

LINKING NUMERICAL LANDSCAPE
EVOLUTION MODELS WITH
HIGH-RESOLUTION
METEOROLOGICAL DATA

A THESIS SUBMITTED TO THE UNIVERSITY OF MANCHESTER
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

2017

Declan A. Valters

School of Earth, Atmospheric, and Environmental Sciences

Contents

Abstract	9
Declaration	10
Copyright Statement	11
Layperson's abstract	12
Acknowledgements	13
1 Introduction	14
2 Modelling landscape evolution	15
3 Meteorological data and models	16
3.1 Introduction	16
3.2 Rainfall radar - the UK 1km Radar Composite Product	16
3.3 Numerical Weather Prediction - the Weather Research and Forecasting model	17
4 Rainfall representation in current landscape evolution models	18
4.1 Introduction	18
4.2 Simple models and proxies for rainfall variation	19
4.2.1 1D models	19
4.2.2 2D models	22
4.3 Distributed models	23
4.3.1 Hydrological models	24
4.3.2 Landscape evolution models	27

4.3.3	Summary of current model capabilities	32
4.4	Research needs for landscape evolution modelling	33
4.4.1	Technological advances	34
5	Development of a numerical landscape evolution model for high-performance computing	36
5.1	Introduction	36
5.2	Model description and parallelisation	38
5.2.1	Origins: CAESAR-Lisflood	38
5.2.2	HAIL-CAESAR	38
5.2.3	Process representation	40
5.2.4	Parallelisation implementation	42
5.2.5	Potential speed up from parallelisation	47
5.3	Results	48
5.3.1	Regression testing	48
5.3.2	Performance comparison with CAESAR-Lisflood	49
5.3.3	HPC profiling and speed-up	51
5.3.4	Performance scaling on HPC compute nodes	51
5.3.5	Strong scaling	51
5.3.6	Weak scaling	53
5.3.7	Thread profiling	54
5.4	Discussion	55
5.4.1	Parallel scaling	55
5.4.2	Load balancing issues	56
5.4.3	Loop scheduling	57
5.5	Conclusions	58
6	Rain or rock? Sensitivity of a flood-inundation model to rainfall distribution and erosional parameterisation	64
6.1	Introduction	64
6.2	Experiment Design & Method	65
6.2.1	Study area	66
6.2.2	Hydro-Meteorological conditions	66

6.2.3	Numerical model configuration	68
6.3	Sensitivity analysis	69
6.4	Results	70
6.4.1	TOPMODEL sensitivity	70
6.4.2	Catchment hydrology	71
6.4.3	Spatial variation in flood inundation	72
6.5	Discussion	74
6.5.1	Implications for longer-term landscape evolution	75
6.6	Conclusions	76
6.7	Fixed parameters	76
7	Hydrometeorological controls on landscape erosion and evolution	79
7.1	Introduction	79
7.2	Method	82
7.3	Results	82
7.3.1	Catchment sediment flux	82
7.3.2	Spatial variations in sediment and bedrock erosion	82
7.4	Discussion	83
7.5	Conclusion	84
8	Conclusions	87
Bibliography		88
A	Linking high-resolution NWP model output with a landscape evolution model	95
A.1	Introduction	95
A.2	Modelling framework	96
A.3	Method	96
A.3.1	Modifications to the CAESAR-Lisflood model	96
A.3.2	Landscape evolution model set up	97
A.3.3	Modelling framework	97
A.4	Results	97
A.4.1	NWP modelling results	97

A.4.2	Landscape evolution/hydrology modelling results.	97
A.5	Discussion	97
B	Code availability	98
C	Key code and algorithms for the cellular automaton LEM	99
D	Configuration of the WRF model for the simulation of the North York Moors storm	100

Word count xxxxx

List of Tables

4.1	User defined parameters in Han and Gasparini's (2015) orographic rainfall model implemented in CHILD.	31
5.1	Summary of most expensive program functions, from a Boscastle 48 hour flood simulation	43
5.2	Summary of most computationally expensive program functions, from a Boscastle 48 hour flood simulation, with erosion processes enabled . .	43
5.3	Idealised potential speed-up according to Amdahl's law (equation 5.8) Value for P calculated using profiling results from the Boscastle test simulation after 48 hours simulated time.	48
5.4	Initial performance and compilation comparison with CAESAR-Lisflood, using the River Swale test case, 50m DEM, 10 year simulation	50
5.5	Strong scaling test cases used to benchmark the HAIL-CAESAR model, using Swaledale and Boscastle DEMs.	52
5.6	Weak scaling test cases with the Boscastle DEM resampled at increasing resolutions from 0.5m to 5m. The absence of run-time results for certain erosion simulations is due to memory limitations preventing higher resolution simulations from running.	54
6.1	Outline of the ensemble simulations carried for all case studies.	65
6.2	Table showing key characteristics of each storm event.	66
6.3	Start and end times (UTC) for each 72 simulation. The major rainfall event occurs during the 2nd day in each simulation.	68
6.4	TOPMODEL m parameter values used to run sensitivity simulations for each catchment.	70

List of Figures

5.1	Flow chart showing a simplified outline of the HAIL-CAESAR program flow. Grey shaded boxes indicate sections of the code parallelised with OpenMP. Rounded rectangles indicate output and input files.	59
5.2	Water discharge rates over first 250 days of simulation. Outputs from the Swale 10 year at 50m resolution test case, showing results from the original CAESAR-Lisflood model and HAIL-CAESAR model compiled under two different compilers.	60
5.3	Catchment sediment flux over first 250 days of simulation. Outputs from the Swale 10 year at 50m resolution test case, showing results from the original CAESAR-Lisflood model and HAIL-CAESAR model compiled under two different compilers.	60
5.4	Cumulative catchment sediment flux over full duration of simulation. Outputs from the Swale 10 year at 50m resolution test case, showing results from the original CAESAR-Lisflood model and HAIL-CAESAR model compiled under two different compilers.	61
5.5	Speed-up achieved relative to serial code on 2, 4, 8, 12, 16, 20, 24, 30, 36, 42, and 48 cores. Four sets of simulations are used representing a range of model uses from short, single event episodes (48–72hr) to 1 year simulations.	62
5.6	Weak scaling with the Boscastle DEM at increasing resolution (see Table 5.6). Hydrological and erosion-enabled simulations shown. Each simulation uses c.150 000 grid cells per CPU. The absence of results for certain erosion simulations is due to memory limitations preventing higher resolution simulations from running.	63

5.7 Thread profiling of the Swale 1 year simulation with 50m DEM. Data from major functions only displayed.	63
6.1 Discharge at Ryedale catchment outlet for varying values of the TOP-MODEL m parameter. The measured discharge at the catchment gauging station is overlain in dashed line. The results from the simulations with $m > 0.008$ are omitted for clarity due to the low peak discharges they produced.	72
6.2 Boscastle hydrographs (discharge over time at catchment outlet) for each simulation of the 2004 Boscastle event listed in Table 6.1. Inset shows detail of main flood peaks around hour 40 of the simulation.	73
6.3 Ryedale hydrographs (discharge over time at catchment outlet) for each simulation of the 2005 Ryedale event listed in Table 6.1.	74
6.4 Flood extents in the Boscastle catchment at the time of maximum river discharge for each simulation.	77
6.5 Flood extents in the Ryedale catchment at the time of maximum river discharge for each simulation.	78
7.1 Boscastle sediment flux (total sediment volume output per hour at catchment outlet for each erosion-enabled simulation of the 2004 Boscastle event listed in Table 6.1.	83
7.2 Ryedale sediment flux (Total sediment volume output per hour at catchment outlet) for each erosion-enabled simulation of the 2005 Ryedale event listed in Table 6.1.	84
7.3 Boscastle. Erosion.	85
7.4 Ryedale. Erosion.	86

The University of Manchester

Declan A. Valters

Doctor of Philosophy

Linking numerical landscape evolution models with high-resolution meteorological data

February 3, 2017

This thesis addresses a limitation in numerical models of landscape evolution regarding how the spatial variation of precipitation is represented or parameterised within such models. Numerical models of landscape evolution typically forsake a realistic representation of rainfall patterns in favour of a simpler treatments of rainfall as being spatially homogeneous across the model domain, despite the fact that many geomorphic processes being sensitive to thresholds of sediment entrainment and transport, driven by the movement of water within the landscape. The thesis presents the development and application of a series of computer-based models of landscape evolution with improved routines for the representation of rainfall spatial variability. These modifications are then applied to exploring the sensitivity of landscapes to spatial variation in rainfall inputs over a range of timescales from single storm events, to longer term topographic evolution.

