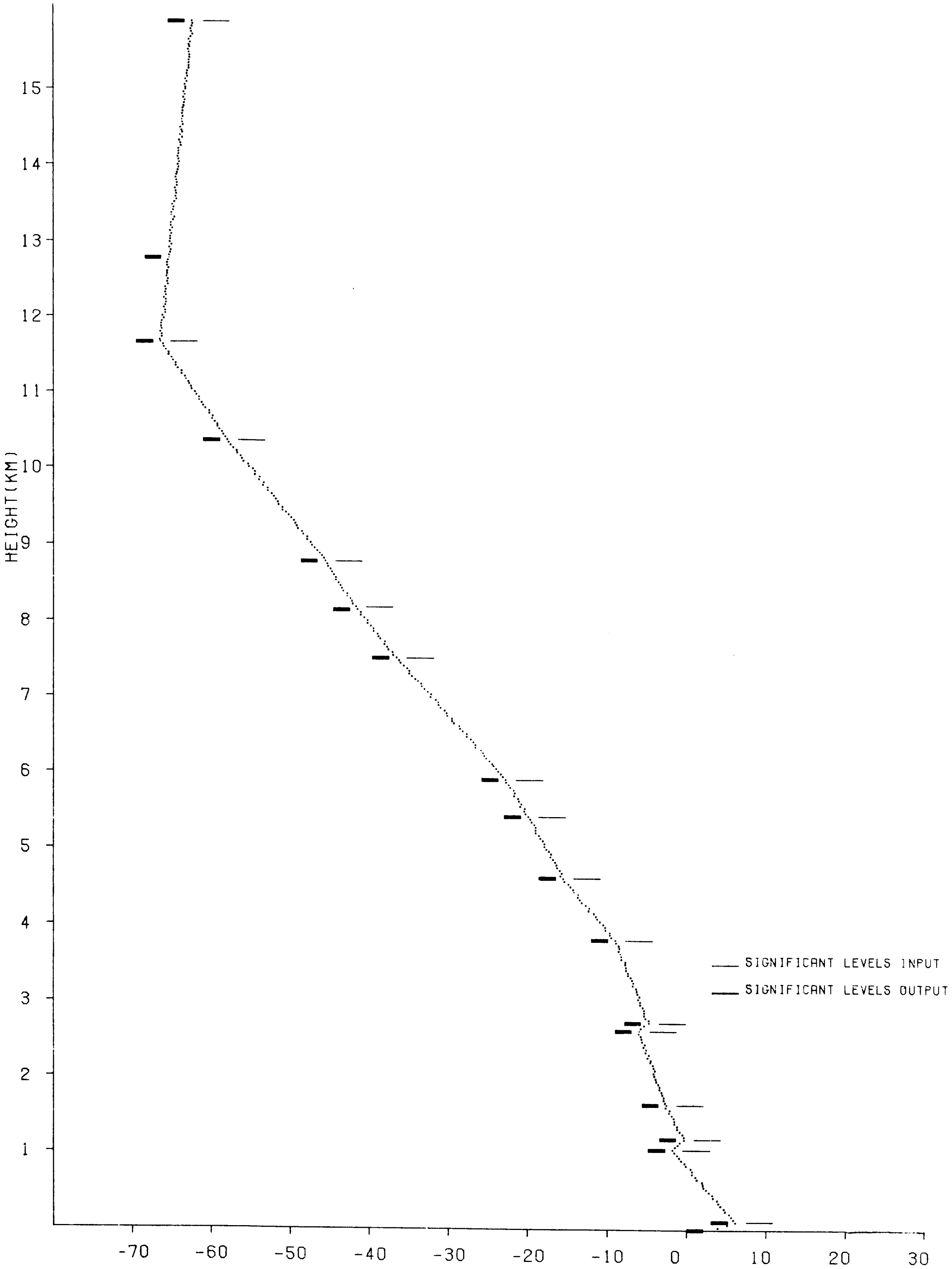

FIG.11 HYPOTHETICAL TEMPERATURE PROFILE.

VERTEX SET SOLUTION PRODUCED BY

SHORTEST PATH METHOD (APERTURE 0.2°C).

checked that the aperture condition, e.g. 0.2°C, is still fulfilled in regard to the foregoing vertex selected. As soon as the condition does not hold any more the last point is added to the vertex set and the process is continued until the top is reached and added as last profile vertex.

For humidity profiles the same procedure can be followed as for temperature profiles utilizing a humidity aperture of about 5 per cent. And lastly when both the temperature and humidity profiles are explored simultaneously a "compound" aperture statement may be used.

5.5 Results of experiments

The vertex set of optimal polygonal solutions for wind profiles is very sensitive to real meteorological fluctuations, instrumental noise, measurement range and aperture criteria. In this respect strong upper winds are easier to handle than weak winds: with weak upper winds the fluctuations and noise dominate and these will considerably enlarge the vertex sets of optimal solutions.

In order to get an impression of the overall efficiency of the data reduction when generating optimal vertex sets we have subjected the data of four ascents daily (00, 06, 12 and 18 GMT) at De Bilt in a 6-day period to the following computation methods:

a) WMO-method on the speed profile (aperture 10 knots)

b) Shortest path method on the speed profile (aperture 10 knots)

c) WMO-method on the wind direction profile (aperture 10 degrees)

d) Shortest path method on the wind direction profile (aperture 10 degrees)

e) Merging of the vertex sets of the WMO methods a) and c)

f) Shortest path method on both the speed and direction profiles (10 knots, 10 degrees)

The 24 cases having been calculated we compared the results 7a and b, c and d and e and f. The comparison of the WMO and shortest path methods (a and b, c and d) gives an indication of the overall data reduction capability when solving the optimization problem. The extra reduction in data attributable to the shortest path method, applied to the wind-speed profile curve (cases a and b) ranged from 0 to 66 per cent, with an average of 31 per cent. Applied to the direction profile it ranged from 0 to 33 per cent, on the average 11 per cent.

