12

Space–Time Modelling and Error Propagation

The world we have considered so far is one of environments, landscapes, and interactions that change over space. Now in this chapter we will explore how dynamics over time can be included in our spatial modelling and analysis. Natural and human processes continually act on our environment at scales ranging from micro-scale rock weathering to global atmospheric interactions, and from individual humans through to international communities spanning the globe. Capturing those changes and the causes, interactions, and effects of natural and human processes through dynamic spatial modelling allows us to learn more about those environments. While every model is a limited abstraction of reality, simulations under present, past, and future conditions can help us understand subsequent states of environments that result from processes such as physical, chemical, biological, economic, and social reactions. Models thus have an important role in scientific explorations, management, and policy development.

Learning objectives

In this chapter we will focus on computational models representing space–time dynamics, which are developed in, or linked to, a GIS. By the end of the chapter, you will:

- **understand the various components and inputs to a computation model**
- **have learned how space–time processes may be represented in a model**
- **be able to describe how GIS may be used in space–time modelling**
- **understand how errors in data or process representations are propagated through modelling, and the impacts these have on resulting outputs.**

Modelling is behind so many of our day-to-day activities: from the shape of car we drive, or the weather forecast we listen to, through to the interest rates we pay on our loans. It impacts our understanding of the universe and even the fears we might hold for the future of planet Earth. Models are all simplified representations of a reality, yet with the advances in computational power that are now available, the processes and environments we can recreate and simulate in digital models seem to capture ever greater complexities of interactions, allowing us to explore how systems work and the reactions within. Models not only support our learning about current systems, they also allow us to explore new futures through investigating impacts on an area of any possible changes (a different policy, a change in demographics, or the effects of climate change on a country's water and food security).

We model for many reasons, but these fall roughly under two main headings:

1 To aid our understanding of a complex reality through simplifying, diagnosing, and examining interactions, causes, and effects.
2 To predict events, states, and outcomes under particular conditions.

In the following sections we will explore the basics of a computation model and then how these may be developed in a GIS environment.

12.1 Introducing computational modelling

Models come in various types and forms (conceptual, **physical**, and **analogous**) and use different devices to capture the reality that is of interest. For example, a scaled representation of a new housing development, made of cardboard (a physical model) such as the one shown in Figure 12.1 helps us understand and visualize the space, shape, and connections of its various elements and the surrounding environment. Each type of model has particular uses and brings insight and understanding to different areas of enquiry. In this chapter, the focus is on **computational models** as this is the environment of GIS.

Figure 12.1 Physical model of a new sustainable housing development (image copyright Centre for Sustainable Energy)

With this type of modelling, mathematical and logical operations are coded in computer binary (see Chapter 3) to represent the state of an environment and the processes involved. Simulations (experimentation with models) using varying scenarios and conditions can then be run to yield insights into dynamical changes and interactions, shown by numerical values for different variables.

Taking a very simple, spatially static model, we can represent mathematically the changes found across systems that come about as a reaction to variations in the conditions. For example, we can take an established statistical linear regression model relationship between the temperatures in the Swiss Alps and elevation: this is defined in equation 12.1. We can see from this equation that with increasing height it becomes colder:

$$T = 5.697 - 0.00443 * E \qquad 12.1$$

where T is in degrees Celsius and E is elevation in metres.

We can use this equation to determine the temperature across an area of the Swiss Alps. Taking a digital elevation model (see Chapter 11 for detailed examination of DEMs) as an input, the equation can be run for individual grid cells (each is treated as a separate entity). The resulting gridded temperature map show values across an area, and highlights the possibilities of applying a simple mathematical model to generating data for a new variable.

The art/science of formulating computational models requires us to consider a number of factors which will define how and what we represent. The goal is to produce a model that is realistic enough that it reflects the essential aspects of the phenomena being modelled, but simple enough that it can be translated into mathematical and computational formulae and code. This will be determined by answering various questions:

- Why do we want to model this?
- Where do we want to model and at what time and space scales?
- What is our understanding of the environment and processes acting in this area?
- What data exist that we can use in the modelling?
- What models exist already that could meet our needs?

Once these questions have been answered, we have a more bounded idea of the overall model environment, and the four basic stages of defining and developing a computation model can be followed:

a) Conceptual model—Describing, using words/diagrams, our understanding of how features and processes of a system work and interact.

b) Mathematical model—Translating the conceptual model into a series of mathematical equations that represent the various reactions and interactions between the variables that have been used to define the system. These may then be translated into numerical and analytical solutions where the complexity is great.

c) Computational model—Converting the mathematical equations into definitions and code that can be represented in a computer system, and establishing the input and output data set routines.

d) Linking models to reality—Comparing the model outputs to real observations is an important process, involving verification, calibration, and data assimilation to ensure the model represents as accurately as possible the world it is intended to represent.

The mathematical translation of a conceptual model involves defining the various elements of the representation—in particular the **forcing functions** and the processes acting—across defined time steps. It can be fundamentally represented as:

$$Z_{1...m} = f(I_{1...n}, P_{1...l}) \qquad 12.2$$

where $Z_{1...m}$ are the model output state variables, the model structure f is defined by a function or set of functions, forcing functions/external variables (inputs) $I_{1...n}$, with associated parameters $P_{1...l}$ being defined in two or three spatial dimensions.

In more detail, five main elements are used in defining a model mathematically and computationally (Jorgenson 2009):

1. ***State variables***—these describe the state of the system at a moment in time, resulting from processes acting on, and influenced by, the forcing functions. In a GIS, there are the derived points, lines, polygons, and raster values that are the outputs of the different model iterations, and variables might include the amount of precipitation, the level of groundwater in an aquifer, or the state of an ecosystem.
2. ***Forcing functions or external variables***—these are the data sets of independent variables that influence the system of study. In GIS terms, these may be stored as attribute values of points, lines, polygons, and rasters, and are the input variables, such as atmospheric physical and chemical conditions in climate change modelling, or economic and social settings that affect human activities.
3. ***Processes***—biological, chemical, physical, economic, and social—are represented by functions such as mathematical operations, behavioural rules, or logical equations (as given in Box 12.1). They describe the relationship between the forcing functions and the state variables, and between the state variables, and are based on theoretical or empirical relations (defined from real-world data).
4. ***Parameters***—coefficients in the mathematical representations of the processes described in 3, which represent the influence of a subprocess or other variable of influence that is not fully defined in 1, 2, or 3 and may be a constant. For example, in modelling run-off from a hillslope, parameters might be used for fixed or varying infiltration capacity and hydraulic conductivity of the soils.
5. ***Universal constants***—such as gravitational acceleration of 9.80665 m/s².