The thesis starts by exploring current limitations in rainfall representation in landscape evolution models, followed by an assessment of precipitation data sources that could be potentially used as realistic inputs to landscape evolution models. Models of synthetic rainfall generation are also discussed in this section. A numerical model of landscape evolution is developed for deployment on high-performance parallel computing systems, based on the established CAESAR-Lisflood model (Coulthard et al., 2013). This model is benchmarked, showing performance benefits compared with the serial code. The improved model code is shown to produce results commensurate with the CAESAR-Lisflood model it is developed from.

The model is applied to assessing the sensitivity of fluvial erosional processes within river catchments to the spatial variation in rainfall during extreme storm events. Two real storm events that triggered flooding in the UK in 2004 and 2005 are used a test cases. Landscape response to single storm events is shown to be sensitive to the spatial resolution of radar rainfall input across the model domain.

Finally, a software framework is presented for driving a landscape evolution model with input from a numerical weather prediction model. Using high-resolution NWP simulations of 250m grid spacing, mesoscale rainfall features are resolved at the river catchment scale. The UK storm case studies described in the previous chapter are re-assessed using data from NWP simulations of each case to investigate the response of catchments to the mesoscale details of severe rainfall events.

Declaration

No portion of the work referred to in the thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright Statement

- i. The author of this thesis (including any appendices and/or schedules to this thesis) owns certain copyright or related rights in it (the “Copyright”) and s/he has given The University of Manchester certain rights to use such Copyright, including for administrative purposes.
- ii. Copies of this thesis, either in full or in extracts and whether in hard or electronic copy, may be made **only** in accordance with the Copyright, Designs and Patents Act 1988 (as amended) and regulations issued under it or, where appropriate, in accordance with licensing agreements which the University has from time to time. This page must form part of any such copies made.
- iii. The ownership of certain Copyright, patents, designs, trade marks and other intellectual property (the “Intellectual Property”) and any reproductions of copyright works in the thesis, for example graphs and tables (“Reproductions”), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property and/or Reproductions.
- iv. Further information on the conditions under which disclosure, publication and commercialisation of this thesis, the Copyright and any Intellectual Property and/or Reproductions described in it may take place is available in the University IP Policy (see <http://documents.manchester.ac.uk/DocuInfo.aspx?DocID=487>), in any relevant Thesis restriction declarations deposited in the University Library, The University Library’s regulations (see <http://www.manchester.ac.uk/library/aboutus/regulations>) and in The University’s Policy on Presentation of Theses.

Layperson's abstract

An optional section suggested by the UoM thesis preparation guide.

Acknowledgements

Writing this thesis has been an exercise in sustained suffering. The casual reader may, perhaps, exempt themselves from excessive guilt, but for those of you who have played the larger role in prolonging my agonies with your encouragement and support, well... you know who you are, and I hope you get what you deserve.

My suffering was funded by the Natural Environment Research Council doctoral training grant number NE/L501591/1.

1

Introduction

This chapter introduces the topic in a broad setting within the context of landscape evolution under different climatic conditions and meteorology (rainfall spatial patterns). The topic is discussed in its historic context as well briefly. It then goes on to clarify the topic of interest, and relates it to both topical research questions in landscape evolution theory and also the societal impact of intense, erosive rainfall events on the landscape.

2

Modelling landscape evolution

This is a general overview of landscape evolution models and the key concepts and limitations they have.

In the next chapter, Chapter 3, there is a more focused review of the specific area of Landscape evolution models that I am working on – i.e. rainfall representation in LEMs. I only touch upon rainfall parametrisation here.

This chapter could be based on the *Geomorphological Techniques* book chapter (Valters, 2016).

3

Meteorological data and models

3.1 Introduction

This chapter discusses the various sources of meteorological data and numerical models used to generate rainfall input to the landscape evolution models discussed in Chapters 2 and 4. In this thesis, two sources of rainfall data were used to drive the hydrological inputs of landscape evolution models: precipitation radar data from the UK 1km rainfall radar composite product (Section 3.2) and rainfall outputs from a numerical weather prediction model (Section 3.3)

3.2 Rainfall radar - the UK 1km Radar Composite Product

Rainfall radar are used to infer the spatial distribution and intensity of rainfall over a spatial range of up to several hundred kilometres. Electromagnetic radiation in the microwave spectrum is emitted in pulses through radar antenna that focuses them into a narrow directional beam. When the microwaves encounter hydrometeors (or other obstacles in the path of the beam), the reflected microwave beams are backscattered towards the radar dish. The location and intensity of precipitation can then be calculated using the time taken for the returned radar waves to reach the radar dish, and the amount of backscattered microwave radiation. Radar rainfall measurements are not direct measurements of rainfall, rather they are inferred by making a series

of assumptions of how the radar backscatter – the radar reflectivity – relates to the quantity and other characteristics of hydrometeors. The amount of radar reflectivity is determined by the size, shape, composition, and distribution of hydrometeors that are sampled by the focused radar beam. Radar reflectivity, denoted by Z , is related to the rainfall rate, R , by the formulation:

$$Z = aR^b \quad (3.1)$$

3.3 Numerical Weather Prediction - the Weather Research and Forecasting model

Numerical weather prediction models (NWP) are used to predict

4

Rainfall representation in current landscape evolution models

4.1 Introduction

This chapter reviews how current landscape evolution models represent rainfall input into the landscape system. It is worth stating here what exactly is meant by rainfall input, in the context of the atmosphere–land-surface system as represented in numerical models. Perhaps equally as importantly, it is worth discussing what aspects of rainfall are *not* represented at all in any numerical models of landscape evolution, to clarify how geomorphologists conceptually think of meteorological processes acting on the landscape.

For most purposes, rainfall input in landscape evolution models is simply the quantity of water added to a surface cell or node, or to the whole model domain. In practise, no numerical models represent rainfall in the sense of it actually falling from the sky and hitting the ground. While this may seem a somewhat trivial point, the impact of individual rain drops on the land surface is known to be an important contributor to surface erosion. Rain-splash erosion, as it is termed, is a well-studied phenomenon [cite a review of rainsplash erosion, if there is one?]. The interaction of raindrops with the surface is complex; it depends on the size of raindrop, falling velocity, angle of attack, soil exposure, soil mineralogy, and cohesion of the soil surface. All of these factors could affect both erosion on the landscape hillslopes, as well as the route that water takes to runoff and reach the rivers, before fluvial erosion can happen.

If we briefly turn to physical analogue models of landscapes, rainfall representation implicitly accounts for some of the above factors in rainsplash erosion and runoff, because of the physical need to generate a rainfall source from above the model, such as through a fine-meshed sprinkler [CITE]. In fact, geomorphologists using physical analogue models of landscape evolution attempt a degree of rainfall realism by ensuring the raindrops they generate are reasonably well scaled to the size of their landscape analogue (Meyer, 1994). By contrast, numerical models of landscape evolution begin their representation of rainfall input at the surface – in effect rainfall input in most landscape evolution models has nothing to do with *falling* rain or its impact on the ground. Conceptually, rainfall input in numerical models is the amount of water that would be added at the surface from one or more (usually many more!) raindrops, once they have reached the ground. It ignores any effects from the physical collision raindrops make with the ground. This simple conceptual model of rainfall input is used throughout the rest of this chapter when referring to rainfall input in landscape evolution models.

4.2 Simple models and proxies for rainfall variation

4.2.1 1D models

Isolated aspects of landscape evolution and hydrology can be studied using 1D models of features such as hillslopes profiles and longitudinal river profiles, or storm hydrographs in the case of hydrology. Though the work in this thesis focuses on 2D models, it is useful to consider the work done by others investigating the feedback from rainfall variability on 1D models of landscape evolution, before progressing to full 2- or 2.5D¹ models over an x, y model domain.

Roe et al. (2002) modify a simple 1D model for river profile evolution (Seidl and

¹Occasionally, the terms 2D and 2.5D are used interchangeably when referring to landscape evolution models, although in effect they both produce what looks like a ‘3D’ terrain surface from their output. The ‘third’ dimension (or extra 0.5D in 2.5D terminology) comes from the fact that the elevation variable can be used to reconstruct a 3D picture of the landscape based on the value for each grid cell or node. In practice, nearly all of the process models in landscape evolution models are 2D, e.g. water routing over the surface does not account for turbulent flow in x , y and z directions, such as in computational fluid dynamic models. Sediment transport does not account directly for 3D particle motion or collisions between particles. I use the term 2D landscape evolution model throughout the work.

