STUDIA UNIV. BABEŞ-BOLYAI, MATHEMATICA, XXXVI, 3, 1991

TOPOGRAPHICAL DATA MANAGEMENT SYSTEM

CRISTIAN LENART*

Received: August 4, 1991
AMS subject classification: 68T10

Rezumat. - Sistem de gestiune a datelor topografice. Articolul prezintă un sistem original de gestiune a datelor topografice implementat sub sistemele de operare RSX şi PC DOS. Informaţia preluată de pe hărţi o constituie coordonatele punctelor de observaţie (puncte în care s-au efectuat anumite determinări calitative sau cantitative), precum şi entităţile grafice curbe, regiuni, semne convenţionale, texte). Culegerea datelor se realizează prin digitizare, sub controlul unui editor grafic. Exploatarea bazei de date presupune extragerea şi reprezentarea datelor situate în fereastra de lucru, definirea de noi entităţi grafice, calcule simple (arii, medii ale unor funcţii de parametri cantitativi).

0. Introduction. The aim of this paper is to present an original topographical data management system for the acquisition and processing of data taken from maps. We may also consider, instead of maps, any kind of drawing consisting of curves, regions, conventional signs and texts. This system was implemented under the operating systems RSX and PC DOS.

1.Map entities. An item (data element) on a map will be called topographical entity. Two kinds of topographical entities are considered: observation points and graphical entities. Observation points are those points on a map where certain qualitative or quantitative parameters were determined. For instance, on a geological map, mineral resources and petrographical types are qualitative parameters while percentages of certain chemical elements are quantitative parameters. Graphical entities are curves (opened or closed), regions (areas

* *"BABEŞ-BOLYAI" University, Faculty of Mathematics, 3400 Cluj-Napoca, Romania*

delimited by a closed curve and filled with a certain colour), conventional signs (circles, triangles, cross-marks, etc.) and texts. Graphical entities are characterized by three features: their type, the plotting mode (for instance the type and the colour of line for curves, the filling colour for regions, the character set for texts) and the user code. This last feature is an integer associated by the user with each graphical entity in order to handle it easier. The position on the map of a graphical entity is defined by a variable number of points in a given order in the case of curves and regions and by a fixed number of points for the other two entities; thus, one point is needed to indicate the position of a standard conventional sign or text, while two points are needed for variable radius circles or inclined texts.

Observation points are numbered; their qualitative and quantitative parameters as well as the coordinates of their positions on the map are stored in separate files. A group of graphical entities is formed by a single entity with variable number of points or by several entities with fixed number of points and the same features. Three files are used to store graphical entities: an entity file, a coordinate file and a text file. The first one contains a record for each group of graphical entities; this record consists of three features of the graphical entities from the group (entity type *ET*, plotting mode *PM*, user's code *UC*) and two pointers *FP*, *LP* to the first and the last point in the coordinate file which define the position of entities. The coordinate file contains sequences of (x, y, z) - coordinates corresponding to the groups or graphical entities from the entity

file. Points indicating the position of a text have instead of the z - coordinate a pointer to the corresponding text in the text file. This file contains all the texts from the map; the end of each text and the letter of which position was indicated in the coordinate file (if not the first one) are marked. The described structure of graphical entities is given in fig. 1.

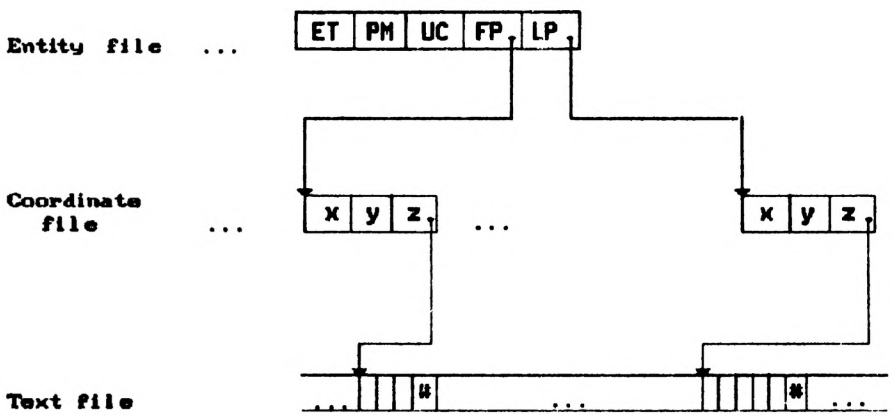


Fig. 1

2. **Data acquisition.** We are now concerned only with the acquisition of coordinates which indicate the position of topographical entities. This is done by means of a digitizer. The map may be fastened to its plane table in any position since the acquisition programs make a corrective rotation. The absolute coordinates are then computed according to the map scale and the coordinates of its origin. Thus, maps of adjacent zones can be assembled to form the general map.

Observation points can be digitized consecutively, in the order specified by the user, examining only not yet digitized points from a given interval or a certains group of points at a

time. In this latter case, we have to provide a file with the number of group to which each observation point belongs.

