Concepts of Map Algebra

Unit: VI

Semester: II

Paper Code: GIS 07
Name of Paper: Geographic Information System

PG Diploma in RS & GIS

Dr. SHYAMA PRASAD MUKHERJEE UNIVERSITY, RANCHI

1.1 Concepts of Map Algebra

Map algebra is an informal and commonly used scheme for manipulating continuously sampled (i.e. raster) variables defined over a common area. It is also a term used to describe calculations within and between GIS data layers, according to some mathematical expression, to produce a new layer; it was first described and developed by Tomlin (1990). Map algebra can also be used to manipulate vector map layers, sometimes resulting in the production of a raster output. Although no new capabilities are brought to GIS, map algebra provides an elegant way to describe operations on GIS datasets. It can be thought of simply as algebra applied to spatial data which, in the case of raster data, are facilitated by the fact that a raster is a georeferenced numerical array.

Map Algebra models the surface of the earth as a multitude of independent, coincident layers or themes. The layers interact according to mathematical models and are typically based on real world observations. Planners develop layers on development and population (Steinitz et al. 1976). Social scientists develop layers on demographics, ethnicity, and economic factors (McHarg 1969). Applying Map Algebra model to input layers produces a new layer, which may be a physical map sheet, a vision perceived through a stack of mylars on a light table, or an electronic dataset displayed on a computer screen. Regardless of mechanism, the result allows its users to explain complex phenomena, predict trends, or make adjustments to the model.

However, it is the mechanism which bounds usability of Map Algebra. How easy it is for scientists to perform simple tasks? Can complex models be developed and tested? Historically layers were plotted on individual transparent maps which, when superimposed and registered provide a visually integrated view of the data. The manual process of map overlay is slow and tedious.

1.1.1 Data types

This section focuses on map algebra operations available for gridded datasets, such as those implemented in ESRI's Spatial Analyst extension for ArcView. Similar operations are available in other grid-based GISes, such as GRASS (Geographic Resources Analysis Support System).

The data associated with any grid cell can be of any type whatsoever. It is conceptually useful to divide data types into several classes, however. These include:

- Categorical data: These are non-numerical data. Grids that classify land use or land cover exemplify this category. Other examples are **proximity grids** (values identify the nearest object) and **feature grids** (only two values are possible: one value for cells where features occur, another value--typically zero or NoData--where features do not occur).
- Integral data: These data may be relative **ranks** or preferences or they may be counts of occurrences or observations, for example. Thus, what they measure is inherently integral.
- Floating-point ("real" data). These typically represent a real surface, such as elevation, or the values of a scalar function (a "conceptual surface," if you will). Examples of such functions would be temperature, slope, amount of sunlight received per year, distance to the nearest feature, population density.
- Vector data: These are ordered tuples of real values that represent fields of directions. For example, hydraulic gradients (for two-dimensional groundwater models), wind velocities (again for two-dimensional models), and ocean currents are two-dimensional vector fields. Vector data may have more than two dimensions, even though they are defined over a strictly two-dimensional domain. For example, models using astronomical data, such as climate models, may make use of information about the three-dimensional location (on the earth's surface) of each grid point.

(Scientific visualization systems usually have built-in support for vector data, whereas most GISes require the modeler to represent vector data as an ordered collection of floating-point grids).

1.1.2 Working with null data

An essential part of map algebra or spatial analysis is the coding of data in such a way as to eliminate certain areas from further contribution to the analysis. For instance, if the existence of low-grade land is a prerequisite for a site selection procedure, we then need to produce a layer in which areas of low-grade land are coded distinctively so that all other areas can be removed. One possibility is to set the areas of low-grade land to a value of 1 and the remaining areas to 0. Any processes involving multiplication, division or geometric mean that encounter the zero value will then also return a zero value and that location (pixel) will be removed from the analysis. The opposite is true if processing involves addition, subtraction or arithmetic mean calculations, since the zero value will survive through to the end of the process. The second possibility is to use a null or No Data value instead of a zero. The null is a special value which indicates that there is no digital numerical value. In general, unlike zero, any expression will produce a null value if any of the corresponding input pixels have null values. Many functions and expressions simply ignore null values, however, and in some circumstances this may be useful, but it also means that a special kind of function must be used if we need to test for the presence of (or to assign) null values in a dataset. For instance, within ESRI's ArcGIS, the function ISNULL is used to test for the existence of null values and will produce a value of 1 if null, or 0 if not. Using ER Mapper's formula editor, null values can easily be assigned, set to other values, made visible or hidden. Situations where the presence of nulls is disadvantageous include instances where there are unknown gaps in the dataset, perhaps produced by measurement error or failure. Within map algebra, however, the null value can be used to great advantage since it enables the selective removal or retention of values and locations during analysis.