In Appendix B it is explained that when merging the vertex sets of case a and c the union set need not satisfy the imposed criteria for wind speed and wind direction anymore.

Setting this aside for the moment it is interesting to investigate the extra gain in data reduction when the data are subjected to the shortest path algorithm using both the aperture criterium for the wind speed and direction (cases e and f). The extra reduction of data ranged from 6 to 39 per cent, on the average 23 per cent.

A number of experiments was carried out using the simple criterium for the wind vector departure. Here a direct comparison with the traditional method is not well possible. A vector departure of 5 to 7.5 knots appears to be well matched with the known WMO standards of 10 knots and 10 degrees.

It is a relevant question to ask what bounds are really the best with regard to the aperture or tolerance width. Are the WMO standards themselves adequate? Current observational techniques do suggest that the aperture criteria could best be adapted directly to the observational error statistics or to a quantitative measure of the accuracy of the measurement. When this principle is followed the bounds should be variable. For example when processing rawinsonde observations these bounds should at least be height dependent. Experiments underlying this principle performed on winds showed a reduction of the optimal vertex sets in the upper (stratospheric) layers.

As to the simple method based on lexicographic ordering the number of elements of the corresponding vertex set proved to lie somewhere between that of the WMO method and the shortest path method.

5.6 Special cases

A. All pathfinding algorithms, the shortest path algorithm included, have the weakness that very pronounced levels like tropopause, maximum winds, inversions may be reproduced unsatisfactory or be missing from the vertex sets of significant levels. This is simply because of deficiences inherent in the methods (cf. Appendix B). To be sure that such pronounced levels are present in the vertex set, one can simply call the routines in subranges bounded by these obligatory levels, for example the subrange between the starting point and the maximum wind and the subrange between the maximum wind and the ending point.

In the subranges themselves multiple solutions are possible. Evidently the obligatory levels in the final vertex set are pivot points in the multiple "compound" solutions.

B. It may be required that the vertices should not surpass a given spacing. This will enlarge the vertex set with some dummy points. Then, in order to solve the profile problem, the aperture conditions need simply to be extended with an extra spacing condition when the set A of the graph G(X, A) is specified.

A direct consequence of the spacing restriction is that the process time on the computer is reduced drastically. For example when applied to the temperature profile of Fig. 11, limiting the spacing of vertices to 3 or 4 km, the process time is merely a 10 per cent of that without a spacing restriction.

C. The cardinality of the vertex sets or sets of significant levels appears to be strongly variable. In particular when the upper winds are weak the vertex sets may become too large making the PILOT and TEMP messages too extensive for encoding and distribution. A whole gamut of artifices may be tried out to remedy this so that the number of significant levels stays within practical limits. One of the means to accomplish this is by iteration: change at each step the aperture conditions so that the cardinality of the vertex set falls within a certain range. As the cardinality is an integral step function of the aperture value this requires a special technique (see Fig. 12).

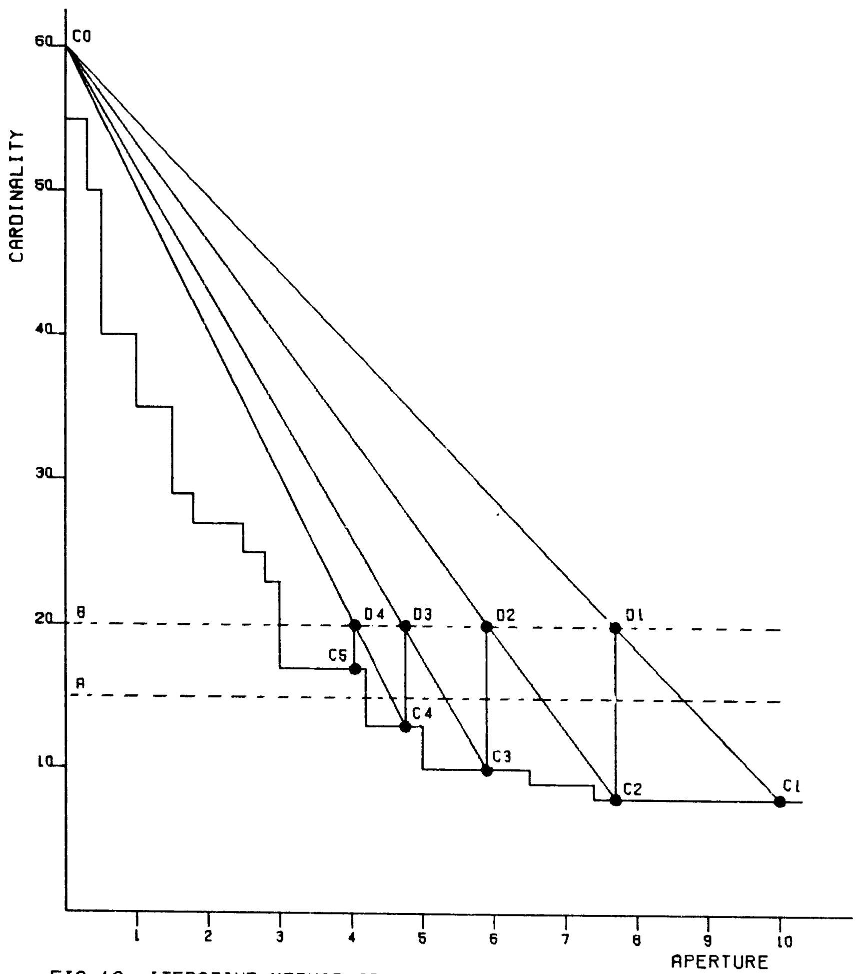

FIG.12 ITERATIVE METHOD.SEE TEXT

When requiring the cardinality to stay within a certain interval <a, b>, the following iteration method can be proposed: generate the shortest path algorithmic computation based on an arbitrary aperture value to start with. Let the cardinality of the corresponding vertex set be denoted by the point C1. If the aperture value is zero the corresponding cardinality equals the number n of data points in the original data set, point C0 in Fig. 12. Connect C1 and C0 and determine the intersection point D1 of $\overline{C0C1}$ and a boundary line a (or boundary line b). Re-activate the algorithm using the aperture belonging to the intersection point D1. A new vertex set is found and cardinality denoted by the point C2. Connect C0 and C2 giving point D2 and continue until the point Ci is in the interval <a, b>. If Ci never arrives in <a, b> there is no solution. The last point Ci is then a compromise solution.