Dietrich 1992; Howard et al., 1994; Whipple and Tucker, 1999) to incorporate a feedback for orographic precipitation based on changing elevation along a steepening river profile. Their precipitation feedback model accounts for two precipitation regimes: the first typical of midlatitude, shallower, and narrower mountain ranges such as the West coast of North America, and one for broader and taller ranges such as the Sierra Nevada, European Alps or the Southern Alps of New Zealand. The former represents rainfall patterns that are dominated by the prevailing upslope winds, increasing precipitation with distance upstream, whereas the latter represents environments where atmospheric moisture content exerts more control over precipitation, resulting in decreasing rainfall at higher elevations, and a rainfall shadow on the leeward side of the range. In a later work (Roe et al., 2003) the 1D model incorporating orographic rainfall feedback is extended to the 1D relief structure of mountain ranges. The maximum relief is found to be strongly dependent on the type of precipitation regime chosen - with the prevailing upslope wind regime favouring lower relief, symmetric mountain ranges, and the atmospheric moisture-limited regime favouring higher relief mountain ranges.

Further 1D models have been developed to determine the relative importance of rainfall variability compared to other boundary conditions, such as tectonic uplift or base level fall. The 1D river profile model of Wobus et al. (2009) uses a transport limited formulation of river profile evolution (Meyer-Peter and Muller, 1948) with a simple parameterisation of rainfall based on modifying the exponent to the discharge-area approximation given by:

$$q_w = k_q A^c \quad (4.1)$$

where q_w is the water discharge, k_q a dimensional coefficient, A the contributing drainage area, and c the exponent that relates which portions of the drainage basin contribute to gathering precipitation and converting it to water discharge. A decrease in c represents a shift to more rainfall being gathered in the upper reaches of the stream. An increase in c represents rainfall being gathered in the lower reaches. The situation where $c = 1$ implies rainfall input is equal along all sections of the river profile. The end result is perhaps intuitive – more rainfall input in the upper reaches of the stream (decrease in c) results in more incision in the headwaters. However, the study reveals

a key difference in the way that climatic and tectonic signals propagate along a river channel. Numerical results show that rainfall-driven perturbations propagate from the channel head downstream, whereas tectonic perturbations invariably propagate from base-level upwards towards the channel head. The authors, however, reach this conclusion without simulating the scenario where there is more contributing rainfall from the lower reaches, i.e. the value of c is higher. Given the setting of the study though, (streams draining a mountain front) it is perhaps reasonable to assume an increasing precipitation gradient upstream towards the mountain range.

In the one-dimensional cases discussed, there is a key limitation, which is often acknowledged by the authors. Channels profiles in 1D form are modelled with out their tributary streams. The main stem of the channel is assumed to be representative of the entire catchment as a whole. This implies that tributary channels, and hillslopes feeding the main channel, experience the same precipitation patterns, or that differences between the main channel and its contributing water sources can be ignored.

River channel profiles are not the only markers of landscape evolution, though they do dominate the range of 1D modelling studies investigating sensitivity to the spatial distribution of rainfall. Owen et al. (2010) address the sensitivity of hillslopes to average precipitation rates, although spatial variation of rainfall along hillslope profiles is not considered. The study reveals hillslopes are most sensitive to average precipitation rates when there is a lack of vegetation. Hillslope bedrock erosion decreases according to a power law as mean rainfall rates decrease, from semi-arid to hyperarid environments. In general though, the study of hillslope sensitivity to the spatial distribution of rainfall remains under-studied, particularly in the case of 1D profile evolution.

One-dimensional profile models are useful tools for exploring aspects of landscape evolution. By their definition though, they restrict studies of rainfall spatial variability to a single dimension along the landform profile. Rainfall spatial variability from tributary channels, or from runoff over hillslopes is lost, or ‘smeared-out’ (Roe et al., 2002). The effects of water routing within a drainage network are also lost, and interesting relationships between rainfall distribution, river network connectivity and erosion are potentially overlooked. Complex parameterisations of rainfall production

are often reduced to a single number or exponent in an equation describing the evolution of the landform profile of interest. Rainfall spatial patterns are often complex over correspondingly complex terrain, and only 2D models may suffice to fully explore the sensitivity of landscape process and form to rainfall spatial distribution.

4.2.2 2D models

Early 2D numerical models of landscape evolution were often driven by single process laws of fluvial incision, and the topography that resulted from them was a product of the parameters in the fluvial incision laws. Simple fluvial incision laws, implemented in 2D numerical models resulted in topography broadly similar to the fractal patterns of river networks observed in nature [CITE Ahnert/Turcotte], with the hillslope features between neighbouring river channels been formed by what was ‘left behind’ from fluvial incision patterns. In other words, separate process laws were not implemented to describe the typically diffusive processes observed in hillslope formation. [Roerring, Hurst, citations from Hurst]

A typical form of the simple stream power law for fluvial incision takes the form:

$$E = KA^m S^n \quad (4.2)$$

where K is termed the coefficient of erodibility, and is a catch-all term for climatic processes (amongst others) including the role of rainfall on the fluvial incision process. The K term itself could be considered a proxy for rainfall variation over time, assuming all other factors remained constant. [*are there studies that do this, I thought there were somewhere...not sure now?*]

Another simple model is the excess shear stress model for fluvial incision, where the incision or erosion rate, E is given as a function of shear stress, τ above a threshold level, τ_c :

$$E = k_e(\tau^a - \tau_c^a) \quad (4.3)$$

With this simple model of landscape evolution, one of the first studies to study the 2D evolution of topography under varying climatic conditions was that of Rinaldo (1993). The study implemented a cyclic variation through time on the parameter of

critical shear stress, the threshold for erosion, τ_c . Since shear stresses driving incision are determined by river discharge, which in turn is controlled by rainfall input, the cyclical variation in critical shear stress, τ_c can be used a proxy for temporal variations in rainfall over the catchment at geological timescales. When the value of τ_c is low during the model this effectively represents a period of high rainfall intensity, and when τ_c is high this represents a period of lower intensity rainfall (Rinaldo, 1993). In the resulting topographies from these simulations, drainage density and fractal dimension were shown to increase in response to a decrease in critical shear stress, or an increase in rainfall input over time, assuming other factors such as uplift remain constant.

Other studies to expand on:

- CHILD (Tucker) - Precipitation Stochastic Model.
- Colberg and Anders (2003)
- Solyom and Tucker (2004) - is this distributed or not? Probably not because rainfall-runoff is a parameterisation.
- Solyom and Tucker (2007)

4.3 Distributed models

Distributed models² are grid-cell based (or based on a grid of ‘nodes’) and allow certain variables to vary spatially across the model domain, from cell-to-cell or node-to-node. The term is less frequently used with regards to landscape evolution modelling, but is useful to distinguish those models which represent a spatial variability in meteorological input from those that treat it through a proxy variable or another parameterisation. There are comparatively few landscape evolution models that allow spatially variable rainfall input to be distributed across the model domain, and some of the examples discussed here are from purely hydrological models. However, the principal of modelling spatially distributed rainfall remains the same and there are potential applications in hydrological modelling that can be extended to landscape evolution purposes.

²I borrow the term ‘distributed model’ here from hydrological modellers.

4.3.1 Hydrological models

In the world of hydrological modelling, distributed rainfall inputs are more commonplace. A range of meteorological input data sources have been used to drive distributed hydrological models. Three main sources of spatial rainfall data commonly used are dense-network rainfall-gauge data, precipitation radar, and precipitation outputs from numerical weather prediction models. Each one of these sources has a range of merits and demerits as a raw data source, but the discussion here focuses on their suitability as spatially heterogeneous rainfall datasets for numerical landscape evolution models, rather than an appraisal of their relative accuracies in reporting precipitation distribution.

Precipitation data generated by numerical weather prediction models has been successfully used in distributed hydrological models to make hydrological forecasts, as well as to analyse historic flooding events. Hay et al. (2006) use the MM5 model (mesoscale meteorological model)³ to generate gridded rainfall data over a five-year period. The rainfall data is used to drive the PRMS distributed hydrological model – the Precipitation Runoff Modelling System – over a corresponding five-year period. The numerical weather prediction model is run at grid cell spacings of 20km, 5km, and 1.7km, the finest of which resolves individual valleys and massifs, and captures the resulting rainfall patterns over the catchment at high resolution. The study also compares the way that rainfall input zones in the hydrological model are represented. In the hydrological model, different zones of rainfall input can be defined along natural topographic boundaries, which are termed *Hydrological Response Units*. These rainfall zone units tend to follow sub-catchment boundaries within the main catchment watershed. Alternatively, the catchment can be divided up more simply into rainfall input zones corresponding to a regularly spaced grid at a cell-spacing that matches the resolution of the input data. In general, increasing rainfall input resolution in the Hay et al (2005) study results in a greater accuracy when compared with observed river discharge values. Using irregular-shaped hydrological response units based on natural sub-catchments, rather than a regular gridding of input data, results in better agreement with observation. However, as resolution increases towards the 1.7km grid-cell spacing, the difference seen from using irregular shaped hydrological response units

³A precursor to the Weather Research and Forecasting model, WRF.

and regular grids of comparable resolution decreases.

