

Arbeitsgruppe IV/1

Helmut Schumacher
Klein & Stekl GmbH
7000 Stuttgart

EINE METHODE ZUR EINHEITLICHEN SPEICHERUNG UND VERAR-
BEITUNG GEOGRAPHISCHER FLÄCHENDATEN FÜR DIE REGIONAL-
PLANUNG

Zusammenfassung

Es wird über eine Methode berichtet, beliebige Flächen-
daten (z.B. Gemeindeflächen) und Klassifizierungsdaten
einheitlich und effizient zu speichern.

Die Daten werden in einem festen Koordinatensystem als
eine Hierarchie von Rastern gespeichert. Damit sind Such-
und Verknüpfungsoperationen (wie etwa die Bestimmung der
Flächennutzung einer Gemeinde) einfach und schnell durch-
zuführen.

Der Bericht schließt mit Erfahrungen, die bei der Imple-
mentierung der Methode auf einem Großrechner gemacht
wurden. Rechenzeit und Speicherbedarf bei einer Anwendung
in der Regionalplanung werden abgeschätzt.

1. Einleitung

In der Regionalplanung müssen bei der Flächenbilanzierung
Klassifizierungsdaten und "Flächendaten" (wie z.B. Gemeindeflächen) verknüpft werden. Während Klassifizierungsdaten
gewöhnlich in herkömmlicher Weise gespeichert werden (Bild-
elemente werden in Zeilenstruktur angeordnet), werden
Flächendaten oft in Form von Polygonzügen festgehalten.
Eine Abspeicherung der Flächendaten in Zeilenstruktur ist
sehr speicheraufwendig, eine Erzeugung der Zeilenstruktur
aus den Polygonzügen zeitaufwendig.

Daher wird hier über eine einheitliche Datenstruktur ("Rasterstruktur") berichtet, bei der Auswertungen wie z.B. Flächenbilanzierungen direkt, also ohne Konversionen, vorgenommen werden können.

Die hervorstechenden Eigenschaften der Rasterstruktur sind:

- geringer Speicherbedarf: dieser korreliert stark mit der Komplexität der Flächen, jedoch kaum mit ihrer absoluten Größe;
- Einfachheit der Verknüpfungen: den Mengenoperationen der Flächen entsprechen Boolesche Operationen in der Rasterstruktur; die Operationen können rekursiv durchgeführt werden;
- Schnelligkeit der Verknüpfungen: die Operationen innerhalb der Rasterstruktur sind schnell, während die Kosten für die Integration der Flächendaten relativ hoch sind.

Die vorzustellende Methode ist im Rahmen einer Projektstudie bei der Deutschen Forschungs- und Versuchsanstalt für Luft- und Raumfahrt (DFVLR) in Oberpfaffenhofen auf einem Großrechner Amdahl 470/V6 in PL/1 implementiert worden, weshalb Erfahrungswerte für Speicher- und Rechenzeitbedarf vorliegen, auf die in Kap. 4 eingegangen wird.

2. Datenstruktur

Bei der Erstellung einer gemeinsamen Datenbasis für Flächen- und Klassifizierungsdaten müssen zuerst

- ein Gesamtgebiet, innerhalb dessen alle Daten liegen, sowie
- eine gemeinsame, feste Auflösung (z.B. 50x50 m je Bildelement)

festgelegt werden.

Entsprechend der gewählten Auflösung wird das Gesamtgebiet mit einem Gitter von Rasterelementen überzogen, wobei jedes Rasterelement einem Bildelement in der gewählten Auflösung entspricht. Jedes Rasterelement ist durch ein Koordinatenpaar in der Lage bestimmt.

Flächendaten werden in der Weise abgespeichert, daß zu jedem Rasterelement angegeben wird, ob es zur Fläche gehört oder nicht (entsprechend der charakteristischen Funktion bei Mengen).

Klassifizierungsdaten werden als Vereinigung von Flächen-
daten interpretiert, wobei jede Klasse durch eine mehrfach
zusammenhängende Fläche repräsentiert wird.

Selbstverständlich verbietet sich ein explizites Abspeichern dieser 1-Bit-Informationen für alle möglichen Rasterelemente. Deshalb wird davon Gebrauch gemacht, daß benachbarte Rasterelemente mit hoher Wahrscheinlichkeit entweder beide zur Fläche oder beide nicht zur Fläche gehören. Dazu wird eine Aufteilung des Gesamtgebietes in eine Hierarchie von Rasterebenen vorgenommen. Alle Raster einer jeden Stufe überdecken das Gesamtgebiet vollständig.