D. There is a class of profile problems which involves more than one option at a time e.g. the class of aircraft meteorological observations. Temperature, pressure and wind data are known in a sequence of positions along the flight track. The scope of the data reduction problem becomes narrower then, its solution is more straightforward because the aperture conditions for entirely different options can be summarized in one logical expression such as the aperture conditions for temperature and wind. This case will be elaborated further in the next section using a sequence of AIDS/ASDAR observations. Incidently, this will hightlight also the use of data reduction methods in other than vertical profiles.

5.7 Data reduction in a series of AIDS/ASDAR observations

In the framework of the first GARP global experiment a number of prototype ASDAR units was built and tested. This was followed by a new generation of ASDAR units built to production standards in 1985. Prior to the international co-ordination of the ASDAR programme one could already have access to a category of ASDAR-type data stored on magnetic media, called AIDS (Aircraft Integrated Data System).

The sampling rate applicable in AIDS/ASDAR data is potentially very high. The exchange of high frequency data with a sampling period in the order of seconds is at the limit of the capacity of satellite communication. This necessitates a drastic preselection and reduction of the data to be carried out.

The new ASDAR unit will be a version developed from that used during FGGE. There will be substantial improvements however. The temperature will be of higher accuracy (1-2°C). In normal flight data will be collected at seven minute intervals, in ascent and descent they will be collected at 50hPa intervals except for the 100 hPa layer adjacent to the ground where they will be collected at 10 hPa intervals.

When in the future a great number of aircraft will be equipped with ASDAR or ASDAR-like systems the supply of data may be so copious that further data reduction is required.

Individual ASDAR data series define trajectory profiles including ascent and descent. Such profiles may be approximated by polygons in the same way as those of rawinsonde observations under the condition that the data are given as a function of a monotonically increasing or decreasing parameter value. Time or distance suit the purpose here. The aperture conditions refer to two options: temperature and wind. They can be grouped together in one logical expression.

Although easy to perform along the whole flight, we used the shortest path method only for the en-route portion of the flight between top of climb and top of descent.

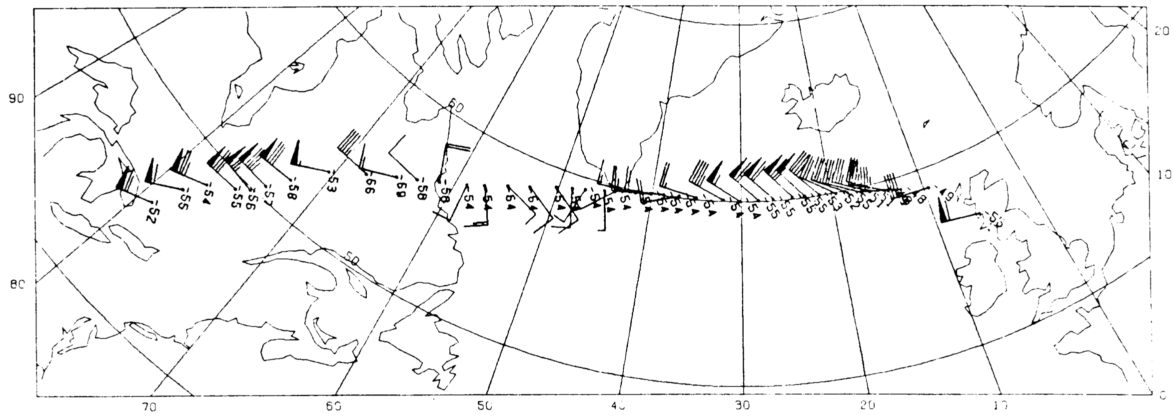

FIG.13A PLOT OF AIDS/ASDAR DATA SERIES FOR A FLIGHT
DIRECT AMSTERDAM-TORONTO ON 13/8'1974

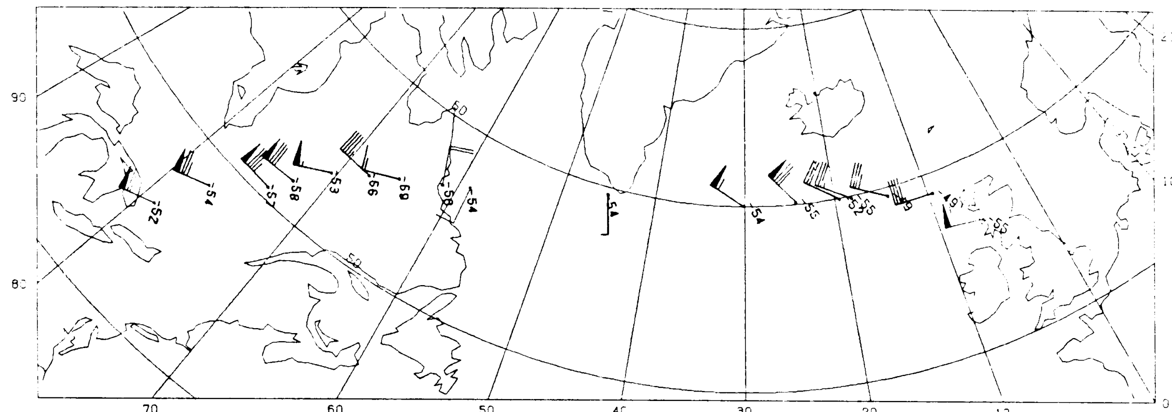

FIG.13B OPTIMAL VERTEX SET PRODUCED BY SHORTEST PATH
METHOD BASED ON APERTURE TEMPERATURE 2 °C
AND APERTURE WIND 20 KNOTS