A study that uses high resolution numerical weather prediction model data to drive a hydrological model (Younger et al., 2007) tests the suitability of rainfall forecast data for making hydrological predictions and improving flood forecasting. High resolution (250m grid spacing) simulations using the United Kingdom Met Office Unified Model are used to generate input rainfall data to drive a TOPMODEL-based (Beven and Freer, 2001) hydrological model. The semi-distributed *Dynamic-TOPMODEL* hydrological model groups topographically similar regions of the catchment and calculates runoff-prediction for each of the these self-similar zones. The runoff calculation is then assigned to each node in that particular zone (see Beven, 2002, for a full explanation of the TOPMODEL concepts.) Computationally, this is more efficient than performing runoff calculations for every single grid cell in the catchment domain. The Younger et al. (2007) study considers two events, a summer convective rainfall-event and a winter stratiform rainfall event. Although the hydrological simulation using the dense-network of rainfall gauge data produced outputs more closely matched to discharge observations, simulations with the NWP rainfall forecast also produce accurate results. The authors highlight the potential of using high-resolution rainfall forecast data to improve flood-forecasting in the future, giving greater prediction lead-in times compared to nowcasting from rainfall radar or real-time raingauge measurements. Rainfall data from numerical weather prediction models lends itself well to use as input data for hydrological modelling; it is typically written in a gridded data output format, and if the user has control over both the generation of the NWP output as well as the hydrological or landscape evolution model, generating compatible data formats can be more straightforward.

A consensus has yet to emerge on whether distributed hydrological models are sensitive to the spatial distribution of rainfall input. Nictoina et al. (2008), in a study that assesses rainfall resolution in distributed hydrological models, note that several studies are in disagreement, even when comparing catchments of similar sizes and in similar environments. In terms of the peak discharge and the time to the peak from the onset of heavy rainfall during a flood, modelling rainfall input as a spatially heterogeneous boundary condition appears to have little impact on the predicted hydrographs. (Krajewski et al., 1991; Shah et al, 1996). It is noted that antecedent

conditions may determine some of the relative sensitivity in catchment hydrological response (Shah et al., 1996), but only when initial water saturation levels are low. The work by Shah, and that of Segond et al., (2007) indicate that variability in runoff production mechanisms are the dominant control on runoff response. Whether variability in rainfall heterogeneity also contributes to the runoff response depends on antecedent conditions, as catchments may be able to dampen spatial heterogeneities in rainfall (Segond et al., 2007). In the simulations run by Nicotina et al. (2008), the source of rainfall data is from a network of rain gauges. Rainfall resolution is varied by first interpolating the rain gauge data with inverse weighted kriging method to 100m resolution. The 100m resolution data is then upscaled to coarser grid-sizes of 10km and 50km, giving three sets of simulations. Their study uses two catchments of 1560km² and 8000km² in area. The authors select catchments of relatively large size compared to previous studies. Their choice of larger catchments is based on one their hypotheses being that smaller catchments are closer in size mesoscale rainfall features, and therefore less likely to experience truly heterogeneous spatial rainfall patterns. The results of the Nicotina study show small differences between flood hydrograph peaks, which is more pronounced for the larger (8000km²) catchment. A further set of simulations also compares a conservative upscaling of rainfall resolution to a non-conservative upscaling – i.e. the total volume of rainfall is not necessarily the same post-upscaling. The non-conservative upscaled rainfall resolutions display a greater difference in maximum flood discharge over the three rainfall resolutions than the conservative upscaling method. The authors assert that catchments are more sensitive to the total volume of precipitation than its spatial heterogeneity, although this is perhaps to be expected if the non-conservatively upscaled experiments simply add more water to the catchment at coarser rainfall resolutions. The authors' further experiments with different runoff-generation mechanisms show a much more marked sensitivity in hydrograph response, compared to rainfall spatial heterogeneity.

From a hydrological perspective, it would appear that getting the total rainfall volume and runoff-generating mechanisms accurately represented in a hydrological model are more important than the spatial pattern of rainfall (Gabellani et al., 2007; Nicotina et al., 2008). However, the approach of previous studies has been to focus primarily on the flood hydrograph during these simulations, which is essentially the water discharge

modelled (or measured) at a single point at the catchment outlet. Very few studies, if any, have properly addressed the 2D spatial extent of floodwaters in response to spatially variable rainfall inputs over a catchment. It seems an odd omission to investigate a boundary condition that is by definition spatially heterogeneous over three dimensions (the areal spatial pattern of a rainstorm, as well as the storm depth or intensity), and then to reduce the output to a modelled parameter at a single x, y coordinate on the model domain. This could be remedied in future research projects.

Intuitively, one might expect that in a river catchment system with its well defined boundaries and singular output point, that any mass-conserving model would produce similar results given water inputs of equal volume (here, I am excluding the non-conserving rainfall upscaling method used by Nicotina et al., 2008). The details of interest may lie in what goes on inside the model domain, rather than what comes out the outlet point. Nevertheless, the work done by the hydrological modelling community has laid some of the foundations for using spatially variable rainfall data in 2D landscape evolution models. A range of data input sources, and interpolation methods that have been successful in hydrological modelling. Some of the basic findings will also help guide the research in the later chapters, and development of an existing landscape evolution model in Chapter 4.

4.3.2 Landscape evolution models

Few of the currently available numerical landscape evolution models explicitly allow the user to vary the spatial distribution of rainfall across the model domain (Valters, 2016). At longer timescales, it can be argued that spatial variation in climatic conditions such as rainfall will eventually be averaged out over centuries and millennia, in effect negating any variation in rainfall spatial patterns (Solyom and Tucker, 2007; Tucker 2010). However, this assumption only holds true if we believe that storm location and rainfall patterns bear no relation to the underlying topography of a landscape or river catchment. In other words, the assumption is that on the short term there is no orographic influence, and on the longer term, that there is no link between evolving topography and evolving weather patterns in a region. Only in recent years, and in a select few studies, have geomorphologists begun to question this assumption. As interest in this question has grown, models have evolved to accommodate this feature.

At the short term end of the landscape modelling spectrum (days to centuries), the latest releases of the CAESAR-Lisflood model (Coulthard et al., 2014) now allow for spatially variable rainfall input data.

Coulthard and Skinner (2016) In a sensitivity study that systematically varied the rainfall input data spatial resolution, Coulthard and Skinner (2016) assessed landscape evolution model sensitivity in terms of sediment and water flux, and the spatial distribution of erosion in a mid-sized upland catchment (415km^2). Rainfall input data was sourced from precipitation radar, and rainfall data resolution is varied at 5km, 10km, 20km resolutions, as well as a ‘lumped’ input where rainfall is averaged spatially across the whole catchment. When the source data is upscaled to finer resolution, the total volume of rainfall is conserved (in contrast to the non-conserving upscaling methods used by Nicotina et al, 2008). The simulations are run with typical rainfall data that is extended over a 30 year period. Compared to the uniform (lumped) precipitation data, increasing the rainfall data grid resolution increases sediment flux from the catchment. In the case of the highest resolution rainfall simulation (5km), sediment flux increases by over 100% compared to the uniform rainfall case. Coulthard and Skinner’s study separates natural spatial variation in rainfall patterns by randomizing the rainfall cell ‘tiles’ from the precipitation radar data, in an attempt to remove any effects from orography in the catchment. In essence, their study is focused solely on the effects of rainfall data resolution alone, rather than the spatial patterns of rainfall in nature, which are often influenced by topography. The rainfall field randomising technique minimise biases from naturally occurring organisation in storm cells and orographic rainfall enhancement.

