OPTIMAL TILING FOR LARGE CARTOGRAPHIC DATABASES

Michael F. Goodchild
National Center for Geographic Information and Analysis
University of California
Santa Barbara, CA 93106

ABSTRACT

We define a tiling as a partitioning of a spatial database using geographical and thematic criteria. Several options for the ordering of tiles are seen to be analogs of traditional map indexing systems and to fall into a general class of digit interleaving schemes. A general scheme for generating indexes is proposed. Optimality in tile arrangement is defined through an objective function in a form of the quadratic assignment problem. Solutions are described for a number of simple arrangements under assumptions about device behavior and the nature of applications.

INTRODUCTION

The term **tiling** is used in various ways in spatial data handling, but for the purposes of this paper we use it to refer to any system by which a database is partitioned geographically. The existence of such tiles or partitions may or may not be known to the user: in some systems the user manipulates tiles explicitly through a tile manager or map librarian, while other systems hide the management of tiles from the user and present the appearance of a seamless database.

We assume that some form of tiling is inevitable if physical constraints are not to be placed on the potential size of a spatial database. The acquisition of spatial data is limited only by time and cost, and it is often observed that expectations about the coverage and level of spatial resolution of geographical data increase at least as fast as our technical ability to service them. Tiling may be required because of the limited capacity of storage devices, or in order to optimize the efficiency of various algorithms and processes, or to optimize search and retrieval. The distinction between tiling and indexing is clearly somewhat blurred: the term tiling is associated with a physical partitioning of the database, and also implies that manipulation of tiles will occur at the level of the operating system, and that tiles may be distributed over different types of devices.

The traditional analog of tiling is the map sheet, which represents a partitioning of a database into physically separate units. There are parallels therefore between the systems of tile indexing to be discussed below, and the systems devised for indexing map sheets. Traditionally, the map sheets in a series share a common scale and size (ignoring variation due to the earth's curvature and the projection used), which is determined by the constraints of

the printing, distribution and storage technology of paper maps. On the other hand the number of objects represented on the map sheet is not constrained except indirectly by cartographic issues.

In the vector-based domain, the storage requirements of a tile are determined by the number of objects present, and are therefore highly variable even though tiles may be uniform in area. The same is true in the raster-based domain: although the number of pixels may be constant over a set of uniform-area tiles, run-length encoding or hierarchical subdivision will produce a variable volume of data in storage. We define a **fixed** tiling as one in which the area covered by the database is divided into tiles of equal area and shape (usually rectangles) and an **adaptable** tiling as one in which tile size is allowed to vary, probably by hierarchical subdivision, in order to achieve an approximately equal volume of data per tile.

Adaptable tiling clearly has advantages, since the volume of data in each tile can be set to the optimal volume for search, retrieval and other types of processing. Fixed tiling is generally suboptimal, but has advantages in cases where the volume of data changes through time, or is otherwise not known in advance. The costs of restructuring an adaptable tiling in response to new data, in order to maintain optimality, can be substantial.

We assume that in a GIS context it is necessary to store a number of layers or coverages, corresponding to different themes, for the area covered by the database. In the vector domain these coverages will be populated by different classes of objects, and in the raster domain they will consist of layers of pixels. Since a tile can contain any number of classes of objects or layers of pixels, it is clearly possible to reduce the size of a tile either by subdividing its geographical extent, or by subdividing the set of themes.

The purpose of the present paper is to review the concept of tiling from the perspective of optimality. The second section looks at methods of indexing tiles, as a preliminary to the subsequent discussion. The third section proposes a framework for optimization, with specific application to optical stores.

TILE INDEXING

The database of the Canada Geographic Information System (CGIS) (Tomlinson, Calkins and Marble 1976) contains layers of area objects which are partitioned geographically into rectangular tiles of fixed size, known in CGIS as frames. The system of tile indexing was devised by Morton (1966) to ensure that the relative positions of tiles in the database were as far as possible directly related to their relative locations in space. Goodchild and Grandfield (1983) and Mark and Goodchild (1986) proposed measures which could be used to determine the success of different orderings at achieving this objective, and used them to evaluate a number

of standard orders.