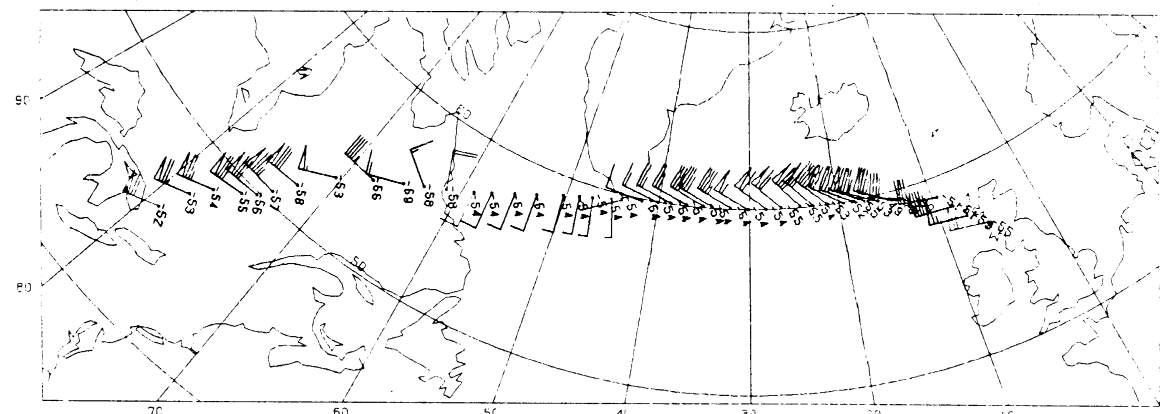

FIG.13C RE-CONSTRUCTION OF THE DATA USING THE
OPTIMAL VERTEX SET AND INTERPOLATION

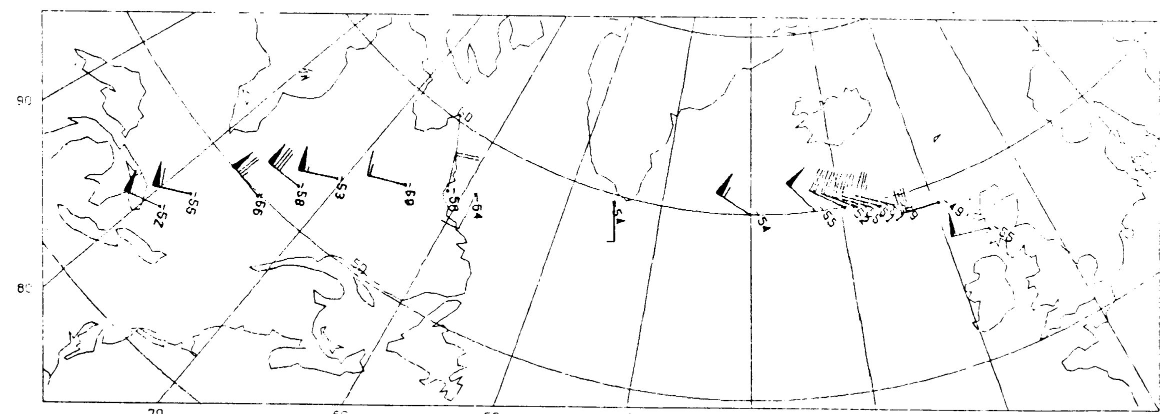

FIG.13D OPTIMAL VERTEX SET PRODUCED BY SHORTEST PATH
METHOD BASED ON TEMPERATURE APERTURE ONLY.

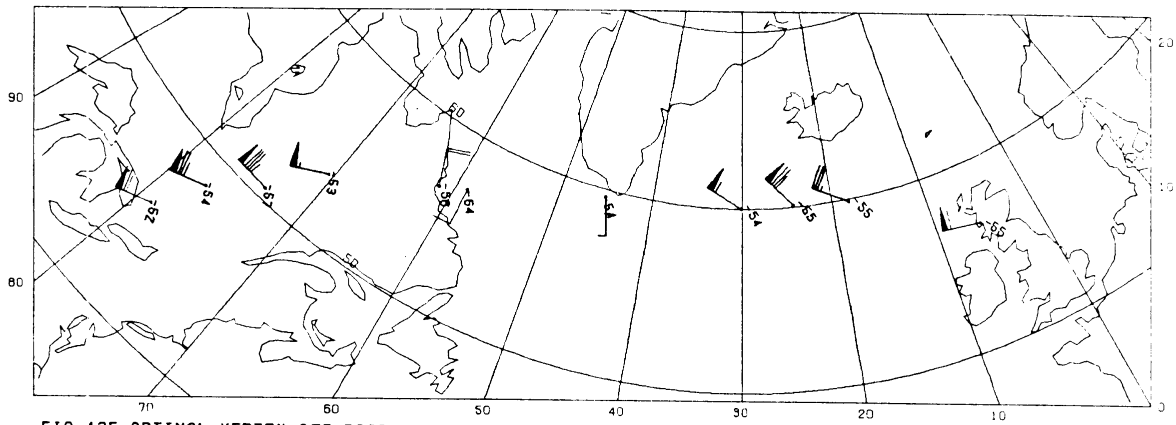

FIG.13E OPTIMAL VERTEX SET PRODUCED BY SHORTEST PATH METHOD BASED ON WIND APERTURE ONLY.

Fig. 13a depicts a plot of the meteorological data as drawn from an AIDS data cassette describing a flight from Amsterdam to Toronto. As wind vector departure we took 20 knots and a temperature departure of 2°C.

These are quite coarse values but this is justified by the somewhat relaxed data requirements of ICAO in aviation and the relatively poor accuracy of temperature reporting in the prototype ASDAR equipment.