Taking water flowing across a hillslope as an example, there are many different processes at play, including the gravitational pull downhill; the friction and other stresses from the landscape surface that slow this movement down; the movement of water into the soil and then through to the groundwater systems or into the channel. The following data might be used:

- *State variables*: soil moisture levels; river flow; groundwater level.
- *Forcing functions*: precipitation; temperature; land cover; soil types; slope and elevation.
- *Processes*: flow over the surface; evapotranspiration; infiltration; flow through the soil system; flow in the groundwater.
- *Parameters*: these are coefficients that account for subprocesses such as flow resistance from vegetation cover, or hydraulic conductivity.
- *Universal constants*: gravitational constant as this influences the flow rate downslope and in the channel.

12.2 Capturing spatio-temporal dynamics in computation modelling

The ability to define and represent each of the five modelling elements depends on the state of our knowledge and the information we have available to describe them. This is particularly complex when variations over both space and time are included in this definition. Establishing the spatial and temporal data models that form the basis of the modelling will not only influence the form of the input data, but also how the processes are defined. This leads to different types of computational models: lumped, semi-distributed, or distributed; and black-box, physical, and probabilistic.

Defining space and time representations

Defining the spatial dimension involves demarcating the extents of the study area as well as how the space within it is divided up and the resolution of the smallest unit in the model. The spatial data model adopted will influence how the various components are defined. Where the functions, variables, and processes are represented as spatially continuous (known as distributed modelling) then a raster approach is usually adopted (cf. Chapters 2 and 3). An alternative is where a series of distinct spatial units are defined and within each unit the variable values are the same for a given time step. This is known as **semi-distributed modelling**, and may be based on either aggregated grid cells or vector-based entities. Lumped modelling is a further simplification of the spatial representation in which fewer, more aggregated spatial units are defined. For example, in hydrological modelling a catchment may be divided into sub-basins and water flow responses are generated for each lumped unit, and these are then integrated to derive predictions for the dynamics of the whole system.

In defining the representation of time dimensions, the temporal resolution of the models is specified in terms of length of the model's time step and is usually based on equal time sequences across a period, or centred on an event. The time step and period of modelling can range from seconds and minutes through to decades and millennia. Where fine resolution time steps are used, or the modelling covers long periods, large numbers of data files may be used, requiring considerable processing and storage capabilities. This is typical of climate change analysis today.

Defining processes

Representing processes involves converting the conceptual understanding or theoretical knowledge into definitions based on mathematical or logical operations (given in Box 12.1). A number of modelling approaches may be used, with the representation reflecting factors such as the purpose of the model, the knowledge of the system, and available field data.

Many mathematical environmental models have used the relatively simple 'black-box' approach in which the real world is described through statistical regressions and correlations, as equation 12.1 exemplified. The definition of the process is based on a statistical or other simply defined relationship between one or more independent variables and their influence on dependent variables. This relationship is developed using measured data for both the forcing functions and the state variables. In the resulting, usually statistically derived, mathematical formulae, the processes are not explicitly defined but the influences of the forcing functions on the state variables are represented through parameter values which may be adjusted to account for the extent of this influence.

For example, **black-box modelling** of hillslope run-off processes is often centred on simple regression equations expressing the relationship between rainfall and run-off, with processes such as interception, infiltration, overland, and throughflow described through constant parameter values (see equation 12.3). Relationships defined by such regression equations are usually based on local field data (empirical) evidence, so the environment and its responses captured in the formulae apply only to this area. They should not be extrapolated or transferred to environments outside the ranges of the data originally used to define them.

$$\text{Run-off} = \text{Rainfall}^{\,a,\,b,\,c,\,d} \qquad 12.3$$

where a, b, c, d are numerical parameters that represent different processes affecting run-off.

In contrast, where there is access to greater knowledge or data, or more detailed insight is needed, a physically based (also called mechanistic) approach involving more complex sets of mathematical functions may be used. This has been used in a wide variety of applications, for instance to represent causes and effects in crop growth, air quality, groundwater movement, pesticide leaching, and epidemiological hazards. The dynamics of the system are based on theoretical knowledge of the dynamics, and are represented using differential or partial differential equations, which are resolved using analytical or numerical methods to obtain exact or approximate solutions.

Navier–Stokes equations, which describe the physics of the motion of fluids, are an example of differential equations used in some modelling systems today. The equations arise from applying Newton's second law and other influences on the movement of air and water. They mathematically represent fundamental processes and are used in modelling for weather forecasting or for exploring phenomena such as ocean currents. The differential equations are solved across an area using numerical and analytical approximations such as finite difference (regular grid) or finite element (irregular triangles, polygons, etc.)—see Section 2.6.

In this exploration so far, we have assumed that the state variables at any moment in time may be explained directly in terms of their previous state, the external forcing functions, and the process definitions. This approach may be expressed very simplistically as follows:

$$Z_{(t+1)} = aZ_{(t)} + by(t) \qquad 12.4$$

where:

$Z_{(t)}$ and $Z_{(t+1)}$ = state of system at times t and $t + 1$

a and b = constants

$y(t)$ = forcing functions and process definitions.

The assumption in **deterministic modelling** is that there is only one solution for the state variable, given particular defined inputs. As such, it does not take on board the inherent uncertainty or variability in defining the processes and variables. The alternative approach, **probabilistic modelling**, allows many possible outputs to be generated for a given input and state. Models can then utilize the whole range of possible states of the variables, and the result is a probability distribution of the model outputs rather than one single value. The corresponding simplistic equation (cf. equation 12.4) in probabilistic modelling is:

$$Z_{(t+1)} = aZ_{(t)} + by_{(t)} + r \qquad 12.5$$

where the new term, r, is a **random variable**.

The randomness may be introduced in models through values for forcing functions, parameters, and process definitions. Methods such as Monte Carlo techniques have been used for exploring the complete range of outputs that a model can generate as a result of the probability distributions of both the input data and the model parameters. **Monte Carlo simulation** adds random values for r from a defined probability function, for any factor that has inherent uncertainty. It then calculates results for different iterations—each time using a different set of random values from the probability functions (such as normal, log-normal, Gaussian distributions)—so it may involve hundreds or even thousands of recalculations. The end result is a distribution of possible outcome values.