Von Ruette, et al (2014) So far in this chapter, the discussion has been on landscape evolution models and studies that focus on hydrological, fluvial, and hillslope erosional processes. Numerical models of whole-landscape evolution have a recognized bias towards temperate-humid landscapes (Pazzaglia, 2004; Tucker and Hancock, 2010; Valters, 2016) and tend to focus on a limited gamut of geomorphic processes: hydrology, fluvial erosion, hillslope evolution, and sediment transport. However, the sensitivity of other landscape processes may well be sensitive to the spatial distribution

of rainfall over a landscape. Landsliding is an often overlooked, yet important process in landscape evolution and frequently omitted in numerical models (Tucker and Hancock, 2010; Valters, 2016). Von Ruette et al. (2014) investigate the sensitivity of shallow landslide initiation to the spatial distribution of rainfall in a catchment, using a physical based catchment-scale landscape evolution model designed specifically for investigating landslide triggering, the *CHLT* model (von Ruette et al, 2013). In their modelling study, they examine the initiation of shallow landslides under spatially uniform rainfall and a coarse grid-based spatially variable rainfall input, from a real event occurring in 2002. The rainfall input data is a product of integrated rain gauge data and rainfall radar measurements. As the coarseness of the data is high relative to the size of the study catchment, the authors use an inverse distance weighting interpolation method⁴ to downscale the data to a 2.5m grid cell size, the same resolution as the digital elevation model data used in the study. The authors generate a further set of simulations with a set of artificial rainfall input grids at 500m grid cell size. In the model of landslide initiation in the authors' model, the main sensitivity is the rainfall intensity and the infiltration capacity of the soil. If rainfall intensity is too high, water will runoff before it can fully infiltrate the soil; there exists a sweet-spot where rainfall intensities are low enough that the soil will become saturated more readily, and more landslides will be initiated. In the simulations run with equivalent rainfall intensities, spacial heterogeneity exerts some control over the distribution of landslides, as certain grid cells experience high rainfall rates, whereas others experience lower rainfall rates, closer to the rainfall rate 'sweet-spot', and consequently more landslide initiation. The findings of the von Ruette (2014) study are complex; sensitivity of landsliding initiation to rainfall spatial heterogeneity is dependent on a number of other conditions such as soil moisture capacity, infiltration rate, rainfall rate, and rainfall intermittency. Rainfall spatial distribution in a catchment exerts a control on whether these conditions will be optimal for landslide initiation, since it controls local rainfall intensities. Von Ruette et al. conclude that both the spatial distribution of landslides and the total number of landslides triggered are sensitive to the spatial distribution of rainfall in a catchment, assuming other conditions such as infiltration capacity are near-uniform

⁴An interpolation that gives preferential weighting to points that are closer to each other. The measured values closest to the prediction point of interest have more weighting, which diminishes with distance from the point of prediction.

across the catchment.

Longer term landscape evolution

Solyom and Tucker (2007) Landscape evolution sensitivity to rainfall detail over much longer timescales, on the order of 100kyrs and greater, has been explored to a limited extent by a few studies. Solyom and Tucker (2007) investigate how limited storm size relative to the size and shape of the drainage basin, effects the evolution of landscape topography. In their model, storm cells are represented as circular patterns with peak rainfall intensities at the centre of the circle, decaying exponentially from the centre:

$$I = I_0 \exp(-L_s/L_0) \quad (4.4)$$

where I is the rainfall intensity at a given point in the storm cell, I_0 is the rainfall intensity in the centre of the storm, L_s is the distance from the centre of the storm to a given point in the storm cell and L_0 is a characteristic length scale associated with the spatial decline of rainfall intensity.

Orographic effects on rainfall enhancement are excluded in the model. In Solyom and Tucker's simulation, a set of idealised diamond-shaped catchments are varied in their elongation (length-width ratio), while being subjected to a steady non-uniform rainfall field described by the exponential decay function, centred at the middle of the diamond-shaped catchment. The exact implementation details in the model code is not revealed by the authors of the study. Their simulations reveal that in general non-uniform rainfall patterns introduce a catchment-shape sensitivity to rainfall-runoff production, which in theory should effect the size and distribution of geomorphic processes throughout the catchment as well. The authors do not present examples of topographies generated by the model, but instead show the total catchment discharge in non-dimensionalised form ($Q_p/A * I_0$) compared to non-dimensionalised catchment length (L/\sqrt{A} , where A is the catchment area). Their simulations indicate that the greatest sensitivity occurs when the size of the storm decline rate L_0 is about half of the catchment radius. Solyom and Tucker's interpretation of this is that if storm intensity declines very rapidly over space, i.e. the storm cell is small, then the majority of runoff production occurs in the vicinity of the storm cell, and is therefore insensitive to the

Parameter	Units
Initial cloud water column density	kg m^{-2}
Initial hydrometeor column density	kg m^{-2}
Time constant for conversion from cloud water to hydrometeors	seconds
Time constant for hydrometeor fallout	seconds
Wind speed	m s^{-1}
Mountain half width	metres

Table 4.1: User defined parameters in Han and Gasparini’s (2015) orographic rainfall model implemented in CHILD.

shape of the catchment (assuming the storm falls near the centre of the catchment.) If the storm intensity decline rate is small relative to the scale of the catchment then in contrast the catchment is relatively insensitive to catchment shape.

Han and Gasparini (2015) A more explicit look at the way topography is influenced by spatial variation in rainfall patterns is found in the recent work of Han and Gasparini (2015). Building on earlier work by Roe et al. (2004), who found the geometry of river long profiles to exhibit sensitivity to an orographic rainfall feedback mechanism, they explore the sensitivity of the whole landscape over a 2D domain. Modifying the CHILD landscape evolution model (Tucker et al., 2001), they develop a parameterisation scheme for orographic rainfall based on the model of Smith and Barstad (2004). In their implementation of Smith and Barstad’s model, the user controlled variables governing rainfall production are given in Table 3.1. The model offers considerable control over many meteorological variables determining orographic rainfall. In a series of simulations under differing rainfall conditions, the authors find only a slight sensitivity of the concavity of the main trunk channels under spatially variable rainfall. They conclude that channel concavity is not generally sensitive to orographic rainfall patterns, in contrast to the 1D profile model of Roe et al. (2002) which showed much greater sensitivity. The more revealing topographic metrics were found in planform study – both the hypsometric integral⁵ and the channel steepness index⁶ were found to be more strongly linked to the orographic rainfall gradient.

In the model domain, rainfall input values for each node are now calculated individually, rather than the uniform rainfall field used in standard versions of CHILD.

⁵A measure of the fraction of a catchment above a given elevation, describing the distribution of elevations over the catchment. See Brocklehurst and Whipple (2004); Cohen et al. (2008).

⁶A measure of channel steepness normalised to drainage area; See Wobus et al. (2006).

The calculation is based on a number of factors including the elevation of the current grid node, the direction of the prevailing wind, and factors relating to water content in the atmosphere. As the elevation of grid nodes can change as topography evolves throughout the simulation, and rainfall inputs depend on the elevation of each node, there is an explicit feedback mechanism between orographic precipitation and landscape evolution represented in the model.

4.3.3 Summary of current model capabilities

The capabilities of landscape evolution models have evolved in tandem with research needs in a piecemeal fashion. As climate change has become an important factor in driving research needs and interests, landscape evolution models have evolved themselves to cater for a range of climatic parameterisations at a range of time scales. Two-dimensional⁷ numerical models are increasingly used for forecasting and predictive purposes, as well as just answering theoretical research driven questions. Despite their potential however, 2D models of landscape evolution are only beginning to be developed to allow detailed spatial variation in many of the climatic variables, such as rainfall. This is seen in the CHILD landscape evolution model work of Han and Gasparini (2015) as well of the development of CAESAR-Lisflood (Coulthard and Skinner, 2016) to simulate spatially variable rainfall input fields.

Recent advances in landscape evolution modelling have coupled hydrological model components with the core erosional process modules to produce truly hydrodynamic models that do not assume steady state discharge. For example the CAESAR-Lisflood model (Coulthard et al., 2013), Landlab modelling framework (Tucker et al., 2015), and tRIBS model [CITE] all contain forms of distributed hydrological models to simulate the transfer of water as well as sediment between grid cells or nodes. At longer timescales, the meteorological processes representing rainfall over a landscape have been parameterised, though the detail of these parameterisation schemes can be quite sophisticated (e.g. Han and Gasparini, 2015).

⁷Or 2.5-dimensional, if the elevation variable is considered a limited 3rd dimension, in the sense that elevation can go up or down in landscape evolution models, though the underlying process representation remains restricted two-dimensions in the x,y plane. For example water flow and sediment transport is not fully realised in 3D in any current landscape evolution model.

