データベースシステム　126-4
（２００２．１．２１）

地理情報の効率的な格納法に関する検討

浅野貴裕[1]・白井靖人[2]

[1] 静岡大学大学院情報学研究科

[2] 静岡大学情報学部情報科学科

地理情報は一般にデータ量が多く、検索に多くの時間を要する。地理情報を扱うアプリケーションを考えると、緯度経度によって座標範囲を指定した検索や、付加情報の分類を指定した検索が頻繁に行われる。そこで本研究では、これらの検索を効率よく行える格納法について検討した。地図上の複数の地点に対して文字情報を付加する場合を想定し、(I)一定の大きさの領域毎に付加情報を管理し、(II)その領域毎にインデックスを付けそこに含まれる情報の分類を記録する、という方法をとる。この格納法について述べ、評価実験により有用性を示す。

Efficient Storage of Geographic Information

Takahiro ASANO [1], Yasuto SHIRAI [2]

[1] Graduate School of Information, Shizuoka University

[2] Department of Computer Science, Shizuoka University

Geographic information consists of a large amount of data, and a search through it tends to take much time. In a geometric application, however, a need often arises for a search by the region or a search by the category associated with each data item. Here we consider a method of storing geographic data which lends itself to the efficient retrieval of the stored data. In particular, a set of character data associated with points is taken as a sample data set. We propose a method of (I)managing the point-associated data in terms of the unit area of a certain extent, and (II)adding an index to each unit area categorizing the type of data residing in that area. Feasibility of the proposed method is evaluated through experimentation.

1. はじめに

地理情報は一般にデータ量が多く、それをコンピュータで扱う場合、検索に多くの時間を要する。地図データを付加情報と共に表示するアプリケーションでは、画面上に表示される情報が多くなるほど、それぞれを記憶装置から取り出すのにかかる時間が大きな問題となる。

本研究では、地図上の複数の地点に対し文字情報を付加することを想定している。地理情報の参照は領域を指定する場合が多い。つまり、これは緯度や経度といった座標を指定した参照である。また、情報の分類を指定し、特定の分類の情報のみを抜き出すといった参照も多く行われる。例えば、静岡県のすべての山名、浜名湖付近の観光施設、といった参照である。そこでこれら二つの条件、座標領域、分類番号を指定した検索に的を絞り、それを効率よく行えるデータ格納法について研究を行った。

目的の情報を素早く取り出すために、インデックスを用いるのは一般的な方法である。しかし、情報の件数が膨大な地理情報では、すべての情報に対してインデックス付けを行うことは現実的ではない。そこで、何らかの規則に従って複数の情報に対して一つのインデックスを付ける必要があり、地理情報では領域ごとにインデックスを付けるのが適している。本研究では、地図をある大きさの単位領域の集まりとして考え、インデックス付けを行う。これにより座標範囲を指定した検索の効率を高める。さらに、単位領域内にどの分類の情報が含まれるかを示すビット列をインデックスに記録することにより、分類を指定した検索の効率を高めるのが狙いである。

2. 格納法

提案する格納法では、詳細データファイルとインデックスファイルの二つのファイルを用いる。

詳細データファイルには付加情報本体を格納する。そして、地図を緯度経度の各方向で等間隔に区切った単位領域を考え、領域毎にインデックス付けを行う(図１)。これをインデックスファイルとして記録する。

評価実験で用いたデータでは北緯 20～50 度、東経 120～150 度の範囲において、各方向に 150 分割している。この結果、およそ 20km 四方の領域が 22,500 個得られる。