Box 12.1 Modelling operations

a) *Arithmetical operations*

New attribute is the result of addition (+), subtraction (−), multiplication (*), division (/), raising to power (**), exponentiation (exp), logarithms (ln—natural, log—base 10), truncation, square root.

b) *Trigonometric operations*

New attribute is the sine (sin), cosine (cos), tangent (tan), or their inverse (arcsin, arccos, arctan), or is converted from degrees to radians or grad representation.

c) *Statistical operations*

New attribute is the *mean*, *mode*, *median*, *standard deviation*, *variance*, *minimum*, *maximum*, *range*, *skewness*, *kurtosis*, etc. of a given attribute represented by *n* entities.

d) *Multivariate operations*

New attribute is computed by a multivariate regression model, or *principal component analysis*, *factor analysis*, *correspondence analysis* transformation of multivariate data.

e) *Rule-based operations*

New attribute is computed based on various conditions being met/not met *If … Then … Else* statements.

f) *Logical operations*

New attribute is computed based on logical operations Truth or falsehood (0 or 1) resulting from union (∨ logical OR), intersection (∧ logical AND), negation (¬ logical NOT), and exclusion (⊻ logical exclusive or XOR) of two or more sets.

12.3 GIS-based computational modelling

GIS have been used as the basis for developing spatio-temporal dynamic modelling for many different applications. At the most basic level, GIS are loosely coupled to a model and used as the platform for generating spatial data for defining the external forcing functions and for visualizing the state variable outputs. In other applications, models are developed within the GIS using built-in spatial functionality, specialist modules, or dynamic modelling/scripting tools as well as the spatial data modelling capabilities to define the processes, forcing functions, and parameters. There are thus a variety of approaches that might be adopted—the following examples will highlight the different possibilities (Beven and Moore 1994; Goodchild et al. 1996; McDonnell 1996; Skidmore 2002; Maguire et al. 2005; Koomen et al. 2007; Pfieffer 2008; Brimicombe 2010; Khalema-Malebese and Ahmed 2012; O'Sullivan and Perry 2013; Wainwright and Mulligan 2013).

Loosely coupled models and GIS

Loosely coupled models may have different degrees of system integration with GIS. At the most basic, a GIS may be used to derive model parameters and variables' values from data layers already held in the database. This

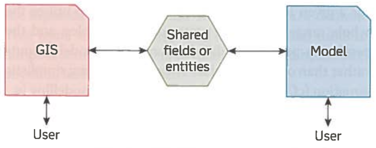

Figure 12.2 Loosely coupled GIS and model

approach harnesses the spatial data-handling capabilities of a GIS by loosely coupling this to more sophisticated time representation in dynamic computational software. The coupling of the GIS and dynamic models requires a cross-matching of the corresponding individual spatial units (entities or grid cells) used in each system.

Other aspects of the models, such as the domains, operators, and algebra, remain separate in loose coupling, so internally the two systems still retain individual identities. There is no sharing of the database or modelling functionality between the two systems, with interchange files passing data between the two systems (as shown in Figure 12.2). Varying degrees of subroutine development or scripting are required, depending on the nature of the GIS and modelling, to bolt the systems together. The link is often not seamless, with, for example, one system being exited before the other runs and the importing/exporting of files between the two.

The advantage of this approach is that the algorithms of the dynamic model are run separately, which allows the functionality and structure of the host programming language to be employed while the GIS permits a *spatially* distributed approach to be adopted. GIS thus serve an important role in preparing variable data for forcing functions and initial state conditions, and estimating parameters, using the various functions and capabilities highlighted in earlier chapters (McKinney and Cai 2002; Imam and Kushwaha 2013). For example, McDonnell (2000) linked a GIS to the dynamic modelling software Stella to explore the impacts of different dam locations and operational schedules on downstream water quantity and quality variables and biota. GIS may be used to derive areally averaged values for variables from various point data using the interpolation methods described in Chapters 8 and 9. Secondary attributes (see Chapters 10 and 11) may also be derived which characterize the spatial variability of specific forcing functions such as potential solar radiation ratio, slope, aspect, and topological data such as road or river network structure. GIS' spatial visualization capabilities are also often used to display the results of the modelling.

The linking of modelling to GIS can be used in many different contexts (Case study 12.1 highlights a case from the Middle East). Gret-Regamey and Straub (2006), for example, loosely linked a Bayesian network to a GIS to develop a spatially explicit risk assessment for avalanches in Switzerland. The Bayesian network linked a model-derived probability assessment of snow pressure at defined locations, with spatial data held in a GIS to estimate the probability of damage to buildings and transport infrastructure, with or without protection measures and strategies in place.

The loosely coupled approach is often used when model definitions—particularly of processes—are complex and the relatively limited functionality of many GIS does not support the mathematical or logical expressions required. Loose coupling is also popular when a model needs to be run many times (e.g. for multiple time steps or for many small spatial units), as these computations can often be processed much faster by stand-alone modelling software, without the overhead of the GIS. Loose coupling recognizes the advantages of the two systems—of GIS for storing large volumes of spatial data, and of models for rapidly computing the changing attribute values—and keeps the advantages of each. This, coupled with the limited capabilities of representing temporal variables in current GIS (discussed in Chapter 2) has ensured that loosely coupled applications are common.

Raster-based modelling with GIS

For some applications, model development is possible within a GIS, bringing with it the benefits of a single, integrated operations interface and data sharing (e.g. Pagelow and Olmedo 2005). The raster spatial data model is by far the most commonly adopted basis for this modelling. The approach builds directly on the ideas, formal logic, and primitive operatives set up under map algebra and the **Map Analysis Package** (MAP) for undertaking spatial modelling across two or more raster data layers (Tomlin 1983, 2012)—described in detail in Chapter 10. Using these concepts, spatial operations may be defined, based on one or more raster cells in the corresponding positions in the different data layers, to derive a new model output value. As the following list highlights, the first two are local operations with model outputs defined by the state, and the external functions and processes affecting a cell. The operators used to define the processes can be linked through scripting to derive complex process representations:

a) Point operations: operations calculated locally on individual cells or point locations.

b) Neighbourhood operations: operations in which a value is computed for a particular cell, based on the values of surrounding cells.