Table 1.1 Operations categorized according to their spatial or non-spatial nature.

Output	**Spatial attributes involved?**	
	Yes	**No (not necessarily)**
Map or image	Neighbourhood processing(filtering), zonal and focal operations, mathematical morphology	Reclassification, rescaling (unary operations), overlay (binary operations), thresholding and density slicing.
Tabular	Spatial autocorrelation and variograms	Various tabular statistics (aggregation, variety) and tabular modeling (calculation of new fields from existing ones), scattergraphs

(Source: After Bonham-Carter, 2002)

1.1.3 Logical and conditional processing

These two processes are quite similar and they provide a means of controlling what happens during some function. They allow us to evaluate some criterion and to specify what happens next if the criterion is satisfied or not. Logical processing describes the tracking of true and false values through a procedure. Normally, in map algebra, a non-zero value is always considered to be a logical true, and zero, a logical false. Some operators and functions may return either logical true values (1) or logical false values (0), for example relational and Boolean operators. The return of a true or false value acts as a switch for one or other consequence within the procedure. Conditional processing allows that a particular action can be specified, according to the satisfaction of various conditions; if the conditions are evaluated as true then one action is taken, and an alternative action is taken when the conditions are evaluated as false. The conventional if–then–else statement is a simple example of a conditional statement:

if i < 16 then 1 else null where i = input pixel dn

Conditional processing is especially useful for creating analysis 'masks'. In Fig. 1.1, each input pixel value is tested for the condition of having a slope equal to or less than 15º. If the value tests true (slope angle is 15º or less), a value of 1 is assigned to the output pixel. If it tests false (exceeds 15º), a null value is assigned to the output pixel. The output could then be used as a mask to exclude areas of steeper slopes and allow through all areas of gentle slopes, such as might be required in fulfilling the prescriptive criteria for a site selection exercise.

1.1.4 Other types of operator

Expressions can be evaluated using arithmetic operators (addition, subtraction, logarithmic, trigonometric) and performed on spatially coincident pixel DN values within two or more input layers (Table 1.2). Generally speaking, the order in which the input layers are listed denotes the precedence with which they are processed; the input or operator listed first is given top priority and is performed first, with decreasing priority from left to right.

A relational operator enables the construction of logical functions and tests by comparing two numbers and returning a true value (1) if the values are equal or false (0) if not. For example, this operator can be used to find locations within a single input layer with DN values representing a particular class of interest. These are particularly useful with discrete or categorical data.

A Boolean operator, for example AND, OR or NOT, also enables sequential logical functions and tests to be performed. Like relational operators, Boolean operators also return true (1) and false (0) values. They are performed on two or more input layers to select or remove values and locations from the analysis. For example, to satisfy criteria within a slope stability model, Boolean operators could be used to identify all locations where values in one input representing slope are greater than 40º AND where values in an elevation model layer are greater than 2000m (as in Fig. 1.2a).

Logical operators involve the logical comparison of the two inputs and assign a value according to the type of operator. For instance, for two inputs (A and B) A DIFF B assigns the value from A to the output pixel if the values are different or a zero if they are the same. An expression A OVER B assigns the value from A if a non-zero value exists; if not then the value from B is assigned to the output pixel. A combinatorial operator finds all the unique combinations of values among the attributes of multiple input rasters and assigns a unique value to each combination in the output layer. The output attribute will contain fields and attributes from all the input layers.

All these operators can be used, with care, alone or sequentially, to remove, test, process, retain or remove values (and locations) selectively from datasets alone or from within a spatial analysis procedure.