The Morton sequence for a square array of n by n tiles can be generated by a simple algorithm as follows. Number the rows of tiles from 0 to n-1, and express each row number to base 2. Similarly number the columns from 0 to n-1 and express each column number to base 2. The position of each tile in the Morton sequence can be obtained by interleaving the row and column bits, resulting in a binary number between 0 and n^2-1. More formally, let $\{r_1,r_2,\ldots,r_p\}$ denote the ordered set of binary digits forming the base 2 representation of the row number i, $0\le i<n$, $2^{p-1}<n\le 2^p$ and let $\{c_1,c_2,\ldots,c_p\}$ similarly represent the ordered set of binary digits forming the column number j, $0\le j<n$. Then the ordered set $\{r_1,c_1,r_2,c_2,\ldots,r_p,c_p\}$ is the binary representation of k, $0\le k<n^2$, the position of the tile (i,j) in the Morton sequence.

If the concept of bit interleaving is generalized to arbitrary bases, it turns out to be identical to many more traditional approaches to indexing tiles or map sheets. Assume as before that the row address of a tile is represented by an ordered set of p digits, but allow each digit s to have a corresponding base x_s. The array is no longer assumed to be square; let m denote the number of rows and n the number of columns. The column address is an ordered set of q digits, digit t having base y_t. We now assume that the bases have been chosen such that the number of rows is equal to the highest number defined by the set of bases:

$$\prod_s x_s = m, \quad \prod_t y_t = n \qquad (1)$$

Then the simple row by row sequence starting at row 0 column 0 is generated by setting p=1, q=1, $x_1=m$, $y_1=n$ and interleaving. Putting the column digit before the row digit will generate a column by column sequence.

A more elaborate example is provided by the GEOLOC geographical referencing system (Whitson and Sety 1987), which indexes every 100 acre parcel in the continental US. The first level of partition consists of 2 rows and 3 columns, each tile being 25 degrees of longitude by 13 degrees of latitude. These tiles are ordered row by row from 1 to 6. At the next level each tile is divided into 26 rows of one half degree latitude and 25 columns of one degree longitude, the area covered by one 1:100,000 USGS quad. Each of these subtiles is given a two-letter designation by concatenating the letter representing the row (a base 26 digit A through Z) with one representing the column (base 25, A through Y).

Each subtile is divided into 4 rows and 8 columns of 7.5' quads, numbered row by row from 1 to 32. At the next level these are divided into 4 rows and 2 columns, designated by assigning the letters A through H row by row. Finally, each of these divisions is divided into 5 rows, lettered A through E, and 10 columns numbered 0 through 9 to produce 50 cells of approximately 100 acres each. An example of a full

designator for a 100-acre parcel (in the Los Angeles area) is 4FG19DC6, or the set of 7 digits {4,F,G,19,D,C,6}, with associated bases {6,26,25,32,8,5,10}. Of these, digits 2, 3, 6 and 7 result from a simple interleaving of row and column digits (digits 2 and 6 from rows, 3 and 7 from columns). Digits 1, 4 and 5 are obtained by first concatenating row and column digits with bases x_s and y_t respectively and then reexpressing the result with base $x_s y_t$. So in full, the technique requires the interleaving of a set of 5 row address digits with bases {2,26,4,4,5} and a corresponding set of 5 column digits with bases {3,25,8,2,10}, and expressing the result as a set of 7 digits with bases {2×3,26,25,4×8,4×2,5,10}. Generalized digit-interleaving schemes such as these are common in the index systems used for numerous national map sheet series. In general, then, digit interleaving allows tiles to be placed in an order which approximates Morton's earlier objective. The GEOLOC ordering of 7.5' quads clearly comes closer to doing so than a system of ordering alphabetically by state, and alphabetically within state by quad name.

Further generalizations of the digit interleaving concept result when complement operations are allowed. Let the notation r^*_s indicate that the subsequent element in the ordered set is complemented, i.e. its value c_t is replaced by $y_t - c_t$, under certain conditions. For example, c_t might be complemented whenever r^*_s is odd. The example $\{r^*_1, c_1\}$ now generates a sequence in which each alternate row is reversed (boustrophedon, or the row prime order of Goodchild and Grandfield 1983). A more complex example of complementing operators generates the Hilbert Peano or Pi order.

OPTIMIZATION

Transition probabilities

We now introduce a new notation as the basis for a discussion of optimization. Let i,j denote a tile in the database, consisting of some collection of spatial information, which might be objects or pixels, for some geographical area i and theme subset j. A second such tile/theme combination is denoted by k,l. Now consider the likelihood of requiring both tiles i,j and k,l in some GIS process. For example we might wish to search both tiles for objects having specified attributes, or to display both tiles simultaneously, or to undertake the edgematch operation of matching objects across the common boundary of the two tiles. In a final example we might wish to change the current display from the contents of i,j to k,l.