Case study 12.1
Example of loosely coupled modelling: the MAWRED programme

Given the natural aridity in the Middle East North Africa (MENA) region, managing water effectively and efficiently is a vital concern for decision-makers. With agriculture continuing to use the lion's share of the water, the challenge is to balance water and food security demands, which are often directly at odds with each other. With groundwater resources steadily declining, but management hampered by limited data availability in most countries, there is an urgent need to develop new insight through means other than field observations.

The *MAWRED* programme (Modelling and monitoring Agricultural and Water Resources Development) uses hydrological modelling supported by observations derived from remote sensing to give an overview of water resource availability. By linking to crop maps and irrigation water use models, an understanding of the current extent of the use of water in agriculture (currently largely unknown), and so the impacts of current policies, were derived. NASA's MENA-Land Data Assimilation System, a catchment land surface model developed for the MENA region, was run to generate groundwater, surface water, soil moisture, and evapotranspiration values at the sub-basin levels. The catchments were defined using digital elevation data in a GIS and transferred to the model. Irrigation area data were derived from the image processing of stacked remotely sensed data and then further analysed in a GIS for use in the hydrological modelling.

The initial model outputs were compared to groundwater values derived from the GRACE (Gravity Recovery and Climate Experiment) satellite system (see Figure 4.4d) for the region, using data assimilation methods within the model simulation exercises, to maximize the model's

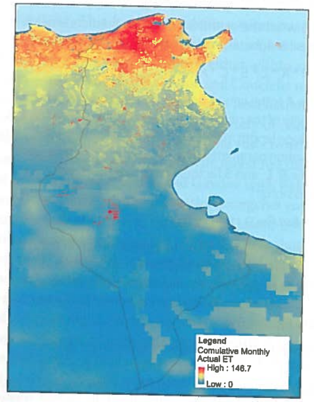

Figure 12.3 Modelling and satellite data were used to derive values for actual evapotranspiration over Tunisia, North Africa

accuracy. The final model outputs were then verified with field data where available. Once the most accurate results were obtained, monthly maps were generated by the model and the high-quality maps (see figure) and exchange files were created in the GIS so that decision-making organizations across the region could use this information in their own spatial analytical systems. (*Mawred* means 'the source' in Arabic; see http://www.mawredh2o.org for more details.)

c) Zonal operation: operates on cells with the same value, such as all the cells in a given land cover class, or urban zone.

d) Global operations: where an operation generates a value for the whole area.

e) Network operations: operations that generate values for nodes and arcs based on their topological connections in a network.

Building from these static (single time step) modelling concepts (see Chapter 10 for more detail and possible variable calculations), dynamic raster modelling incorporates temporal changes by allowing values to be computed for each time step on a cell-by-cell basis, or across neighbourhoods or regions. Values for the previous time step are used to deterministically calculate updated values for the spatial units for the next time step, following equation 12.4. The modelling thus extends non-temporal spatial models to those based on multiple time steps.

As an example of point-based raster modelling, potential soil erosion values from maize cultivation were generated for the Kisii District of Kenya introduced in Chapter 7. Agricultural intensification may increase soil losses here, and modelling can be used to identify which areas are most prone to this. Two different empirical erosion models were used to simulate the soil losses: the Universal Soil Loss Equation (USLE) (Wischmeier and Smith 1978) and the Soil Loss Estimation Model for Southern Africa (SLEMSA) (Stocking 1981; Elwell and Stocking 1982) (see Box 12.2). These are both well known and easy to use, and the data for both are readily available in many countries. The disadvantage of these empirical models, though, is that they are oversimplifications

Box 12.2 Empirical soil loss models

The Universal Soil Loss Equation (USLE)

USLE Wischmeier and Smith 1978) predicts erosion losses for agricultural land by the empirical relation:

$A = R * K * L * S * C * P$

where A is the annual soil loss in tonnes h^{-1}, R is the erosivity of the rainfall, K is the erodibility of the soil, L is the slope length in metres, S the slope in per cent, C is the cultivation parameter, and P the protection parameter.

The R, L, and S factors are derived from empirical regressions:

R factor: $R = 0.11\,abc + 66$

where a is the average annual precipitation in cm, b is the maximum day precipitation occurring once in two years, in cm, and c is the maximum total precipitation of a shower of one year occurring once in two years, also in cm.

L factor: $L = (l/22.1)^{1/2}$ where l is the slope length in metres.

S factor: $S = 0.0065s^2 + 0.0454s + 0.065$ where s is the slope as per cent.

The Soil Loss Estimation Model for Southern Africa (SLEMSA—Stocking 1981; Elwell and Stocking 1982):

Control variables:

E	Seasonal rainfall energy (J/m^2)
F	Soil erodibility (index)
i	Rainfall energy intercepted by crop (per cent)
S	Slope steepness (per cent)
L	Slope length (m)

Submodels:

Bare soil condition $K = \exp[(0.4681 + 0.7663F)$. $\ln E + 2.884 - 8.1209F]$

Crop canopy $C = \exp[-0.06/]$

Topography $X = L^{0.5}(0.76 + 0.53S + 0.076S^2)/25.65$

Output: predicted mean annual soil loss (tonne ha^{-1})
$Z = KCX$

of erosion processes, but specialist models are often location-specific or require data that are not readily available.

The models were run for the Kisii area, computing soil losses per grid cell. As the models are point-based,

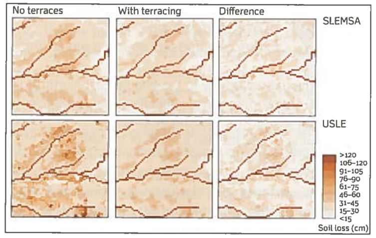

Figure 12.4 Results of grid-based soil erosion modelling using the SLEMSA and USLE models and with different land protection practices

there is no accounting for the soil loss or gains in neighbouring grid cells. Simulations were run both with and without the inclusion of the 'trash line' terraces—a practical measure to reduce losses, used by the farmers and constructed with maize stalks along the contours. The resulting estimated soil losses, with and without the effects of the trash lines, suggest that erosion will be concentrated in certain areas (see Figure 12.4). Given that empirical equations were used in the analysis, it is important to verify the results using local field data as we have no way of knowing their accuracy otherwise.