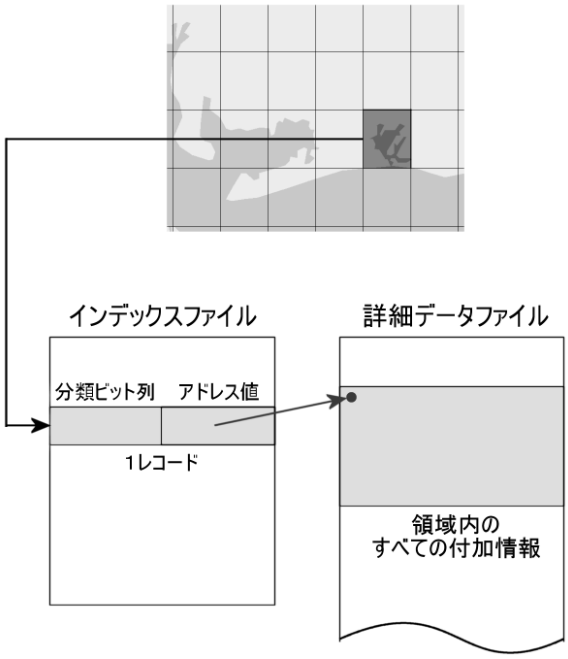

図 1. 格納法のイメージ

2.1. 詳細データファイル

「詳細データファイル」には、付加情報本体を格納する。文字情報を付加する場合にユーザーから入力されるのは、分類番号（情報の分類を表す。地名、山名、公共施設…）、座標値（緯度および経度）、文字列（付加する文字情報）である。最低限必要なこれらに、管理用として区切り文字、文字列長を加え、一つのレコードとする（図 2）。

区切り文字はファイル中でのレコードの区切りを表す。緯度、経度は固定小数点数で表す。

例えば北緯 34 度 43 分 17 秒であれば、34.4317 という数値として緯度のフィールドに格納する。文字列長フィールドには 2 バイトが割り当てられ、格納できる文字列は最大 65,535 文字である。

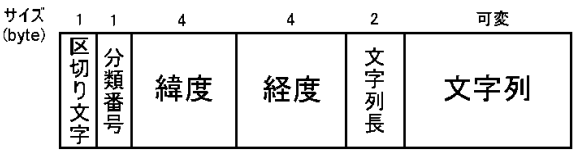

図 2. 詳細データファイルのレコード構造

本格納法では、座標範囲を指定した検索を早めるため、単位領域毎にインデックスが付けられ、同じ領域に関するレコードがすべて一続きになるように並べていく。

2.2. インデックスファイル

インデックスファイルには、詳細データファイルのインデックス情報を格納する。単位領域一つにレコード一つが対応しており、その領域にどの分類の情報があるかを示す分類ビット列と詳細データファイルにおける各領域の先頭レコードのアドレスを格納する(図 3)。分類ビット列には 32 ビットを割り当て、32 個の分類のそれぞれについて 1 と 0 で存在するかどうかを表す。領域内に同じ分類の情報が複数あることが考えられるが、あくまで存在するかどうかを示し、その個数についての情報は含まない。

図 3. インデックスファイルのレコード

インデックスファイルのレコードは、対応する単位領域が南西端で始まり北東端で終わるように並べる。また、付加情報が含まれていない領域についてもレコードは用意される。さらに、レコードは固定長であるため、参照したい単位領域が判明すれば、それに対応するインデックスファイルのレコードのアドレスを容易に算出可能である。

単位領域の総数は 22,500、一つの領域に対するインデックスのレコードサイズは 8 バイトであるので、インデックスファイルのサイズは 22,500×8 バイト= 180,000 バイトとなる。

3. 検索プログラムの動作

ここでは本格納法における検索プログラムの動作と、どのようにして検索効率を高めているかを述べる。

検索条件として指定されるのは座標範囲と分類番号である。座標範囲が指定された場合には、インデックスファイル中の参照すべきレコードを絞り込む。分類番号が指定された場合には、詳細データファイル中の参照すべきレコードを絞り込む。これらの絞込みによって検索効率の向上を図る。検索プログラムの大まかな流れを図 4 に示す。

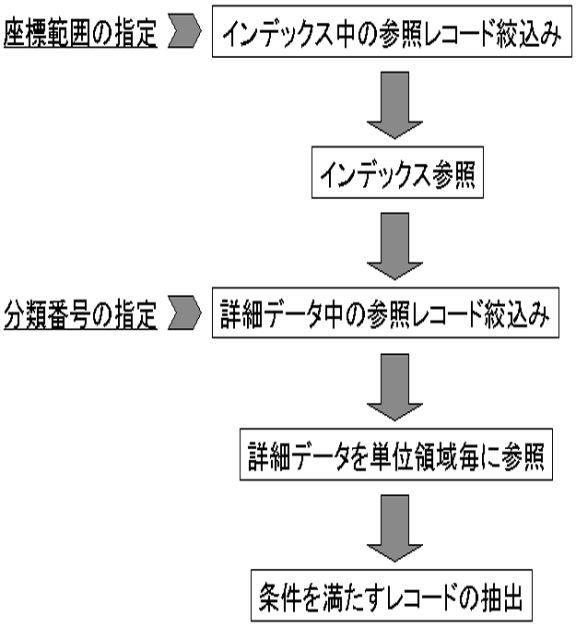

図 4. 検索プログラムの流れ

3.1. 座標範囲指定による検索

座標範囲が指定されると、その値を基にして参照すべきインデックスのレコードを絞り込む。そして、そのレコードに格納されている詳細データファイル上のアドレス値を読み込み、単位領域毎に参照する(図 5)。インデックスのレコードは規則的に並んでおりサイズが固定長のため、アドレスは容易に算出可能である。