First a transformation is carried out converting the geographical co-ordinates into Cartesian co-ordinates. The directed acyclic graph $G(X, \Gamma)$ is here composed of the set X of data points along track and the correspondence Γ adding to each vertex x_i those vertices $x_j (j>i)$ for which $|\Delta T| \leqslant 2°C$ and $|\Delta\vec{v}| \leqslant 20$ knots along the arc $x_i x_j$.

ΔT and $\Delta\vec{v}$ are departures of the observed values from the linearly interpolated data along the connection $x_i x_j$ in the Cartesian co-ordinate system. Fig. 13b shows the (optimal) vertex set generated by the shortest path algorithm. The cardinality of the set is 17. Wind-maxima, troughs ridges and other pronounced features are clearly discernable.

Using this optimal vertex set the trajectory profile may be reconstructed by interpolation. The result of this is represented in

Fig. 13c. The algorithm could also have been put in action for the graph G when the correspondence Γ would have been based on a temperature aperture criterium only or a wind aperture criterium only. The resulting vertex sets could then have been merged producing one union set. Fig. 13 d shows a plot of the vertex set corresponding to an algorithmic computation based on the temperature aperture condition only ($|\Delta T| \leqslant 2°C$). The cardinality of the set is 18. Fig. 13e is a plot of the vertex set obtained when run for the wind vector aperture condition only ($|\Delta\vec{v}| \leqslant$ 20 knots). The union set (cardinality = 22) is not represented here. This set is not necessarily identical to the optimal set of Fig. 13b acquired with the "compound" aperture condition. The union set in general is larger then the optimal set. The aperture conditions need not be fulfilled anymore in the union set for the same reasons as experienced in the profiles produced by rawinsonde observations.

The data reduction technique such as described above can be carried out also for the total flight provided that the profiles are given in terms of time or distance. Characteristic navigational points may be introduced as compulsory points and the spacing of vertex points may be restricted just as with radiosonde profiles.

6. Discussion and conclusion

By formulating a problem in graph theoretical terms we incur the limitations and pitfalls, as well as the advantages, of graph theory.

Here we cite HARARY [8]:"While it is very interesting to have accurate estimates of the order of magnitude of a solution, as well as lower and upper bounds, these will not be regarded as settling the problem. There is also plenty of room for differences in opinion among enumeration experts themselves. At one extreme there is the function logician who regards all such problems as trivial since each involves only a finite number of steps, and hence can be settled by brute force. Then there are some physicists who consider the question closed when the first few cases are calculated. The computer-oriented type feels that all is right when a program has been written for the problem. Another more strict viewpoint does not accept recursion relations as embedded in cycle indices of presentation groups but insists on explicit and elegant formulae only. And finally there is the purist who demands not only a formula for the number of graphs of each kind, but also all the diagrams of the graphs themselves. His needs are a bit more difficult to satisfy".

One of the characteristic features of graph theory is that many problems have multiple solutions. The multiplicity of solutions was manifest in all problems we have examined in this report. It is always a challenge to explore which member of the family of these solutions rivals the others when putting forward a certain aspect.

In the profile problem for example, the most regular spaced solution was sought. We could have done the same in two dimensions with respect to the plotting problem.

When we impose to each vertex or link of a graph a number c_i, the "costs", we may ask for the minimal cost solution. Costs in metaphorical sense referring to some or other attribute (capacity, real costs, length, probability). A famous problem of this kind in graph theory is the travelling salesman problem: seek amongst all Hamiltonian paths or circuits, provided these exist for the given problem setting, the one having the lowest link costs.

Should, by way of example, "costs" in the vertices of a graph of the plotting problem relate to a normative measure of the information content of weather reports, then it would be very interesting to seek for that member of the family of solutions which provides for a maximum of (meteorological) information.

<u>note</u>: this surely enhances the importance of the way charts are plotted especially when a critical weather situation prevails. The selective information may be represented by the expression $-k\sum_{i}^{I} p_i \log p_i(x_i)$, where p_i is the probability of occurence of the state x_i and k is a conversion factor (towards number of bits). The summation is over all reported physical states.

The matrix, directly involved in the definition of a graph G and describing it completely, is the <u>incidence</u> matrix. Denoting this matrix by A, its elements a_{ij} are equal to one when there is a link between the vertices i and j, otherwise they are zero.

A close inspection of the graph problems reveals that this incidence matrix is a powerful tool in trying to solve graph problems.

Let we consider the profile problem. Then it can be shown that all non-zero elements of the matrix $A^k = A \times A \times \ldots A$ (k times) define the point-pairs in the graph for which path solutions exist consisting of k links. The value of the element in A^k precisely equals the multiplicity of solutions between the point pair. When for a certain n A^{n+1} is the null matrix, A^n is the matrix containing the path information (point pairs and multiplicity) of all shortest paths composed of maximally n links.

<u>note</u>: The matrix A^n still lacks information on the identity of the paths themselves. To fill this gap one may relate the problem to the famous <u>random walk problem</u>. There the matrix A may be modified in such a way that A^n also contains information on the paths themselves. For more details cf. HARARY [8] and KASTELIJN [9].

When new data reduction techniques are considered for operational use in meteorological practice it is obviously necessary to assess practically useful tolerance widths or apertures. These need to be determined empirically. WMO limits established for encoding reports of upper winds need to be reviewed as soon as it is decided to introduce new reduction procedures. A method such as proposed for the data reduction of upper winds based on a vector departure probably requires a value of 5-7.5 knots for this departure.

7. Acknowledgement

The author is most grateful to Dr. A.P.M. Baede, Drs. A.W. den Exter Blokland and Dr. P. Lynch for careful reading of the paper giving inspiring criticism and pointing to some errors.

Thanks are also due to Ms. Birgit Kok for the excellent typing of the manuscript and the skilful use of a word processor.

8. References

[1] Berge, C., 1962: The theory of graphs, Methuan, London.

[2] Bron, C. and Kerbosch, J., 1973: Algorithm 457 - Finding all cliques of an undirected graph. Comm. of ACM, 16, pg. 575-577.