The Kisii example highlights environmental modelling in a GIS involving relatively simple mathematical functions. Among the many commercial and open-source GIS there is a great variety in the operational functionality that is available to represent the definitions of processes across the raster environment. Many support a relatively limited range, with mathematical and logical operations most commonly offered (cf. Box 12.1). More purpose-developed GIS provide a developed modelling language that supports more sophisticated representations of processes and the influencing variables. One of the first raster GIS to incorporate spatio-temporal dynamic modelling was the open-source PCRaster, which incorporated a dynamic generic modelling language including libraries of model building blocks and analytical functions (van Deursen 1995; Wesseling et al. 1996; Burrough 1998; Karssenberg et al. 2001; Burrough et al. 2005).

The core modelling component of this system is the PCRcalc module, a raster map calculation engine. It provides a large number of functions and operators which take maps as their input and return maps as their results. These can be combined and nested to perform complex operations. Users script the data generation and process representations using its internal language PCRcalc, or through the more generic Python (Karssenberg et al. 2007). An example of PCRaster use is given in Box 12.3.

Box 12.3 Example of raster-based dynamic modelling using PCRaster

In the Netherlands, many heathlands formerly dominated by heather (Calluna) have become dominated by grasses (Deschampsia). These changes are thought to be induced by factors such as local nitrogen enrichment—eutrophication—and the influence of the heather beetle. During heather beetle outbreaks it is observed that Calluna is more severely affected. Further, it is hypothesized that outbreaks of the beetle are stimulated by nitrogen enrichment, and that they become more severe and occur more frequently under eutrophic conditions. No outbreaks are observed in young Calluna stands (< 5 years). During an outbreak of this beetle plague, Calluna plants die off almost completely over large areas, opening up canopy for enhanced growth of grasses.

The CalGIS (Calluna GIS model) was developed in PCRaster to simulate the heather–grasses competition in the Netherlands, using the soil map as the basic input map (Van Deursen and Heil 1994). It optimistically assumes sandy soils in the Netherlands to be potential heather habitat. The spatial dynamic behaviours of the grass, heather, and beetle are calculated over a 100-year period with each year represented as a file in a dynamic map stack.

For example, the heather growth is simulated using the following PCRcalc statements within a model script:

```
h_growth = (0.005 * nutrient * heather + 0.22 * heather)
           * (100-grass-heather)/100

h_death = 0.12 * heather

heather = min(max(heather + h_growth - h_
          death, 0), 100)
```

The various dynamics can be visualized and analysed using the 'Animate map stack' button or through time plots showing the spatially averaged development of the heather over the 100 time steps. The simulation can be run for different eutrophication scenarios by selecting another level for the nutrient variable.

Cellular automata modelling

A form of spatio-temporal modelling used increasingly in a variety of applications in recent years is **cellular automata** (CA). This form of modelling, often based on a regular grid of cells, is by definition spatial, but in recent CA models it has been developed either within a GIS or loosely coupled to one (Takeyama and Couclelis 1997; see Baetens and Baets 2012 for irregular tessellations). CA has been shown to simulate well a variety of real-world systems, including in studies of people and crowd movement, urban landscapes and growth, forest fire diffusion, and disease spreads (Clarke and Gaydos 1998; Batty et al. 1999; Geertman et al. 2007; Piyathamrongchai and Batty 2007; Stevens et al. 2007; Yassemi et al. 2008; Yeh and Li 2009; Collin et al. 2011; Liu et al. 2013). The unique character of this relatively simple form of modelling is that processes (often social, physical, or biological activities) are defined using sets of rules that affect the state of each grid cell, suiting applications where behaviours cannot be easily represented by strict mathematical equations. The end result is that complex macro-level phenomena (patterns) can emerge from relatively simple interactions between values of local grid cells. Geographical constraints or boundary conditions can also be represented implicitly through the arrangement of the grid cells.

The modelling is based on four major elements: cells, states, and neighbourhood and transition rules. Each grid cell has one of a finite number of states, with the initial state (and boundary conditions) specified through constructed or generated data, and these can have a large influence on the subsequent evolution of the system. With each time step, the new state of each cell is determined according to some fixed deterministic or probabilistic transition rule, based on its current state and that of the cells in its neighbourhood. The transition rule is usually the same for each cell, does not change over time, and is applied to the whole grid simultaneously. It is developed from typical operators—usually +, –, *, /, div, mod, and or not, or and, =, !=, <, >, ≤, ≥—and defined as a deterministic or probabilistic mathematical function, or:

If expression then Else statement

If expression then statement 1 Else statement 2

The definition of the neighbourhood itself—the zone of influence of a grid cell—can vary with the two typical configurations known as von Neumann (1966) and Moore (shown in Figures 12.5 and 12.6 respectively). It may also be in either one or two dimensions: in two dimensions it might be like Figure 12.5b or Figure 12.6b.

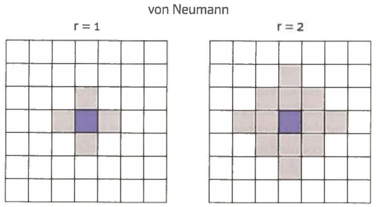

Figure 12.5 The von Neumann cellular automata neighbourhood configuration for (a) one and (b) two dimensions

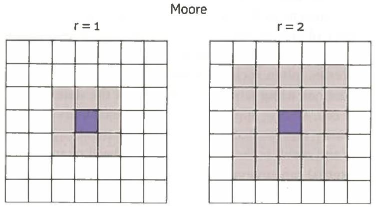

Figure 12.6 The Moore cellular automata neighbourhood configuration for (a) one and (b) two dimensions

A good example is the research of Yassemi et al. (2008) in which fire behaviour was explored using a raster-based model in a GIS, with CA modelling incorporated using its application programming language. The study is based on a deterministic model of fire spread with a CA Moore configuration of eight cells. The transition rules are based on the understanding that fires spread from one neighbouring cell to the central one only when the neighbour is completely burning. Thus the burn state of a cell is important in determining its impact on neighbouring cells, and this is defined as a ratio of burned to total area, with values ranging from 1 (completely burned) to 0 (unburned) along a continuous scale. The state of each cell is calculated at each time step, using trigonometry to derive the direction of travel and speed of the fire and so the area burned at the end of the time interval.