4.4 Research needs for landscape evolution modelling

The sensitivity of landscape evolutionary processes to the spatial details of climate and precipitation is still relatively unexplored. Though the subject is more advanced in purely hydrological studies (Krajewski et al., 1991; Smith et al. 2005; Segond et al. 2007; Nicotina et al. 2008), there is still a lack of agreement on when sensitivities to rainfall heterogeneities become most pronounced, given the dependence on other aspects of catchment hydrology. The role of runoff generating mechanisms, the influence of vegetation, the influence of groundwater routing pathways, are all affected by the spatial distribution of rainfall in a catchment, yet further investigations into these competing factors are required to reach more consensus among the hydrological community. Though some authors claim there is an insensitivity of hydrological processes to rainfall heterogeneity over a catchment (Krajewski et al. 1991; Smith et al. 2005), a key difference in landscape evolution modelling is that many erosional processes are threshold dependent. When rainfall is uniformly applied across a catchment model, the shear stresses generated by water runoff and river discharge tend to follow a uniform distribution as well. Findings by Coulthard and Skinner (2016) find a pronounced sensitivity to rainfall data resolution in term of sediment flux from a catchment (upto 100% increases), in contrast to the relatively small differences observed in purely hydrological models (e.g. Nicotina et al. 2008). Apart from the Coulthard and Skinner (2016) paper, no other studies have been found that systematically explore landscape evolution model response to rainfall data resolution. Studies have yet to explore the effect of different spatial patterns of rainfall on the geomorphic impacts of single severe storms.

With regards to data sources for rainfall input into landscape evolution models, the most typical source is rainfall gauge data, for single sites or sparse networks across a catchment. Rainfall radar has also been explored as a potential source offering higher spatial resolution than most rain gauge data typically available (Coulthard and Skinner, 2016). Other potential sources include output from numerical weather prediction (NWP) models, or the use of artificial weather generators. These two sources offer the potential to explore a variety of different spatial patterns of rainfall

data, without having to source them directly from historic events. High resolution rainfall radar data only goes back [XX] number of years [CITE] for example. With methods using NWP models to simulate idealised weather conditions, or using weather generators, researchers have the potential to explore sensitivity to the spatial patterns of rainfall for a variety of meteorological conditions, and the potential to systematically explore different distributions of rainfall on landscape evolution.

There is still a great deal of unexplored ground for developing landscape evolution models beyond their current capabilities. Developments are needed to accommodate further types of spatially variable climatic input data and their interpolation (e.g. von Ruette et al, 2014; Coulthard and Skinner, 2016), to develop new feedback models between topography and rainfall generation (e.g. Han and Gasparini, 2015), new parameterisations of storm cell morphology (e.g. Solyom and Tucker, 2007), and to develop models to take advantage of high-performance computing facilities.

More on the research needs here...Perhaps an itemised summary of outstanding questions yet to be answered.

4.4.1 Technological advances

Landscape evolution modellers have in general been reluctant to take advantage of emerging technology or high performance computing systems to explore bigger problems, or to explore uncertainty in model output through ensemble simulations. By way of contrast, in fields such as meteorology, mineralogy, particle physics, and engineering, the use of high-performance compute facilities is commonplace. In part, this is due to many problems in landscape evolution modelling stemming from a lack of agreement over geomorphic process laws. There is still considerable uncertainty over which geomorphic ‘laws’ are best suited to represent certain natural processes, and the answer can be dependent on the environment being studied. As such, modelling simulations in landscape evolution have often focused on investigating the big-picture, broad-brushed questions about how landscapes evolve as a supplement to empirical field based studies. Geomorphologists, perhaps quite justifiably, have not yet required large-scale computing facilities used in other fields, for their questions can be answered satisfactorily with reduced complexity numerical models. This is especially true as a

large body of numerical landscape evolution modelling is used in an exploratory manner (Tucker et al, 2010, some other citations tyo go here...and in this paragraph) – geomorphologists have been accused by some (Hancock et al., 2003; Pelletier, 2015) of being satisfied simply if their modelled landscape "*looks about right...*" The era of purely qualitative geomorphology has long since passed, and new quantitative methods that can be applied such as the use of topographic metrics should be employed. Returning to the comparison with other fields and their use of high-performance computing (HPC), these fields often suffer uncertainty that geomorphology does in the choice of process law or parameterisation used in a numerical simulation. However, this does not stop them from judicious employment of HPC. In fact, one of the strengths of HPC facilities is the capability to assess many hundreds, if not thousands, of scenarios in ensemble simulations – addressing the uncertainty in process laws and model parameters that have been noted by others in the modelling field (Tucker and Hancock, 2010; Pelletier, 2015). A drive towards making use of high performance computing technologies is needed in geomorphology.

5

Development of a numerical landscape evolution model for high-performance computing

5.1 Introduction

Computer-based numerical models of landscapes have evolved since their introduction in the late 1970s, (e.g., Ahnert, 1976) to a point where they are now used to investigate a range of interacting processes in landscape systems including catchment hydrology, sediment transport, hillslope mass movement, and biological processes (Tucker and Hancock, 2010; Willgoose, 2005; Pazzaglia, 2003). Such models have developed over time to become useful tools not only for understanding how landscapes have formed in the past, but also how they will evolve in the future, such as in response to climatic or environmental change (Bras et al., 2003; Church, 2003; Pelletier et al., 2015). Landscape evolution models, as they are often collectively termed, enable catchment scientists, hydrologists, and geomorphologists to investigate landscape change on a range of timescales and spatial resolutions. They address societal needs to forecast the landscape's response to environmental change, as well as further the understanding of individual geomorphological processes and their interaction in forming whole landscapes (Dietrich et al., 2003).

The data driving landscape evolution models, such as the digital elevation models (DEMs) used to represent the landscape surface, and other inputs such as climatic

data, have increased rapidly in spatial resolution in recent years. (e.g., Gesch et al., 2002; Rabus et al., 2003; Tarolli et al., 2009; Casas et al., 2006; Krishnan et al., 2011) Topographic datasets have increased in resolution to sub-metre scales, particularly since the advent of terrestrial LiDAR-derived digital elevation models, which are now a common data source in geomorphological analyses and modelling studies (Bates et al., 2003; Clubb et al., 2014; Passalacqua et al., 2010). The growth of high resolution input data is a double-edged sword: it presents the numerical modeller with the opportunity of studying processes at finer scales, often allowing sub-grid processes such as channel morphology to be resolved at the grid-cell scale (Schoorl et al., 2000). However, higher-resolution data also increases the computational cost of model simulations, as the number of calculations at a given model timestep increases with increasing number of grid cells used to represent the model domain. Decreasing the grid cell size of an input dataset causes the total problem size – the number of grid cells in the model domain – to increase non-linearly. Rising demand to study landscape and catchment processes at a regional or even continental scale (Citation?) also increases the computational cost of a model simulation by increasing the total number of grid cells in a model domain. Process representation has also grown in complexity, with many LEMs now supporting a range of rainfall-runoff, flow routing, erosion and slope process laws in a single model (Coulthard, 2001; Tucker and Hancock, 2010; Hobley et al., 2016), further increasing the computational demands made by modelling studies.

Rapid growth in computing power has occurred in tandem with the computational demands made by the hydrological and landscape evolution modelling communities. Individual processor power has steadily increased in following with the predictions made by Moore’s law (Schaller, 1997; Moore, 1998), through a continued increase in the number of transistors on integrated circuitry. As the rate of increase in individual processor speed has slowed in recent years (Mann, 2000; Colwell, 2013), parallel processing – the use of multiple processors to tackle larger computational problems – has become a indispensable tool in the scientific computing community. The uptake of parallelisation methods by the numerical landscape evolution modelling community is still in its infancy (Valters, 2016). However, disciplines with a close affinity to geomorphology have explored the use of parallel computing technologies in order to study larger problem sets, such as in the hydraulic modelling community (e.g., Ivanov et al.,

2004; Neal et al., 2009; Kollet et al., 2010; Smith and Liang, 2013; Liang and Smith, 2015; Smith et al., 2015). The increasing demand for environmental model codes capable of exploiting the parallelism offered by high-performance computing technologies has lead to the development of the HAIL-CAESAR landscape evolution model.

This paper presents an implementation of the CAESAR-Lisflood model (Coulthard et al., 2013), re-written and adapted to be compatible on for shared-memory parallel computing environments. The version presented here, termed HAIL-CAESAR¹, provides an operating-system independent, open source, hydrodynamic landscape evolution model suitable for deployment on various computing environments, including high performance computing architectures.

5.2 Model description and parallelisation

5.2.1 Origins: CAESAR-Lisflood

The porting and modification of HAIL-CAESAR is based on the CAESAR-Lisflood model (Coulthard et al., 2013), a cellular automaton numerical model of landscape evolution integrated with a hydrodynamic flow model (Bates et al., 2010). CAESAR-Lisflood is an open-source, GUI-based landscape evolution model written in the C# programming language, and distributed as a WindowsTM executable file. As of version 1.9b, there was no platform-independent version of the code, and users are required to run the model in GUI-mode on a Windows-based desktop computer. The model was therefore unsuitable for running on Unix-based operating systems, such as those typically found on cluster computing facilities. High performance computing systems, beyond those of typical desktop computers, could not be taken advantage of to run computationally expensive simulations.