Jedes Raster einer Stufe überdeckt genau eine feste Anzahl von Rastern niedrigerer Stufe.

Für eine zu speichernde Fläche kann zu jedem Raster $R(k)$ einer Stufe k angegeben werden, ob (i), (ii) oder (iii) zutrifft:

- (i) kein Rasterelement, das von $R(k)$ überdeckt wird, ist Teil der Fläche;
- (ii) alle Rasterelemente, die von $R(k)$ überdeckt werden, sind Teil der Fläche;
- (iii) einige, jedoch nicht alle Rasterelemente sind Teil der Fläche.

Lediglich im Falle (iii) ist es notwendig, weitere Informationen über die Rasterelemente, die von $R(k)$ überdeckt werden, abzuspeichern. Dazu werden alle Raster der Stufe $k-1$, die von $R(k)$ überdeckt werden, auf dieselbe Weise untersucht.

Dieser Prozeß endet spätestens, wenn Raster der Stufe 0 untersucht werden. Bei diesen kommt nur (i) oder (ii) in Frage.

Die Speicherung einer Fläche gleicht daher einem Baum, wobei die Wurzel des Baumes dem Gesamtgebiet entspricht. Die Knoten des Baumes sind Raster einer bestimmten Stufe, wobei Raster derselben Stufe auf derselben Ebene des Baumes liegen.

Die Blätter des Baumes, d.h. die Knoten, die sich nicht verzweigen, sind Raster, die in Kategorie (i) oder (ii) fallen. Knoten der Kategorie (iii) verzweigen in Stufe k in Knoten der Stufe $k-1$.

Die Raster werden z.B. als Quadrate ausgelegt, obwohl dies nicht zwingend ist.

Die Festlegung der Größe der Raster und damit die Anzahl der Rasterstufen ist eine wichtige Entscheidung, die beim Entwurf der Datenbasis zu treffen ist. Es besteht ein "Trade-Off" zwischen der Anzahl der Hierarchie-Stufen und der Größe der Raster. Eine geringe Hierarchietiefe und ein dementsprechend großer Unterschied in der Rastergröße beim Übergang zwischen den Stufen bedeutet, daß - vor allem bei kleinen Flächen - in der Regel ein höherer Speicheraufwand dadurch erforderlich ist, daß in den einzelnen Knoten relativ weniger Information gespeichert ist. Eine größere Anzahl von Hierarchiestufen auf der anderen Seite bedeutet, daß der "Overhead" für die notwendige Kontrollinformation (Zeiger) in den Bäumen zunimmt.

3. Operationen mit der Rasterstruktur

Für eine mögliche Anwendung der Rasterstruktur in der Regionalplanung sind eine Reihe von Verknüpfungsoperationen notwendig. Neben der Konversion von Daten in Zeilenstruktur zur Rasterstruktur und umgekehrt sind erforderlich:

- UND-Verknüpfung zweier Rasterflächen;
- ODER-Verknüpfung zweier Rasterflächen;
- Komplement einer Rasterfläche im Gesamtgebiet.

Die UND-Verknüpfung wird z.B. bei der Flächenbilanzierung einer Gemeindefläche angewandt. Für alle gespeicherten Klassen werden Gemeindeflächen und Klassifizierungsraster konjunktiv verknüpft. Die Anzahl der so erhaltenen Raster-elemente der Stufe 0 ist ein Maß für die Fläche einer Klasse innerhalb der Gemeindefläche.

Die ODER-Verknüpfung wird z.B. benötigt, um Auswertungen über die Vereinigung von Flächen durchzuführen.

UND, ODER und Komplementbildung werden beim Ändern von Klassifizierungsdaten benötigt. Sollen für eine bestimmte Fläche F die eingespeicherten Klassifizierungsdaten ersetzt werden, so wird zunächst die komplementäre Fläche G von F im Gesamtgebiet gebildet. Durch konjunktive Verknüpfung von G mit allen Klassenrastern werden die alten Klassifizierungsdaten innerhalb F ausgeblendet. Durch eine folgende Verknüpfung der so verbliebenen Klassifizierungsdaten in G mit den neuen Daten für F werden so die neuen Klassifizierungsdaten für das Gesamtgebiet erhalten.