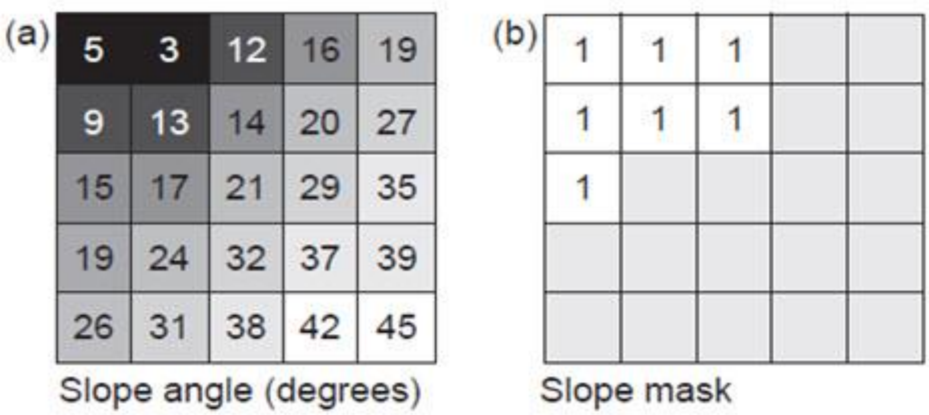

Fig. 1.1. Logical test of slope angle data, for the condition of being no greater in value than 15º: (a) slope angle raster and (b) slope mask (pale grey blank cells indicate null values). (Source: Liu, and Mason, 2009)

Table 1.2. Summary of common arithmetic, relational, Boolean, power, logical and combinatorial operator.

Arithmatic	**Relational (return true/false)**	**Boolean (return true/false)**
+, Addition -. Substraction *,Multiplication /, Division MOD, Modulus	= =. EQ Equal ^=,<>, NE Not equal <=.LT Less than/equal to <=LE Less than/equal to >,GT Greater than >=, GE greater than/ equal to	^,Not Logical complement & AND Logical AND l, OR Logical OR !, XOR Logical XOR
Power	Logical	Combianational
Sqrt, Square root Sqr, Square Pow, Raised to a power	DIFF, Logical difference IN{list}, Contained in list OVER, Replace	CAND, Combinational AND COR, Combinational OR CXOR, Combinational XOR

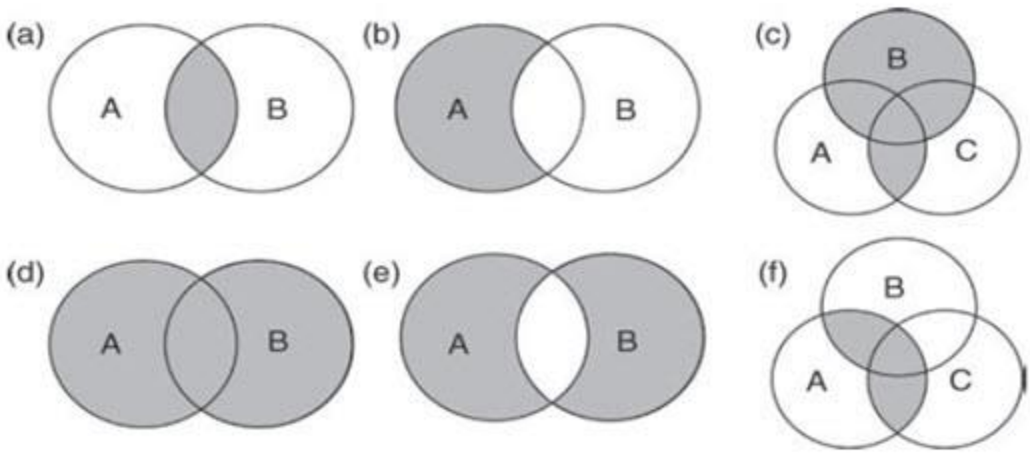

Fig. 1.2. Use of Boolean rules and set theory within map algebra; here the circles represent the feature classes A, B and C, illustrating how simple Boolean rules can be applied to geographic datasets, and especially rasters to extract or retain values, to satisfy a series of criteria: (a) A AND B (intersection or minimum); (b) A NOT B; (c) (A AND C) OR B; (d) A OR B (union or maximum); (e) A XOR B; and (f) A AND (B OR C).

(Source: Liu, and Mason, 2009)

Table 1.3. Summary of local operations.

Type	Includes	Example
Primary	Creation of a layer from nothing	Rasters of constant value or containing randomly generated values
Unary	Conversion of units of measurement and as intermediary steps of spatial analysis	Rescaling, negation, comparing or applying mathematical functions reclassification
Binary	Operations on ordered pairs of numbers in matching pixels between layers	Arithmetic and logical combinations of rasters
N-ary	Comparison of local statistics between several rasters (many to one or many to many)	Change or variety detection

(Source: Liu and Mason, 2009)