In the model, once a cell is ignited fire travels through it at a speed determined by variables such as fuel type, moisture content of the foliage, topographic and weather factors, and the type and duration of prediction. The GIS is used to derive this 'rate of speed vector' for individual cells, combining these different layers. The outputs are raster arrays of fire values at the end of each time interval, which are visualized using the GIS or using dynamic animation which effectively highlights the output fire dynamics.

Agent-based modelling

The modelling described so far has represented the reactions and interactions between forcing functions, processes, and states that are variable across an area. However, the effects on these environments of individual 'agents', such as humans or animals moving across and *interacting* with a landscape, are not included. Important developments over the last decade in **agent-based modelling** (ABM) within or loosely coupled to a GIS now allow these to be included in the simulations, bringing new insight and dimensions to the understanding of certain processes (see Heppenstall et al. 2012 for detailed coverage). Building on the body of knowledge developed from non-spatial ABM applications in areas such as financial markets or immune systems, successful applications using a GIS include studies of interactions of people in an urban environment, environmental degradation by humans, disease spread, invasive species transmission, and shifting cultures (Gimblett 2002; Batty 2005; Brown et al. 2005; Manson and Evans 2007; Crooks et al. 2008; Heppenstall et al. 2012; Torrens et al. 2011; Torrens 2012; Johnston 2013; Crooks and Wise 2013).

ABM is, like CA, structured on a gridded cell base with rules governing the reactions of interacting agents. Underlying environmental conditions only change in response to the agents' actions. Thus the most important part of this modelling centres on representing in detail the ABM agents and their behaviours, using individual rule-based profiles which define their interactions not only with the environment but also with each other. These can include conditional decision-making and other non-linear rules that distinguish them from mathematically based raster modelling systems. Their dynamics and relationships to the wider environment tend to be based on individual and neighbouring cells, and the overall scale of the modelling tends to be more local than the large landscapes of the raster modelling described earlier. It is through these rules and interactions that patterns of reactions and changes emerge, giving a complexity that is difficult to predict.

In recent work on urban growth, Kim and Batty (2011) used agent-based modelling to explore possible changes in the southern fringe of Seoul city, Korea. Taking an area of 25 × 25 km with cell sizes of 50 (i.e. 250 000 cells), potential scenarios were explored for the release of green-belt land and the introduction of new transit stations for a high-speed train. The agent-based modelling links microeconomic residential location choice theory with urban growth, taking on board interactions between buyers and sellers and varying preferences amongst significant sets of agents. By representing various household location decisions, the modelling showed the emergence and evolution of urban growth structures arising from the spatial heterogeneity, and evolving over time.

12.4 Accounting for errors in modelling

In spatial analysis in general, and modelling in particular, it is important to understand the accuracy and uncertainty around the results as it is these that determine the extent to which they can be used and relied on in the application of interest. There are three major determinants of this accuracy:

- **a)** the quality of the data (see Section 4.7)
- **b)** the quality of the model, and
- **c)** the way the data and the model interact through processing.

Most GIS or modelling systems do not provide analytical means to determine the effects of errors or uncertainty on the results of the model, so it is important to undertake further analysis, usually away from the modelling environment. It is particularly important to understand the uncertainties in the input spatial data and the propagation of any associated errors from this through the models during processing.

The effects of uncertainty from input variable data can be compounded by inaccuracy in elements of the model definition itself, such as the parameter coefficients that are included. It is useful to be able to quantify all the errors—δu—in the output maps of u. This knowledge can be used to improve either the model or the data collection.

The options for improvement include:

- **a)** using 'better' methods for spatial interpolation or using numerical models instead of simple logic
- **b)** collecting more data and optimizing sampling
- **c)** collecting different data
- **d)** using 'better' models—either models which are more complete or which give lower prediction errors, or are better matched to the spatial and temporal data resolution available
- **e)** improving the model calibration
- **f)** improving the spatial and/or temporal resolution by matching correlation structures.

Determining error propagation using Monte Carlo simulation

Given that (a) we are aware of the possibility of statistical errors in the data, and (b) we have the means to quantify these errors (using ordinary statistics or geostatistics, stochastic simulation (see Chapter 9) or retrospective validation with independent data), the question arises of how to use this information to quantify and then reduce the inaccuracies that may accrue in the results of computational models. Put simply, if a new attribute U is defined as a function of inputs A_1, A_2, ... A_n, we want to know what the error is that is associated with U, and what the contributions are from each A_n to that error. If we can solve this problem, we can attach a pedigree to the results of particular modelling and can compare the results of different scenarios with confidence.

The simplest, but very computer-intensive, approach to error propagation is to treat each variable or attribute as having a Gaussian (normal) **probability distribution function** (PDF) with known mean μ and variance σ^2 for each entity or cell. In the simplest case we would use a single PDF for all cells in a layer and assume **stationarity**. If more information about spatial contiguity is available we can use conditional simulation to estimate cell-specific PDFs that reflect the location of known data points and the spatial correlation structure of the attributes (Chapter 9).

The arithmetical operation to derive new data is then carried out, not with the single mean values, but using a value that is drawn from the PDFs for each cell. To take care of the variation within the PDFs, the calculations are repeated many times (at least 100 times) to compute the mean result per entity or pixel and its standard deviation. The technique is popularly known as the Monte Carlo method, because of the random or chance element in each estimation.

Although the Monte Carlo method is computer-intensive (known as a 'brute force' technique), it provides interesting information about how possible errors in the data can affect the results of numerical operations in different parts of a geographic area. For example, in Chapter 10 the operations for computing the derivatives of a raster DEM such as slope were described, as well as the impact of RMS errors (root mean squared error) over the whole domain.

In estimating the uncertainty, users are then able to make more informed decisions on how to use the data sets and the results from the modelling exercises. For example, small relative errors in a DEM can strongly influence the locations of computed stream channels; so calibrating a model for these channels alone may optimize it for only one out of a large range of possible catchment characteristics. And the same holds true for validating a model. If validation measurements are made in locations where the probability is large that a stream flow of magnitude F corresponds to a given upstream contributing area, then the validation will be robust. If validation measurements are made in areas where the probabilities are lower, then the chance that the predictions and the validation measurements will converge is naturally smaller. This explains why in hydrological modelling a simple validation based on stream flow at the outlet of the catchment will be robust, while validation measurements made at randomly chosen locations in the catchment may be subject to considerable error.