Die Konversion von Daten in Zeilenstruktur zur Rasterstruktur und umgekehrt sind relativ aufwendig. Demgegenüber sind UND- und ODER-Verknüpfung sehr schnell. Dies rührt daher, daß bei einer Verknüpfung nicht jedes Rasterelement der Stufe 0 herangezogen werden muß.

Infolge der hierarchischen Rasterstruktur können die die Raster verarbeitenden Operationen rekursiv formuliert werden.

UND-Verknüpfung von R(k) mit S(k) zum Ergebnisraster T(k):

falls R(k)="alle" und S(k)="alle" dann T(k)="alle" ;
falls R(k)="keine" oder S(k)="keine" dann T(k)="keine" ;
in den anderen Fällen ist es notwendig, Raster der Stufe k-1 zu untersuchen und in Stufe k fortzufahren, nachdem das Ergebnis für T(k) feststeht.

ODER-Verknüpfung von R(k) mit S(k) zum Ergebnisraster T(k):

falls R(k)="alle" oder S(k)="alle" dann T(k)="alle" ;
falls R(k)="keine" dann T(k)=S(k) ;
falls S(k)="keine" dann T(k)=R(k) ;
in den anderen Fällen ist es notwendig, Raster der Stufe k-1 zu untersuchen und in Stufe k fortzufahren, nachdem diese Ergebnisse vorliegen.

4. Erfahrungen

Bei dem Testsystem, das bei der DFVLR in Oberpfaffenhofen erstellt wurde, war das Gesamtgebiet auf etwa 100x100 km und die Auflösung auf 50x50 m festgelegt. Die Rasterhierarchie war wie folgt gewählt:

R(0): 1 Bildelement =50x50m
R(1): 8x8 R(0) =400x400m
R(2): 4x4 R(1) =1,6kmx1,6km
R(3): 4x4 R(2) =6,4kmx6,4km

Würde das Gesamtgebiet auf ein größeres Gesamtgebiet, etwa die Bundesrepublik ausgedehnt werden (bei der gewählten Auflösung sind dies etwa 100 Mio Rasterelemente), so sollte die Anzahl der Hierarchiestufen auf wenigstens 4 oder 5 erhöht werden. Wegen der "logarithmischen" Struktur der Daten steigt der Gesamtbedarf an Speicher und Rechenzeit beim

Übergang zu einem größeren Gesamtgebiet nur wenig mehr als linear an. Aufgrund dieser Überlegungen und der gemachten Erfahrungen können die folgenden Abschätzungen für Speicher- und Rechenzeitbedarf gegeben werden.

Speicherbedarf:

Bei LANDSAT-Klassifizierungsdaten ergab sich, daß man bei 10 Klassen mit einem Speicherbedarf von etwa 0.5 Bytes pro Bildelement für die gesamten Klassifizierungsdaten rechnen kann. Dies bedeutet, daß man bei einer Auflösung von 50x50m für die Klassifizierungsdaten der gesamten Bundesrepublik mit größenordnungsmäßig 50MB auskommt.

Für kleinere und mittlere Gemeindeflächen ergab sich bei dieser Auflösung (4 Rasterelemente pro Hektar) ein Speicherbedarf von 0.24 bis 0.8 Bytes pro Hektar, bei großen Gemeinden und Kreisen etwa 0.08 bis 0.12 Bytes pro Hektar. Dies bedeutet, daß für Gemeinde- plus Kreisflächen im Schnitt etwa 1 Byte pro Hektar ausreicht. Für die gesamte Bundesrepublik entspricht dies etwa 25MB.

Rechenzeitbedarf:

Beim Integrieren von Klassifizierungsdaten wie von Gemeindeflächen ergab sich eine CPU-Zeit von ca. 0.4msec pro Rasterelement. Bei der erwähnten Auflösung bedeutet dies, daß für die Integration von Klassifizierungsdaten sowie von Gemeindeflächen für die gesamte Bundesrepublik jeweils etwa 12 Stunden CPU-Zeit benötigt werden.

Bei der Flächenbilanzierung von Gemeindeflächen anhand eingespeicherter Klassifizierungsdaten kann im Schnitt von 0.5 bis 1 sec CPU-Zeit ausgegangen werden.