The acquisition of graphical entities is performed by a graphical editor which plots the entities on the display as they are stored into computer. A graphical cursor may be moved in the current window represented on the display (corresponding to a rectangular area from the map) by means of the arrow keys of the keyboard or a mouse and its coordinates are indicated. The position of the digitizer cursor on the display can also be indicated.

Graphical entities which are edited at a time are stored in the computer memory hierarchically, on four levels. This structure enables a quick performing of editing operations. Graphical entities are divided into fragments which are portions of curves or groups of conventional signs or texts placed or not in the current window. Fragment points are divided into sequences which are stored in certain memory locations called pages. We are now able to describe the tree structure of edited data (fig. 2). The first level is a two-way list of data groups referring to groups of graphical entities. Curves which do not intersect the current window at all do not appear in this list. A data group consists of the three features of the corresponding graphical entities (entity type *ET*, plotting mode, user's code *UC*) and two pointers *FF, LF* to their first and last fragments. On the second level are placed the two-way lists of data groups referring to fragments. Chaining of curve fragments observes the order in which they are placed on the curve. A data group contains the

fragment type *FT* (inside or outside the current window), a pointer *GE* to the group of graphical entities to which it belongs and two pointers *FP, LP* to the first and last point of the fragment. Only points of fragments inside the current window are stored in the internal memory; thus, for fragments outside the window, pointers *FP* and *LP* point directly to the coordinate file. On the third level are placed the two-way lists of pages, each one containing a pointer *PF* to the fragment to which it belongs and the sequence of data groups referring to points. Such a data group consists of the (*x, y, z*) - coordinates and two pointers *P1, P2*, which chain in a two-way list the points placed in the same square on the map. The number of squares into which the current window is divided is given by the user. This chaining enables a quick retrieval of points given a certain neighbourhood of them. For instance, when digitizing a point which has already been digitized, we can search for this point in a neighbourhood of the curently digitized point and replace the coordinates of the latter by those of the former. Connection of curves can thus be carried out without errors.

The following editing operations can be performed:

- acquisition of a graphical entity (coordinates from the digitizer, features and texts from the keyboard);

- prolongation of an open curve (the corresponding extremity of the curves is indicated using the graphical cursor);

- connection of two open curves or of the extremities of an open curve to form a closed one (the extremities of the curves/curve are indicated using the graphical cursor and then

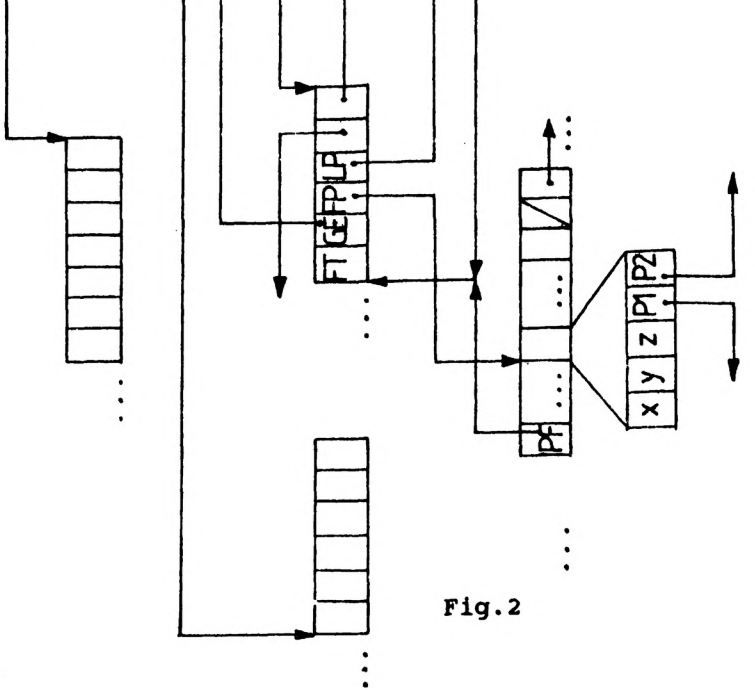
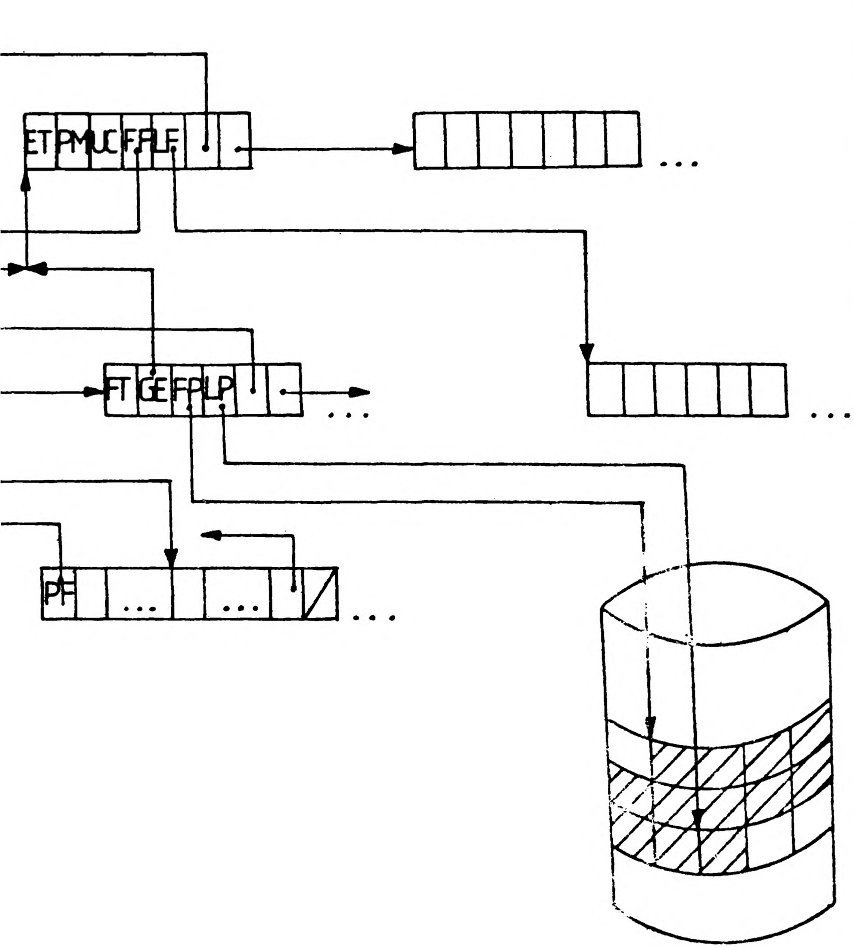