Analytical approaches to error propagation

Although Monte Carlo methods of error analysis are straightforward and can be adapted to many kinds of numerical modelling, even today they require considerable computing resources. Fortunately, for many numerical models used in GIS to compute new attributes from existing properties of entities or cells this can be achieved using the standard statistical theory of error propagation (Heuvelink 1998). The problem is then that, for each entity or cell we should estimate, the error in the output value U as a function of the errors in the input values A_i:

$$U = f(A_i) \qquad 12.6$$

includes only arithmetical relationships (+, –, *, –s–, raising to powers, exponentiation, etc.).

Consider the situation in which the value of an attribute on map A is not exact but has an associated error term δ so that the value of the attribute cannot be better known than $A \pm \delta a_i$. A_i could be the value of readily available water in a soil mapping unit that is assumed to be statistically homogeneous. We wish to combine the readily available soil water with an estimate of irrigation effectiveness A, with an error $A_j \pm \delta a_j$. If the attributes A_i and A_j are statistically independent, and if δa_i and δa_j are each of the order of 20 per cent, it can be shown that the error of total available water δu in the computation of $U = (A_i + A_j)$ is of the order of 28 per cent. For cartographic overlay operations involving more than two steps, the increase in error can be explosive.

Box 12.4 presents the partial differential equations for the simple theory of error propagation. Using these equations we can examine how errors propagate through simple **bivariate** models with $a_1 = 10 \pm 1$ and $a_2 = 8 \pm 1$.

Box 12.4 Simple theory of error propagation

Considering only random, independent errors, for a relationship

$$u = f(a_1, a_2, a_3, \dots a_j) \qquad \text{B12.4.1}$$

in which the a_js are all independent, Su, the standard deviation of u, is given by

$$Su = \left[\sum_{i=1}^{j} (u/a_i)^2 \cdot Sa_i^2\right]^{1/2} \qquad \text{B12.4.2}$$

and the standard error of u, SEu, is given by

$$SEu = \left[\sum_{i=1}^{j} (u/a_i)^2 \cdot SEa_i^2\right]^{1/2} \qquad \text{B12.4.3}$$

where SEa_i is the standard error of a_i.

These formulae hold when there is no correlation between the x_is. When they are correlated, an extra term must be added to express the increase in error in u due to correlation. This term is:

$$\left[\sum_{i=1}^{j}\sum_{j=2}^{j}\{\delta u/\delta a_i \cdot \delta u/\delta a_j \cdot Sa_i \cdot Sa_j \cdot r_{ij}\right] \qquad \text{B12.4.4}$$

Addition or difference operations—no correlation variables

Let $u = a_i \pm a_j \pm \dots$, then $\delta u/\delta a_1 = 1$, $\delta u/\delta a_2 = \pm 1$.

$$Su = \sqrt{(Sa_1^2 + Sa_2^2)} \qquad 12.7$$

so

$$u = 10 + 8 = 18$$

and

$$Su = \sqrt{(1 + 1)} = 1.414$$

The absolute error of u is greater than either a_1 or a_2, but in the case of addition, the relative error (1.414/18 = 8 per cent) is lower than for the original variates (10 and 12.5 per cent). For subtraction, the absolute error Su is the same, but the relative error is now much greater at (1.414/2 = 70 per cent). Whereas addition of two random numbers, and hence of two maps, can be thought of as a benign operation with respect to error propagation, subtraction can lead to explosive increases in relative errors, particularly when a_1 and a_2 are similar in value.

When a_2 is a constant (i.e. $u = a_1$ + constant), there is no difference in the variance of u and a_1. Adding or subtracting constants has no deleterious effect on errors.

Addition operations of correlated variables

When the variables $a_1, a_2, \dots$ are correlated, the term given in equation B12.4.4 (in Box 12.4) must be included in the computation of the error of u. Let $u = a_1 + a_2$, in which ra_1a_2 expresses the correlation ($-1 \le r \le 1$) between a_1 and a_2.

$$Su = \sqrt{\{Sa_1^2 + Sa_2^2 + 2Sa_1 \cdot Sa_2 \cdot ra_1a_2\}} \qquad 12.8$$

and

$$Su = \sqrt{\{1 + 1 + 2 \bullet 1 \bullet 1 \bullet ra_1a_2\}}$$

If a_i and a_j are 100 per cent positively correlated, the error in u can be as much as, but not more than, the sum of the errors of a_i and a_j. If a_i and a_j are negatively correlated, then the error in u, Su, could be less than if a_i and a_j were independent.

Multiplication or division operations—no correlation variables

Let

$$u = a_1^c \cdot a_2^d \qquad 12.9$$

where c and d are assumed exact constants. Then,

$$\delta u/\delta a_1 = ca_1^{(c-1)} \cdot a_2^d \text{ and}$$
$$\delta u/\delta a_2 = da_1^c \cdot a_2^{(d-1)}$$

so by equation B12.4.4

$$Su = \sqrt{\{c^2 \cdot a_1^{2(c-1)} \cdot a_2^{2d} \cdot Sa_1^2 + d^2 \cdot a_1^{2c} \cdot a_2^{2(d-1)} \cdot Sa_2^2\}} \qquad 12.10$$

Therefore if

$$u = a_1 \cdot a_2$$

then

$$u = 8 * 10 = 80$$

and

$$Su = \sqrt{\{a_2^2 \cdot Sa_1^2 + a_1^2 \cdot Sa_2^2)}$$
$$= \sqrt{(64 * 1 + 100 * 1)}$$
$$= \sqrt{164}$$
$$= 12.8$$

Multiplication not only raises the absolute error, but also the absolute error, in this case to 12.8/80 = 16%.

When a_j is a constant, c, i.e. $u = a_1 \bullet c$, the error propagation reduces to

$$Su = \sqrt{\{c^2 \cdot Sa_1^2\}} \qquad 12.11$$

Raising to powers

For

$$u = Ca_1^c \qquad 12.12$$

where C and c are constants, note that a_1 is perfectly correlated with itself so that the error of u, Su, is given by

$$Su = \sqrt{\{C^2 \cdot c^2 \cdot a_1^{2(c-1)} \cdot Sa_1^2\}} \qquad 12.13$$

For $a = 10 \pm 1$ in the expression $u = a_1^2$

$$u = 10^2 = 100$$

and

$$Su = \sqrt{\{(2a_i)^2 \cdot Sa_i^2\}}$$
$$= \sqrt{\{20^2 \cdot 1\}} = \sqrt{400}$$
$$= 20$$

Not only has the absolute error increased, but the relative error (= 20/100 = 20%) has also doubled.