検索プログラムは、単位領域内のすべての情報を参照し、その座標値と検索条件を比べながら、条件に合う情報のみを抽出する。詳細データファイルは単位領域毎にまとめて参照するため、指定された座標範囲が単位領域と完全に重ならない場合には、検索条件に該当しない情報も含まれる。

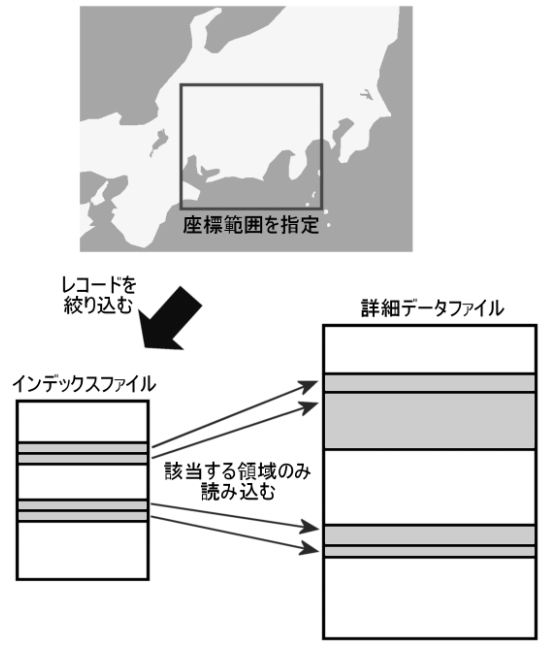

図 5. 座標範囲を指定した検索

3.2. 分類番号指定による検索

分類番号を指定した検索では、インデックスファイルの分類ビット列を参照し、対応する単位領域に指定された分類の情報が含まれるかを調べる。そして、含まれると判明した場合のみ、詳細データファイル中の対応する単位領域のレコードにアクセスする(図 6)。

分類ビット列は、ある分類の情報が含まれているかどうかのみを表し、その個数に関するデータは含まない。よって、単位領域内のレコードはすべて参照する必要がある。

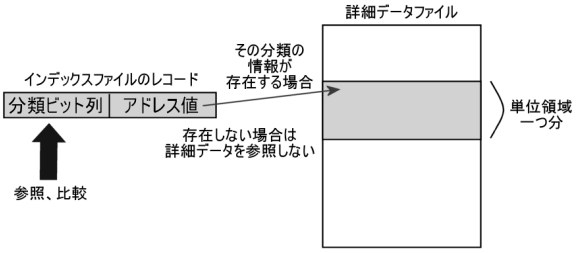

図 6. 分類番号を指定した検索

4. 評価実験

提案する格納法を用いた評価実験を行った。用意した文字情報は、全国の地名、山名、公共施設名などからなるおよそ 3,000 件のデータである[5]。これより多数のデータが必要な場合には複製して利用した。また、実験は(方式 1)インデックスを持たないもの、(方式 2)HDD 上に置くもの、(方式 3)メインメモリ上に置くもの、とで比較を行った(表 1)。

検索条件は表 2 のように設定した。条件 A～C は座標範囲を指定している。A が最も範囲が広く、C が最も狭い。D、E は分類番号を指定している。D は全情報の中で最も件数の多い分類のものを事前に調べておき、その分類を指定する。E では逆に、最も件数の少ない分類を指定する。F は、座標範囲と分類番号を同時に指定した検索である。狭い範囲で、かつ、件数の少ない分類の情報を対象としているため、最も検索効率が高まるはずである。

情報の総件数が 100、1,000、10,000、100,000 の各場合において、前述の条件によって検索を行い、所要時間を計測した。結果を表 3～6 に示す。

表 1 比較用の格納方式

方式１	詳細データファイルのみ (=インデックスを持たない)
方式２	提案する格納方式 (インデックスを HDD に置く)
方式３	提案する格納方式 (インデックスをメインメモリに置く)