5.2.2 HAIL-CAESAR

HAIL-CAESAR, much like the original CAESAR-Lisflood model it is based upon, is a cellular automaton, landscape evolution model for simulating hydrological and sediment transport processes at the river catchment scale. The key components of

¹High-performance Architecture Independent Lisflood-CAESAR model.

the model are a hydrodynamic water flow-routing model, and a sediment erosion and transport model. The model is designed to simulate catchment processes on timescales of hours, years, and hundreds of years. The HAIL-CAESAR model is an open source, platform-independent C++ implementation of the algorithms in the CAESAR-Lisflood model described in the next section. The model is run from a command line or terminal interface, with the user supplying a parameter file that initialises the variables within the model, and specifies the supplementary input files such as terrain DEM and rainfall input data. The user controls the majority of the model's operation by changing the parameter file variables. An outline of the program flow, file inputs and outputs is shown in Figure 5.1.

Object-oriented framework The HAIL-CAESAR model is designed within an object-oriented framework, enabling more advanced users to make modifications to the general model structure (Figure 5.1) by modifying the supplied *HAIL-CAESAR-driver.cpp* file. The modular approach to the model's functionality allows advanced users to create their own custom versions of the model in a structured way using object-oriented principles. The model is also integrated with the LSDTopoTools topographic analysis framework, a C++ software package for the analysis and modelling of landscapes using raster input data. In the object-oriented framework, model simulations are created as object instances, and then methods can be called upon the model object, for example:

```
LSDCatchmentModel mySim("parameters.txt")
// Create an instance of the model
// using an input parameter file.

mySim.loaddata();
mySim.water_inputs();
mySim.depth_update();
mySim.flow_route();
mySim.erode();
```

The flexibility of the object-oriented framework allows the user finer control over the complexity of the simulation in terms of process representation, adding and removing landscape processes as necessary. This modularity in landscape evolution modelling codes is also seen in the CHILD (Tucker et al., 2001b) and Landlab (Hobley et al., 2016) modelling frameworks.

5.2.3 Process representation

Catchment hydrology

Runoff from rainfall inputs to the catchment is generated using an adaptation of the TOPMODEL hydrological model (Beven and Kirkby, 1979). The TOPMODEL approach first calculates a combined surface and subsurface discharge for the cells within a ‘wetted’ zone of the catchment (Coulthard et al., 2002). When local rainfall rate is greater than zero, total runoff, Q_{tot} , is given by

$$Q_{tot} = \frac{m}{T} \log \left(\frac{(r - j_t) + \exp \left(\frac{rT}{m} \right)}{r} \right) \quad (5.1)$$

where m is a parameter that controls the rise and fall of the soil moisture store, j_t , T is the time step in seconds, and r the rainfall rate in metres per hour. The soil moisture store, j_t , is given by the formula

$$j_t = \frac{r}{\left(\frac{r-j_{t-1}}{j_{t-1}} \exp \left(\left(\frac{(0-r)T}{m} \right) + 1 \right) \right)} \quad (5.2)$$

When rainfall rate is zero during the current iteration, the total runoff, Q_{tot} , is given by the equation:

$$Q_{tot} = \frac{m}{T} \log \left(1 + \left(\frac{j_t T}{m} \right) \right) \quad (5.3)$$

with the soil moisture store, j_t , given by:

$$j_t = \frac{j_{t-1}}{1 + \left(\frac{j_{t-1} T}{m} \right)} \quad (5.4)$$

Total runoff, Q_{tot} , is apportioned between surface and subsurface discharge by a user-set threshold for discharge, Q_{min} . When the volume of water for a given cell

exceeds this threshold, it is treated as surface runoff and routed according to the flow routing algorithm described in the following section.

Surface flow routing

Surface water and channel flow is an important driver of catchment scale erosional processes. The amount and velocity of water flow is a variable in both the sediment transport and bedrock erosion laws. The surface water flow equations are based on a simplified form of the shallow water flow equations, a simplification first derived by Bates et al. (2010) and later incorporated into the CAESAR-Lisflood landscape evolution model by (Coulthard et al., 2013). The flow between cells is calculated by:

$$Q = \frac{q - gh_{flow}\Delta T \frac{\Delta(h+z)}{\Delta x}}{1 + gh_{flow}\Delta tn^2|q|/h_{flow}^{10/3}} \Delta x \quad (5.5)$$

Sediment transport and erosion

Transport of loose sediment is governed by the Wilcock and Crowe (2003) sediment transport model. The Wilcock and Crowe model represents transport of mixed sand/gravel fractions based on the surface sediment composition. The rate of sediment transport, q_i , is given as:

$$q_i = \frac{F_i U_*^3 W_i^*}{(s-1)g} \quad (5.6)$$

where F_i is the fractional volume of sediment, for a given sediment fraction, i , U^* is the shear velocity, s is the ratio of sediment to water density. W_i^* is a function relating fractional transport rate to total transport rate (see Wilcock and Crowe, 2003, for a full derivation of this equation). The usage of this sediment transport model is extrapolated here to account for finer particles such as silts (Van De Wiel et al., 2007), as well as the sand-gravel mixture it was originally designed for.

Bedrock incision

A simple model of bedrock incision based on the excess shear stress model (Tucker et al., 2001a; Tucker, 2004) is implemented in the numerical model. The rate of bedrock incision is determined by the amount of shear stress acting on the bedrock, above a

threshold level of stress required to initiate substrate removal (e.g., Snyder, 2003). When bedrock material is removed, it is distributed amongst the sediment fractions according to the fractional proportions set by the user. The rate of bedrock erosion according to the excess shear stress model is given by:

$$\varepsilon = k_e(\tau_b - \tau_c)^{P_b} \quad (5.7)$$

where k_e is the bedrock erodibility coefficient, τ_b is the basal shear stress on the channel bed, τ_c , is the critical shear stress threshold, P_b is the shear stress exponent (Howard and Kerby, 1983; Whipple and Tucker, 1999).

5.2.4 Parallelisation implementation

The HAIL-CAESAR code is parallelised using a shared-memory parallelisation model. In brief, the shared-memory technique works by distributing the processing load to multiple processing units that all have access to the same memory address space. The HAIL-CAESAR code uses the OpenMP application programming interface (API), which is widely supported on a range of software platforms and computing architectures (Dagum and Menon, 1998). Shared-memory parallel codes such as the one described in this paper are suitable for any computing system where the physical processors have access to the same memory space. A wide range of systems can avail of shared-memory parallel codes, from multi-core desktop computers, through high-end multi-processor, multi-core workstations, to individual compute nodes on high performance computing services (HPC). The code presented has been tested on a wide range of architectures including desktop computers, small-scale cluster computing facilities, and national scale HPC services.

The main functions in the program that are parallelisable, and most computationally expensive, are the water routing algorithm (*flow_route*), the erosion routines (*erode*), the water depth update function (*depth_update*), and the catchment wetted-area scanning function (*scan_area*). These were identified by profiling the serial version of the code using the *gprof* profiling tool (Graham et al., 1982), and the results are shown in Tables 5.1 and 5.2.

Table 5.1: Summary of most expensive program functions, from a Boscastle 48 hour flood simulation

Time (percentage of total)	Function name
77.71	flow_route
17.63	depth_update
3.58	scan_area
1.02	catchment_waterinputs
0.03	water_flux_out

Table 5.2: Summary of most computationally expensive program functions, from a Boscastle 48 hour flood simulation, with erosion processes enabled

Time (percentage of total)	Function name
37.96	erode
25.68	flow_route
15.90	scan_area
11.47	depth_update
3.75	(matrix library functions)
1.84	d50
1.63	slide_GS
0.59	sort_active
0.46	catchment_waterinputs
0.20	sand_fraction

flow_route The *flow_route* function is the core water-routing algorithm based on the Bates et al. (2010) algorithm. The *flow_route* function accounts for c. 26% of compute time in a typical erosion-enabled simulation, and 78% of compute time in a flow-only simulation. As one of the most compute-intensive sections of code when the catchment is in flood, the parallelisation of this section achieved significant overall code speed up for a variety of data inputs. The flow routing algorithm is only performed on those cells in the model domain that have accumulated a water depth, i.e. it is not performed in ‘dry’ cells. This means that only a small subset of cells are accessed during the flow routing section of the code, given a typical scenario where there is significant amounts of flow in the channels and floodplain areas, but little or none on the hillslopes. The code and parallelisation of the *flow_route* function is given in outline form below:

```
#pragma omp parallel for
    schedule(runtime)
for (int y=1; y<=jmax; y++)
{
    int inc = 1;
    while (down_scan[y][inc] > 0)
    {
        // Water routing in x direction...
        // Water routing in y direction...
        inc++;
    }
}
```

depth_update Profiling of the serial code identified the water depth update function as being one of the most compute intensive parts of the code for hydrology-only and erosion-enabled simulations. Updating of water depths is done using the same scanning algorithm as described for *flow_route*, updating depths in cells where there is water, or in neighbouring cells to water-containing cells. An outline of the implementation is presented below:

```
#pragma omp parallel for
```

```

reduction(max:l_maxdepth)
schedule(runtime)

for (unsigned y = 1; y<= jmax; y++)
{
    int inc = 1;
    double tempmaxdepth = 0;
    while (down_scan[y][inc] > 0)
    {
        // Update water depths
        // Update suspended sediment
        // concentrations
        // Calculate maximum flow depth
        ...
        inc++;
    }
}