[3] Busacker, R.G. and Saaty, T.L., 1965: Finite graphs and networks. Mc.Graw Hill, New York.

[4] Chen, W.K., 1971: Applied graph theory. North Holland, Amsterdam.

[5] Christofides, N., 1975: Graph theory. An algorithmic approach. Academic Press, New York, London, San Francisco.

[6] Dijkstra, E.W., 1959: A note on two problems in connection with graphs. Numerische Mathematik 1, pg. 269.

[7] Harary, F., 1969: Graph theory. Addison-Wessley, Reading, Massachusetts.

[8] Harary, F., 1967: Graph Theory and Theoretical Physics. Academic Press, London, New York, pg. 17.

[9] Kastelijn, P.W., 1965: Graph theory and crystal physics. Lorentz Institute for Theoretical Physics, University of Leiden.

[10] Moon, J.W. and Moser, L., 1965: On cliques in graphs. Israel JL of mathematics, pg. 23.

Appendix A

Algorithm of Bron-Kerbosch for the generation of all cliques in a non-directed graph.

We present here the ALGOL program listing of the Bron-Kerbosch algorithm. For an explanation we advise the reader to consult the exposé in the original paper: 1973, Communications of the ACM, Vol. 16, No. 9, pg. 575-577.

```
Algorithm
procedure maximal complete subgraphs (connected, N);
  value N; integer N;
  Boolean array connected;
comment The input graph is expected in the form of a symmetrical Boolean
  matrix connected. N is the number of nodes in the graph. The values of the
  diagonal elements should be true;
begin
  integer array ALL, compsub [1:N];
  integer c;
  procedure extend version (old, ne, ce);
    value ne, ce; integer ne, ce;
    integer array old;
  begin
    integer array new [1:ce];
    integer nod, fixp;
    integer newne, newce, i, j, count, pos, p, s, sel, minnod;
    comment The latter set of integers is local in scope but need not be
            declared recursively;
    minnod: = ce; i: = nod: = 0;
DETERMINE EACH COUNTER VALUE AND LOOK FOR MINIMUM:
    for i: = i + 1 while i ≤ ce ∧ minnod ≠ 0 do
    begin
      p: = old [i]; count: = 0; j: = ne;
COUNT DISCONNECTIONS:
      for j: = j + 1 while j ≤ ce ∧ count < minnod do
           if   connected [p, old[j]] then
           begin
             count: = count + 1;
SAVE POSITION OF POTENTIAL CANDIDATE:
             pos: = j
           end
TEST NEW MINIMUM:
      if count < minnod then
      begin
        fixp: = p; minnod: = count;
        if i ≤ ne then s: = pos
        else
        begin s: = i; PREINCR: nod: = 1 end
      end NEW MINIMUM;
    end i;
```

```
    comment if fixed point initially chosen from candidates then number of
      disconnections will be preincreased by one;
BACKTRACKCYCLE:
    for nod: = minnod + nod step - 1 until 1 do
    begin
INTERCHANGE:
      p: = old[s]; old[s]: = old[ne + 1];
      sel: = old[ne + 1]: = p;
FILL NEW SET not:
      newne: = i: = 0;
      for i: = i + 1 while i ≤ ne  do
        if connected[sel, old[i]]  then
        begin newne: = newne + 1; new[newne]: = old[i] end;
FILL NEW SET cand:
      newce: = newne; i: = ne + 1;
      for i: = i + 1 while i ≤ ce do
          if connected[sel, old[i]] then
          begin newce: = newce + 1; new[newce]: = old[i] end;
ADD TO compsub:
      c: = c + 1; compsub[c]: = sel;
      if newce = 0 then
      begin
        integer loc;
        outstring(1, 'clique = ');
        for loc: = 1 step 1 until c do
              outinteger(1, compsub[loc])
      end output of clique
      else
      if newne < newce then extend version (new, newne, newce);
REMOVE FROM compsub:
      c: = c - 1;
ADD TO not:
      ne: = ne + 1;
      if nod > 1 then
      begin
SELECT A CANDIDATE DISCONNECTED TO THE FIXED POINT:
        s: = ne;
LOOK: FOR CANDIDATE:
        s: = s + 1;
        if connected [fixp, old[s]] then go to LOOK
      end selection
    end BACKTRACKCYCLE
  end extend version;
  for c: = 1 step 1 until N do ALL[c]: = c;
  c: = 0; extend version (ALL, 0, N)
end maximal complete subgraphs;
```

Notice that during the extension of the subgraph always a point outside the subgraph is added that has the most connections with other points of the subgraph.

Appendix B

Polygonal approximation of the "wind profile"

(See FM 32-V PILOT, FM 33-V PILOT SHIP, FM 35-V TEMP, FM 36-V TEMP SHIP)

Criteria for significant levels in the wind profile are given in code regulation 32.3.1.1. of the Manual on Codes, WMO-No. 306. In a NOTE a computational procedure is recommended which may suit most national practices. The regulation and NOTE read as follows:

32.3.1.1

Significant levels

The reported significant data alone shall make it possible to reconstruct the wind profile with sufficient accuracy for practical use. Care shall be taken that:

(a) The direction and speed curves (in function of the log of pressure or altitude) can be reproduced with their prominent characteristics;

(b) These curves can be reproduced with an accuracy of at least 10 for direction and five metres per second for speed;

(c) The number of significant levels is kept strictly to a necessary minimum.

NOTE: To satisfy these criteria, the following method of successive approximations is recommended, but other methods of attaining equivalent results may suit some national practices better and may be used:

(1) The surface level and the highest level attained by the sounding constitute the first and the last significant levels.
The deviation from the linearly interpolated values between these two levels is then considered. If no direction deviates by more than 10 and no speed by more than five metres per second, no other significant level need be reported. Whenever one parameter deviates by more than the limit specified in paragraph (b) above, the level of greatest deviation becomes a supplementary significant level for both parameters.