Let $p(k,l|i,j)$ denote the probability that k,l is the next tile required after i,j, either to replace i,j or to be analyzed simultaneously with it. The transition from i,j to k,l may require change of geographical area, or theme, or both. We assume that change of area is independent of change of theme, in other words that the likelihood of moving from area i to area k is independent of the themes involved, and write $p(k,l|i,j)=p(k|i)p(l|j)$.

The set of themes included in any database is clearly limited, and it would be unreasonable to try to build a spatial database containing all possible themes. Instead the set of themes in a database is limited to those appropriate to the application set. But within a given database, the relationship between geographical subdivision and subdivision of themes is complex. In some systems, each tile includes all themes, requiring relatively small geographical subdivisions. In others, themes are split across several tiles, allowing the geographical area of each tile to be relatively large. CGIS is based on a hierarchical system in which one level of tiling, the 1:250,000 map sheet, contains all themes while a lower level, the frame, contains only one theme. In terms of our notation, if $p(j|j) >> p(l|j) \forall l \neq j$ then the rational strategy would be to maximize the geographical area of each tile and minimize the number of themes per tile; if the $p(l|j)$ are roughly equal then it is rational to store all themes together and reduce the geographical extent of each tile accordingly. For the set of themes included in the database we suspect that the second case is a more accurate reflection of the needs of most forms of GIS analysis and sets of users, and is the approach used in most of the systems currently available. However a hierarchical approach in which each tile is further subdivided into single themes, and then into geographical partitions of each theme, is commonly adopted in the interests of processing efficiency. If a tile contains all themes we can drop the $p(l|j)$ term and focus on $p(k|i)$, and additionally ignore $p(i|i)$.

It seems reasonable to assume that $p(k|i)$ is a decreasing function of some measure of the distance between k and i. One possibility is therefore to take $p(k|i)$ to be a constant if k and i are neighbors, and zero otherwise. For example, in a rectangular tiling we might take $p(k|i)=1/4$ for all rook's case neighbors of i. Another is to assume some suitable continuous function of the distance d_{ik} between the centroids of the tiles, such as the negative exponential $\exp(-bd_{ik})$. However in practice we may have access to some additional information which can provide a surrogate for $p(k|i)$. For example in a vehicle navigation system a suitable surrogate would be the probability of the route passing to tile k from tile i, which might be based on the existence of a freeway or on traffic statistics.

Retrieval costs

We assume that all tiles are located on some device, and that there is a cost associated with retrieval of a particular tile which depends on the previous tile retrieved or accessed. Let c_{ik} denote the cost or penalty of retrieving or accessing tile k given that the last tile accessed was i. In the case of a tape, c_{ik} will be dependent on the length of tape separating the tiles, whereas in the case of disk c_{ik} is approximately constant and independent of both i and k. For map sheets stored in cabinets in a map library, we might speculate that c_{ik} shows a complex behavior: low if i and k are adjacent, increasing rapidly with separation if they are almost adjacent, but high and

constant if i and k are separated by more than a few sheets. Certain sheets k are likely easy to find independently of i.

In this paper we are particularly concerned with massive stores with capacity in the gigabyte to terabyte range ($>10^9$ bytes). These include a variety of automatically loading tape systems, in which the individual reel of tape has a capacity on the order of 10^8 bytes. Of particular interest in this paper are massive optical stores (jukeboxes) which provide automatic loading of platters, each with a capacity of 2 gigabytes. Such stores currently provide the only feasible method for efficient storage and retrieval of spatial databases of the size of the USGS's Digital Cartographic Database or the Bureau of the Census's TIGER files.

Such stores are characterized by relatively slow seek times, when the store may be moving its optical read head, sequentially processing tape, or changing tape or platter, and fast bulk transfer rates. For tape stores, c_{ik} is an increasing function of separation on one volume, and roughly constant across volumes: for optical stores, c_{ik} is similarly an increasing function of separation within volume, although numerically much smaller, and high and constant between volumes.

We can now write the expected cost of accessing tile k following tile i as $c_{ik}p(k|i)$, and the expected cost of accessing any tile as its sum over k. The optimum tile arrangement on a given device is that which minimizes:

$$Z = \sum_i \sum_k c_{ik}\, p(k|i)\, N_i \qquad (2)$$

where N_i is the number of accesses of tile i. The problem of minimizing Z falls into the general class of quadratic assignment problems (Koopmans and Beckmann 1957). In the commonest interpretation c_{ik} is the cost of moving a unit quantity of material between two machines i and k located on a shop floor, $p(k|i)N_i$ is the flow of material between the two machines, and the objective is to locate machines to minimize the total cost of movement.