Fig.2


ET
PM
UC
FF
LF
FT
GE
FP
LP
PF

the new curve fragment is digitized);

- modification of a curve fragment (a certain curve fragment is deleted and then its extremities are connected by a new fragment);

- deletion of a curve fragment, of a curve entirely, of a conventional sign or text (the corresponding graphical entities are indicated using the graphical cursor);

- change of the features of a graphical entity;

- hardcopy of the current window;

- asking for help on the editing menu;

Deletion and feature change operations can be performed on several graphical entities at a time without viewing their effects. These entities are also selected by indicating their features (wild cards are permitted for some of the features).

3. **Data base enquiry.** Observation points retrieval is carried out in terms of the following criteria: point number, qualitative and quantitative parameters. Qualitative parameters are specified as codes while quantitative ones as intervals. Selected observation points can be classified according to the value of a grouping parameter; hence cluster identifier is a supplementary retrieval criterion. Retrieval criteria for graphical entities are their three features. Accessing of a map entity is followed by plotting it on the display. A curve can be plotted by joining its points with straight-line segments or by smoothing it. Smoothing can be carried out using a cubic spline interpolation or an original method which iteratively halves the

angle between two neighbour straight-line segments until these segments become sufficiently small. Thus, the user can construct a map of an area he desires and containing only the information he indicates. Moreover, he can add graphical entities, perform some elementary computations and blow up a rectangular area.

Once the map is thus constructed, certain entities on it can be identified using the graphical cursor. A temporary selection of observation points is considered. We now explain how entities are identified and what kind of operations can we perform on identified entities.

a) Operations involving a single observation point:

- viewing the observation point with a given number;
- finding the number of an observation point;
- inserting/removing an observation point in/from the temporary selection;
- displaying the quantitative parameters of an observation point or elementary functions of them.

b) Operations on groups of observation points. A group of observation points is formed by the points of a cluster, by the temporary selected points or by the points placed in a rectangular or arbitrary region indicated by the user. The following operations can be performed on such groups:

- displaying the numbers of observation points from the group;
- inserting/removing the points of the group from the temporary selection;
- means computation of the group points quantitative

parameters or functions of them.

c) Operations on graphical entities.

- adding/deleting a graphical entity on/from the display (we indicate its position using the graphical cursor);

- storing/removing a graphical entity in/from the data base;

- modification of graphical entity features;

- displaying the area of a region and the number of observation points placed in it (these parameters can be used together with the quantitative parameters of the observation points to compute the value of certain elementary functions depending on them);

- displaying the coordinates of the graphical cursor.

We conclude by mentioning that the system developed in this paper is a useful tool for the management of a data base containing topographical data which can be then used by other software to construct 3*D* plottings. Referring to the parameters of observation points, the system is also useful for a primary data analysis and for the interpretation of statistical processing or clustering results.

REFERENCES

1. Lenart, C., *Software for classification*, to appear in Studia Univ. "Babeş-Bolyai" Cluj-Napoca, 1991.
2. Russu, A., *Topografie cu elemente de geodezie şi fotogrametrie*, Ed. Ceres, Bucureşti, 1974.
3. Săndulache, Al., Sficlea, V., *Cartografie - topografie*, Ed. didactică şi pedagogică, Bucureşti, 1970.