表 2 検索条件

A	座標範囲	すべての範囲
B	座標範囲	A の 1 / 9 の範囲
C	座標範囲	A の 1 / 900 の範囲
D	分類番号	最も件数の多い分類
E	分類番号	最も件数の少ない分類
F	座標範囲と 分類番号	C かつ E

表 3 実験結果 総件数 100 の場合

検索条件	該当件数	該当領域数	平均検索時間(秒)		
			方式1	方式2	方式3
A	100	22,500	0.015	0.552	0.125
B	46	2,704	0.016	0.253	0.022
C	1	49	0.009	0.209	0.003
D	100	68	0.009	0.656	0.000
E	0	0	0.013	0.681	0.000
F	0	0	0.016	0.209	0.000

表 4 実験結果 総件数 100,000 の場合

検索条件	該当件数	該当領域数	平均検索時間(秒)		
			方式1	方式2	方式3
A	1,000	22,500	0.172	0.725	0.281
B	629	2,704	0.165	0.381	0.141
C	7	49	0.166	0.215	0.013
D	62	41	0.169	0.653	0.000
E	0	0	0.171	0.675	0.000
F	0	0	0.160	0.206	0.000

表 5 実験結果 総件数 10,000 の場合

検索条件	該当件数	該当領域数	平均検索時間(秒)		
			方式1	方式2	方式3
A	10,000	22,500	1.732	2.499	2.031
B	8,325	2,704	1.725	1.859	1.653
C	955	49	1.722	0.481	0.294
D	4,464	55	1.755	1.541	0.919
E	3	1	1.753	0.678	0.016
F	3	1	1.722	0.218	0.022

表 6 実験結果 総件数 100,000 の場合

検索条件	該当件数	該当領域数	平均検索時間(秒)		
			方式1	方式2	方式3
A	100,000	22,500	17.254	18.690	18.414
B	83,250	2,704	17.290	15.969	15.865
C	9,550	49	17.234	2.981	2.778
D	44,640	55	17.619	9.306	8.838
E	30	1	17.740	0.841	0.184
F	30	1	17.128	0.397	0.181

表中の網掛けの部分は、同じ検索条件において方式１に比べ検索時間が短い場合である。該当件数、該当領域数とは、検索条件に合う情報の数とそれが含まれている単位領域の数である。

情報の総件数が 100 と 1,000 の場合に、方式 2 はすべての検索条件で方式 1 よりも時間がかかっている。これは、情報の件数が少ない分、処理時間中のインデックスを参照する時間の割合が大きいためと考えられる。また、検索条件Ａはすべての情報を指定するものであり、本格納法では、インデックスファイルを参照する分だけ余分に時間がかかる。

すべての場合において言えるのは、本格納法では条件が絞られるほどに検索効率が上がっているということである。特に、全体の情報の件数が多くなるほど、より多くの条件下で効果が現れている。

5. おわりに

評価実験より、本研究で提案した格納法は情報の件数が多い場合には有効であることが確認できた。逆に件数が少ない場合には、インデックスの参照がかえって検索効率を下げることもあった。これについてはインデックスをメインメモリ上に置くことである程度改善できた。当初の目的が、膨大な地理情報の中からいかに早く目的の情報を取り出すかということであるので、本格納法は有用であると言える。

今回は、座標範囲と分類番号を指定した検索に的を絞り、単位領域毎のインデックス付けと分類ビット列の格納という方法を取った。しかしこの方法では、単位領域内のすべての情報を参照しなければならず、改善の余地はある。

今後の課題としては、領域内の情報件数のインデックスへの記録、情報の密度に合わせた領域の大きさの変更、文字列以外のデータの格納、地理情報アプリケーションとの連携、等が挙げられる。

参考文献

文献[1]～[4]は、本研究全般に関わるものであり、本文中には参照個所を明示しない。

[1] Michael J.Folk, Bill Zoellick (著), 楠本博之, 浜名祐一 (共訳) :「ファイル構造」, 1997, 共立出版株式会社

[2] ナビゲーションシステム研究会 :「ナビ研ソフト作成ガイドブック S 規格 (Version 2.2)」, 1997, http://www.naviken.co.jp

[3] 建設省国土地理院監修 :「数値地図ユーザーズガイド(改訂版)」, 1992, (財)日本地図センター

[4] 国土地理院ホームページ, http://www.gsi.go.jp

[5] カシミール 3D/風景 CG と地図と GPS のページ, http://www.kashmir3d.com/