Although a large literature exists on exact and heuristic solutions to the quadratic assignment problem (Francis and White 1974), we require general solutions which are robust across as wide a set of applications as possible. In the next section we consider the effects of some likely simplifying assumptions.

General solutions

We first assume N_i constant; it would be very difficult to assemble the necessary information on which any other value of N_i might be based, and it is unlikely that the result would be robust across the application set of any given database. We further assume $p(k|i)=1/4$ for all k which are rook's case neighbors of i, otherwise $p(k|i)=0$. Again, such a simple assumption has the advantage that it is likely to be robust across applications.

First let us assume that $c_{ik}=a$ if i and k are adjacent in storage, else $c_{ik}=b$, $b>>a$. This leads to a simple solution in which the optimum value of the objective function Z occurs whenever tiles which are neighbors in storage are also neighbors in space. Many arrangements have this property, including the row prime and Hilbert Peano orders, resulting in an objective function value of (a+b)/2 per tile, since every tile incurs a cost of a for each of two of its neighbors and b for the other two. On the other hand the expected cost of the Morton order is (a+3b)/4 since only one of a tile's four neighbors is adjacent in the Morton sequence. For row order the cost is $[(n-1)a+(n+1)b]/2n$ where n is the number of columns because the tiles at the end of each row have only one adjacent neighbor and thus incur additional cost.

Now assume that tape volumes are used for storage, and that the number of tiles on each volume is such that all of a tile's neighbors are stored on the same volume. It seems reasonable to assume for tape that c_{ik} is a linear function of the separation of the tiles on the tape, that is:

$$c_{ik} = \alpha + \beta \, |z_i - z_k| \qquad (3)$$

where z_i is the location of tile i with respect to the beginning of the tape and α and β are constants. Goodchild and Grandfield (1983) show that for row by row, row prime and Morton orderings of an array of n rows and n columns, the mean absolute difference between a cell's position in the sequence and those of its rook's case neighbors is (n+1)/2. In all of these orderings cost is therefore $n^2[\alpha+\beta(n+1)/2]$. Goodchild and Grandfield were unable to obtain a closed-form expression for Hilbert Peano order but their numerical expression gives slightly higher cost. We conclude that Morton order has no advantage over row by row order in this example.

Although we have not been able to prove the general case, we have thus far failed to find a counterexample to disprove the proposition that any ordering of a square array which can be generated by interleaving of digits (including row by row and Morton) has the same mean absolute difference between neighbors of (n+1)/2. The proposition is not true if $n \neq m$: for row by row order the result in this case is $[m(n-1)+n^2(m-1)]/(2nm-m-n)$. The question of which order minimizes Z is therefore unresolved in the general rectangular case.

DISCUSSION

The problem of optimal sizing and arrangement of tiles will become increasingly important in the future as spatial databases grow in size. In this paper we have considered one aspect of the arrangement problem, by making certain simplifying assumptions about the objective to be optimized. The questions of optimal size, and of the optimal balance between geographical and thematic subdivision, for different storage devices and applications, remain open.

Clearly it would be easier to define precise objective functions if more information were available on algorithms and patterns of use. On the other hand solutions developed in the absence of such information are necessarily more robust and general. General solutions are likely to be of considerable value in deciding between different storage options for very large databases, as well as in resolving the more specific questions of tile size and arrangement.

REFERENCES

Francis, R.L. and J.A. White, 1974, Facility Layout and Location - An Analytical Approach, Prentice Hall, Englewood Cliffs, NJ.

Goodchild, M.F. and A.W. Grandfield, 1983, Optimizing Raster Storage: An Examination of Four Alternatives: Proceedings, AutoCarto 6, Ottawa, 1:400-7.

Koopmans, T.C. and M. Beckmann, 1957, Assignment Problems and the Location of Economic Activity: Econometrica 25:53-76.

Mark, D.M. and M.F. Goodchild, 1986, On the Ordering of Two-Dimensional Space: Introduction and Relation to Tesseral Principles: in B. Diaz and S. Bell, editors, Spatial Data Processing Using Tesseral Methods, National Environmental Research Council, Reading, UK, 179-92.

Morton, G.M., 1966, A Computer Oriented Geodetic Data Base, and a New Technique in File Sequencing, unpublished manuscript, IBM Canada Ltd.

Tomlinson, R.F., H.W. Calkins and D.F. Marble, 1976, Computer Handling of Geographical Data, UNESCO, Paris.

Whitson, J. and M. Sety, 1987, GEOLOC - Geographic Location System: Fire Management Notes 46:30-32.