(2) The additional significant levels so introduced divide the sounding into two layers. In each separate layer, the deviations from the linearly interpolated values between the base and the top are then considered. The process used in paragraph (1) above is repeated and yields other significant levels. These additional levels in turn modify the layer distribution, and the method is applied again until any level is approximated to the above-mentioned specified values. For the purpose of computational work, it should be noted that the values derived from a PILOT report present two different resolutions:

(a) Winds at significant levels are reported to the resolution of 5° in direction and one metre per second in speed;

(b) Any interpolated wind at a level between two significant levels is implicitly reported to the resolution of ± 10° in direction and ± 5 metres per second in speed.

The method of successive approximation is generally used in manual and automatic means of processing of rawinsonde observations. It should be realised however that the method has the following weaknesses:

(i) we cannot be certain that the method affords strictly a minimum number of significant levels.

(ii) the method is no guarantee that the reported significant levels alone make it possible to re-construct the wind profile with accuracies as prescribed under items (b) and (c).

For, let S_1 be the set of data meeting the 5 ms^{-1} speed criterium and let S_2 be the set of data meeting the 10° direction criterium.

Then the set S obtained by merging S_1 and S_2 is the final set of significant data to be entered in the code.

In so doing it is tacitly assumed that the set S will meet both the speed and direction limits of 5 ms^{-1} in speed and 10° in direction provided that S_1 and S_2 did meet already the criteria for speed and direction separately.

This is not necessarily true, however. It may be easily shown theoretically and it has empirically been confirmed that after merging S_1 and S_2 the deviation from the linearly interpolated values in the set S may in some cases suddenly surpass one or both of the prescribed limits.

The difficulties arise here when the vectorial wind is resolved in its components and the wind profile is split up in a separate direction and speed profile. It seems a better approach to handle criteria in a wind vector profile directly.

A further complication is that the recommended method does not guarantee that the real maximum wind level(s) are included in the set S_1 for significant wind-speed data, for the following reason: when the speed value at the highest level attained by the soundings is considerably higher or lower than the speed value at the surface level the slope of the straight line connection may cause the greatest deviations not to be found at the maximum wind level, cf. Fig. B1. The wind speed curve consists of the minute-points of the speed versus time. The coded maximum wind is different from the real maximum and caused by the steep slope of the straight line connection between the top and bottom levels.

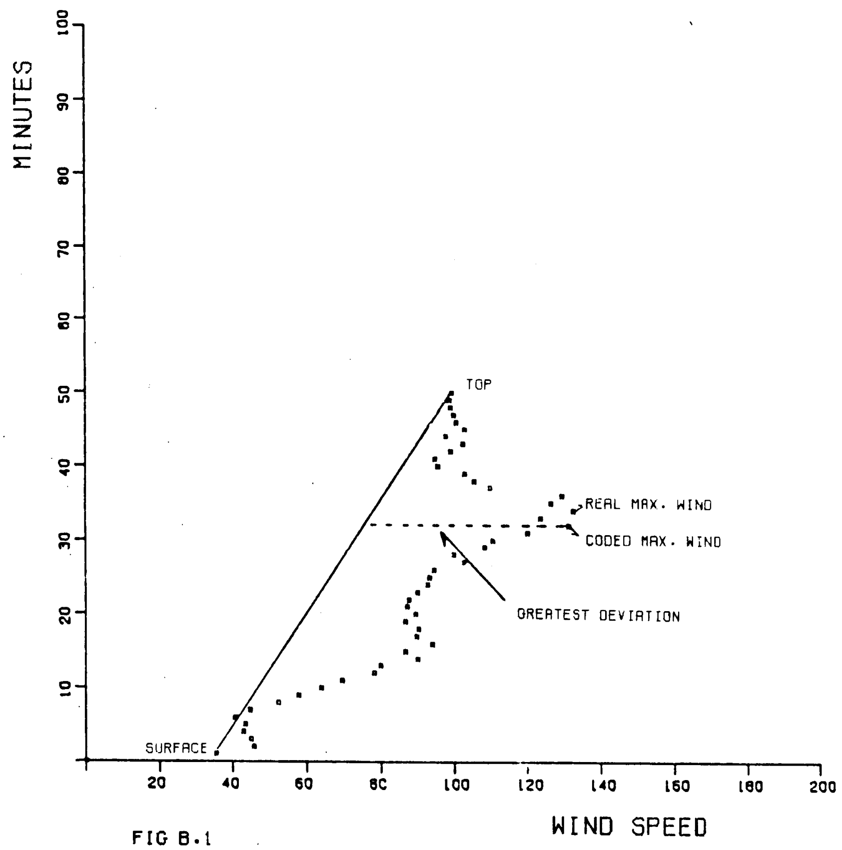

FIG B.1

Appendix C

Finding all shortest path in a directed acyclic graph

Description

In a general arc-weighted graph $G(X,\Gamma)$ with 'arc lengths' specified by a matrix, the shortest path problem is the problem of finding the shortest path from a specific starting vertex s X to a specific ending vertex t X, provided that such a path exists i.e. provided t is an element of the reachable set of the vertex s. The elements of the matrix are positive here (we are considering the case where the elements are unity) so that the shortest length is simply a count of all arcs constituting the shortest path!

DIJKSTRA [6], in a one-page note in 1959, proposed a computational procedure for such graphs which, since then, has proven to be one of the most efficient procedures.

In particular, when the graph is directed and acyclic the algorithm may be simplified.

It is assumed that the vertices are numbered such that an arc $x_i x_j$ is always directed from x_i to a higher numbered x_j. The starting vertex is s, the ending vertex t. To label vertex x_j with $\ell(x_j)$, the shortest path length from s to x_j, perform the operation

$$\ell(x_j) = \min_{x_i \in \Gamma^{-1}(x_j)}\{\ell(x_i) + 1\} \qquad C(1)$$

Then continue to label vertex x_{j+1} using this expression until the ending vertex t is labelled $\ell(t)$. $\ell(s)$ is initially set to zero. Obviously, when labelling vertex x_j, the labels $\ell(x_i)$ are all known for the vertices $x_i \in \Gamma^{-1}(x_j)$. The label $\ell(t)$ is the length of the shortest path from s to t.