Logarithmic operations and other relations

Let

$$u = C \ln a_i \qquad 12.14$$

then

$$\delta u/\delta a_i = C/a_i \qquad 12.15$$

so

$$Su = \sqrt{\{(C^2/a_i^2) \cdot Sa_i^2\}} = C \cdot Sa_i/a_i \qquad 12.16$$

Equation 12.15 shows that increase or decrease in error depends solely on the ratio of

$$C : a_i.$$

If

$$u = C \sin a_i$$

then

$$Su = C \cdot Sa_i \cdot \cos a_i \qquad 12.17$$

where Sa_i and a_i are in radians.

Simple example of estimating error propagation

Taking a simple example to highlight this analysis in action, a farmer may wish to estimate the uncertainty associated with the net returns from his wheat fields, knowing that the yields and the costs of management and harvesting vary spatially over the farm, and that there is also an uncertainty in the price he will receive. For each field, on a tonne per hectare basis, he wishes to evaluate the errors in his predictions of

$$\text{Net value } (N) = \text{yield } (Y) \times \text{price } (P) - \text{costs } (C) \quad 12.18$$

For each field, let

Y be 6 ± 2 t ha^{-1}

P 100 ± 10 currency units per tonne, and

C 40 ± 20 currency units ha^{-1}.

The gross value per hectare is:

$$\begin{aligned} G &= Y * P \\ &= 6 * 100 \\ &= 600 \text{ currency units ha}^{-1} \end{aligned} \quad 12.19$$

The uncertainty in the gross value is:

$$\begin{aligned} S_G &= \sqrt{\{P^2 . S_y^2 + Y^2 . S_P^2\}} \\ &= \sqrt{\{10\,000 * 4 + 36 * 100\}} \\ &= 116.62 \text{ currency units ha}^{-1} \end{aligned} \quad 12.20$$

The net value is

$$\begin{aligned} N &= G - C \\ &= 600 - 40 \\ &= 560 \text{ currency units ha}^{-1} \end{aligned} \quad 12.21$$

and the uncertainty is

$$\begin{aligned} S_N &= \sqrt{\{S_G^2 + S_C^2\}} \\ &= \sqrt{\{13\,600 + 400\}} \\ &= 118.32 \text{ currency units} \end{aligned} \quad 12.22$$

The methods given above can easily be worked out on a pocket calculator for single equations with only two or three variables, but the computations become tedious when they must be repeated for a large number of cells or entities in a database, or when numerical models involve larger numbers of attributes, including correlation. Various software packages are now available to understand and trace errors through complex numerical models in 'point mode' that operate on the attributes of entities or on multiple raster overlays—the ADAM system is one useful example (e.g. Heuvelink et al. 1989; Heuvelink and Burrough 1993).

Consider a multiple regression model in which the a_i values are raster maps of input attributes and the b_i values are the coefficients:

$$u = b_0 + b_1a_1 + b_2a_2 + \ldots + b_na_n \quad 12.23$$

Contributions to the error in output map u come from the errors associated with the model coefficient b_i and from the errors associated with the spatial variation and measurement errors (nugget; see Chapter 9) of the a_i input attributes.

ADAM uses this technique to apply the error propagation to each entity, which in the case of a raster map is the grid of pixels, but in other implementations could be polygons or other entities in vector mode. The regression model coefficients and their errors can be computed from sample data, and the same point samples can be used to create raster input maps of cell values and their errors. ADAM computes the model output and the errors as maps, and also as maps of the spatial variation of the error contribution from all coefficients and input variables.

12.5 Summary

This chapter has reviewed the fundamental components of computational modelling and the development of different approaches possible in a GIS environment. The growing use of the spatial analytical and display capabilities of GIS to support spatio-temporal modelling has been greatly facilitated by the transfer of data between systems using the widely adopted data exchange standards. Despite the benefits of GIS for data storage, retrieval, and display, modelling entirely within a GIS is still relatively limited by processing speed and functionality of GIS compared to specialist modelling software.

In the most widely used GIS, specialist raster modules are provided for the modelling but their functionality is still relatively limited for defining the complex processes of our natural and human environments.

The chapter also highlighted the importance of understanding error analysis and the impacts of uncertainty on modelling and analysis. These methods can also be applied to more general spatial data analysis. While inaccuracies can come through many different parts of the data-collection process, analysis, and in the model development, it is important that this is acknowledged and analysed so that the quality of the results is understood. Simple rules can be followed to reduce error propagation (Alonso 1968):

1. Optimize the data collection by maximizing the number of samples used and considering the impact of sampling methods.
2. Choose appropriate spatial interpolation methods.
3. Avoid inter-correlated variables.
4. Use addition functions in modelling where possible.
5. If you cannot add, then multiply or divide.
6. Avoid as far as possible taking differences or raising variables to powers.

The challenge now is to use this information in developing the best set of procedures and modelling tools to reduce the impact of uncertainty and inaccuracy. A really intelligent GIS would be able to carry out error propagation studies *before* a modelling operation is started, to estimate whether the methods and data chosen were likely to yield the results intended. It would report to the user where the major sources of error come from and would present him or her with a set of options which would achieve better results. This intelligent GIS has yet to be developed but may be an area of future effort, especially given the growing use of data collected by others in analysis and modelling exercises. Some possible areas where an intelligent GIS could help to reduce errors would be summarized as:

a) using optimal interpolation techniques
b) using the appropriate sampling density to collect data about a phenomenon
c) checking for outliers, subgroups, systematic bias, etc.
d) adopting appropriate classifications and class breakpoints
e) improving model calibration and reducing errors in input model parameters.

Questions

1. If you were to design a modelling system to understand the impacts of climate change on your home region, (a) what input data would you use, (b) what process modelling would you include, and (c) how would you present the results to local politicians?
2. How you design a GIS-based modelling system for managing disasters: (a) earthquake, (b) hurricane, (c) train collision?
3. How do you think the errors in the output of any given model will depend on (a) the uncertainties in the regression equation and (b) the spatial variation in the data?