When there is only one solution the arcs forming the path itself may be found in the usual way by tracing backwards so that arc $(x_i x_j)$ is on the path if and only if

$$\ell(x_j) = \ell(x_i) + 1 \qquad C(2)$$

Starting with x_j equal to t, x_j is set at each step equal to the value of x_i say x_i^* satisfying the last equality C(2) recursively until $x_i^* = s$, i.e. the initial vertex has been reached.

As the label $\ell(x_j)$ represents a count and takes on an integer value, the problem has in general a multiplicity of solutions. Then there may be more than one point x_i satisfying the recursive equation C(2).

In this case one could either make an arbitrary choice of x_i for one specific solution or one could try to find a method to obtain the totality of solutions.

With the last option in mind it is worthwhile to attach to each vertex x_j an additional label $m(x_j)$ representing the multiplicity of all shortest paths from s to x_j. The "bookkeeping" of this multiplicity label $m(x_j)$ takes place together with the operation C(1) as follows: if the length of the path between s and x_j via $x_i = \Gamma^{-1}(x_j)$ increases in respect to a foregoing path between s and x_j, the multiplicity does not change. If it decreases then $m(x_j) := m(x_i)$ and when it remains the same $m(x_j) = m(x_i) + m(x_j)$.

At the start m(s) is set equal to 1.

<u>Note</u>: Considerations of symmetry make that one could equally well ask for the label denoting the length of shortest paths from x_j to the ending point t. Let we indicate this label by $\ell'(x_j)$. Then apparently the property holds that for vertices along a shortest path solution:

C(3) $\ell'(x_j) + \ell(x_j) = \text{constant} = \ell'(s) = \ell(t)$ C(3)

As the shortest paths all comprise the same number of graph vertices the paths can be registered in the column sets of a tableau with the number of columns equal to the multiplicity and the number of rows equal to the path length plus one.

Starting with the ending point t one determines first all $x_j \in \Gamma^{-1}(t)$ for which $\ell(x_j) = \ell(t) - 1$.

The vertices, say x_j^*, satisfying this condition are substituted in the last but one row of the tableau.

When $\forall x_j^*$ stands for "for all x_j^*", the property holds $\sum_{\forall x_j^*} m(x_j^*) = m(t)$.

The vertices x_j^* are recorded in the tableau and copied in the row a number of times equal to $m(x_j^*)$.

This goes on so that at a certain step of the backtracking program one seeks for the points x_i for which

$$\ell(x_i) = \ell(x_j^*) - 1, \quad \forall x_j^* \qquad \text{C(4)}$$

It should be remarked here that the same point x_i may correspond to different points x_j^* as many times as it satisfies C(4). This of course holds for the totality of points x_j^*.

The vertices x_i satisfying the condition C(4), denoted by x_i^*, are stored in the corresponding row in the tableau and copied a number of times equal to $m(x_i)$ in there. For the totality of vertices x_i^* the equality

$$\sum_{\forall x_i^*} m(x_i^*) = \sum_{\forall x_j^*} m(x_j^*) = m(t)$$

holds. The process is continued until finally the starting point s is reached.

Once the substitution is finished the tableau indeed contains all information required for identification of all the shortest path solutions between s and t. The information is available in the form of ranking numbers of the vertices of the paths in the columns (cf. 5.4.1, page 28).

The ALGOL 60 implementation of the vertex labelling of shortest paths and the preparation of the tableau presenting the information of the totality of solutions is given below.

Algorithm

```
begin integer array multiplicity, minlength [1:N];
  Boolean array connected [1:N]
comment The input graph is expected in the form of a Boolean matrix connected.
  N is the number of vertices in the graph. The values of the diagonal
  elements should be false;
procedure netpath (s,t); value s, t; integer s, t;
  comment s is the index related to the starting vertex; t is the index
          related to the ending vertex;
  begin integer i, j, mini, length;
    for i: = 1 step 1 until N do
    multiplicity[i]: = minlength[i]: = 0;
    minlength[s]: = 0;
    multiplicity[s]: = 1;
    for i: = s step 1 until t do
    begin
      mini: = minlength[i];
      for j: = i + 1 step 1 until t do
      begin
        length: = mini + (if connected [i,j] then 1 else 2xN);
        if length < minlength then
        begin multiplicity[j]: = multiplicity[i] +
              (if length < minlength [j] then 0 else multiplicity [j]);
              minlength [j]: = length;
        end;
      end j;
    end i;
  begin integer l, m, p, po, p1;
        integer array pathtableau [0: minlength[t], 1: multiplicity[t]];
      l = minlength (t);
      m = multiplicity (t);
RE-USE OF BOOLEAN MATRIX CONNECTED FOR TOTALITY OF SHORTEST PATHS:
      for i: = 1 step 1 until N do
      for j: = 1 step 1 until N do
        if minlength [i] + (if connected [i,j] then 1 else N)
        = minlength [j] then
        connected [j,i]: = true else false;
```

```
PREPARATION TABLEAU "PATHTABLEAU" FOR TOTALITY OF SHORTEST PATHS:
      for i: = 1 step 1 until m do
        begin pathtableau [l,i]: = t;
              pathtableau [0,i]: = s;
        end;
      for j: = l-1 step -1 until 1 do
        begin label iteration; po: = 0;
              iteration: if po < l then
        begin p1: = pathtableau [j+1, po + 1];
          for i: s step 1 until t do
          if connected [p1,i] then
          for p: = 1 step 1 until multiplicity[i] do
          pathtableau [j, po: = po + 1]: = i;
          go iteration;
        end;
      end j;
    end;
  end netpath;
end.
```