```

scan_area The *scan_area* function analyses the catchment to determine which cells contain water or neighbour water-containing cells and sets an index array based on the current wetted area of the catchment. Further functions that involve updating sediment or water transport amounts then use this array as a mask so that only catchment model cells that contain water will be inspected and have their sediment and water totals updated.

```

#pragma omp parallel for
for (int j=1; j <= jmax; j++)
{
    int inc = 1;
    for (int i=1; i <= imax; i++)
    {
        // zero down_scan array
        down_scan[j][i] = 0;
        // and work out scanned area.
    }
}

```

```

    if (water_depth[i][j] > 0
        || water_depth[i][j - 1] > 0
        || water_depth[i][j + 1] > 0
        || water_depth[i - 1][j] > 0
        || water_depth[i - 1][j - 1] > 0
        || water_depth[i - 1][j + 1] > 0
        || water_depth[i + 1][j - 1] > 0
        || water_depth[i + 1][j + 1] > 0
        || water_depth[i + 1][j] > 0
    )
    {
        down_scan[j][inc] = i;
        inc++;
    }
}
}

```

erode The most compute-intensive part of the code when run in erosion-enabled mode is the *erode* function. This function performs all sediment entrainment and transport routines. The serial-run test simulation spent c. 38% of its time in this function, plus an additional 5% of time in function calls within the main *erode* routine.

```

#pragma omp parallel for
reduction(max:tempbedloadmax)
schedule(runtime)

for (unsigned int y = 1; y < jmax; ++y)
{
    int inc = 1;
    while (down_scan[y][inc] > 0)
    {
        unsigned x = down_scan[y][inc];
        inc++;
        // Calculate sediment entrainment
    }
}

```

```

    ...
}

}

#pragma omp parallel for
{
    // Sediment transport in x-direction
}

#pragma omp parallel for
{
    // sediment transport in y-direction
}

#pragma omp parallel for
{
    // Calculate sediment transport
    // from all 4 edges
    // of the model domain
}

```

5.2.5 Potential speed up from parallelisation

The functions described above, *flow_route*, *depth_update*, *scan_area*, and *erode*, account for 98% and 94% of the compute-time in hydrology-only and erosion-enabled modes, respectively, using the 48 hour Boscastle simulation as a test case. Therefore, effective parallelisation of these functions should result in effective parallel speed-up. The potential speed-up as a result of parallelisation can be estimated using Amdahl's law (Amdahl, 1967). The law gives the predicted speed up on N processing units based on ideal scaling over multiple cores. Ideal parallel speed-up, S is given by:

Table 5.3: Idealised potential speed-up according to Amdahl’s law (equation 5.8) Value for P calculated using profiling results from the Boscastle test simulation after 48 hours simulated time.

Number of CPUs	Speed-up (Hydro)	Speed-up (Erosion)
	$P = 0.98$	$P = 0.94$
2	1.94	1.89
4	3.67	3.39
8	6.11	5.63
16	11.03	8.42
32	16.58	11.2
48	19.92	13.4

$$S = \frac{1}{(1 - P) + (P/N)} \quad (5.8)$$

Where P is the proportion of program time that can be run in parallel. (Note that P may vary slightly between simulations, based on the input data and certain model parameters supplied to the program.) Profiling the program with the Intel VTune amplifier performance analysis tool suggests that the program spends 94% of its time in parallel sections of the code during an erosion-enabled simulation and 98% of compute-time in parallel sections during a hydrology-only simulation. Using Amdahl’s law it is possible to calculate theoretical potential speed-up given in Table 5.3. At the maximum available processor count of 48 cores, speed ups of up to c.20 times are predicted for a hydrology-only simulation, and up to c.13 times for an erosion-enabled simulation. The limitations of Amdahl’s law are that it predicts only idealised speed ups, and does not account for any overheads in parallelisation, such as the creation and synchronisation of threads, nor does it account for performance issues due to the speed of memory access in memory-bound computational problems (Hill and Marty, 2008; Sun and Chen, 2010).

5.3 Results

5.3.1 Regression testing

Regression testing is a software development practice used to verify that software that has been modified, interfaced with other software, or re-implemented, still performs as

expected when compared to the original implementation (Wong et al., 1997). To verify that HAIL-CAESAR produces comparable results to the original CAESAR-Lisflood model, we perform a set of simulations with the same test data and input parameters, for both implementations. We compare the catchment hydrological and sediment outputs from a 10-year simulation of the River Swale, North Yorkshire, United Kingdom. The Swale River test case has been well calibrated in previous studies (e.g., Coulthard, 2001) and is supplied with the original CAESAR-Lisflood model as a standard test case. Details of the model testing simulations are shown in Table 5.4. The HAIL-CAESAR model is tested with two different compilers, to assess any potential differences arising from the choice of compiler. Three metrics are used to compare the two implementations: water discharge rate from the catchment outlet, hourly sediment output, and cumulative sediment output from the catchment over the course of the simulation. Figure 5.2 shows the differences in water discharge using the two implementations. (For clarity, only the first 250 days are shown.) The LISFLOOD implementation performs similarly in both models, with the largest discrepancies seen in the first 100 days. An inset in Figure 5.2 shows the typical magnitude and timing of differences in the discharge. The timing and magnitude of the largest flood peaks are comparable, and the largest differences in magnitude on the order of $1\text{--}10\ m^3 s^{-1}$. Figure 5.3 shows the sediment flux from the catchment outlet. While there is low-magnitude variation in the sediment flux signal, the main peaks in sediment discharge are at similar times and of similar magnitude in both implementations. Figure 5.4 shows the cumulative sediment output from the catchment. Initially, both implementations show close agreement, until approximately 700 days into the test simulation where the predicted sediment totals begin to diverge. The CAESAR-Lisflood implementation, in this case, predicts an overall lower total sediment discharge over the 10-year period. The HAIL-CAESAR implementation predicts c. 40% greater total sediment yields after 10 years.

5.3.2 Performance comparison with CAESAR-Lisflood

Though the code base of HAIL-CAESAR and CAESAR-Lisflood differ somewhat, due to differences between the C# and C++ languages, as well as the parallelisation libraries used, the underlying algorithms remain broadly the same. The two models

Model Version	CAESAR-Lisflood 1.9b	HAIL-CAESAR v1.0	
Runtime (hrs:mins)	7:00	4:09	3:39
Programming Language	C#	C++	C++
Compiler	MSVC 14.0	GCC 6.2	Intel v17.0
Optimisation flags	-optimize	-O3, -march=native	-O3, -march=native
Parallel library	C# native parallel library	OpenMP 4.5	OpenMP 4.0
Processor	Intel i7-3770 8 core @ 3.4GHz, 32GiB memory		

Table 5.4: Initial performance and compilation comparison with CAESAR-Lisflood, using the River Swale test case, 50m DEM, 10 year simulation

were compared on similar hardware to assess any gains in speed-up from porting the CAESAR-Lisflood model to a C++ implementation using the OpenMP parallelisation libraries.

The hardware used for this test study was a workstation computer with an Intel i7 8-core processor and 32GiB of memory. The CAESAR-Lisflood test simulations were run on a Windows operating system, as per the requirements of the software; the HAIL-CAESAR model test simulations were run on a Linux operating system on the same machine. The input parameters and input DEM were kept the same for all benchmarking simulations. The details of the test case configurations are shown in Table 5.4.

Using the HAIL-CAESAR implementation, a speed up of c.41% was seen using the open-source GNU C++ compiler (g++) and a speed-up of 48% using the Intel C++ compiler. As the core algorithms remain the same in both versions of the code, the speed up likely comes from the fewer overheads in the command-line based HAIL-CAESAR model, and potentially through better compiler optimisations available with the C++ compilers. Further speed up may come from the fact that the CAESAR-Lisflood model is executed through a *just-in-time* compiler, translating pre-compiled bytecode into machine code at runtime (Aycock, 2003). In contrast, HAIL-CAESAR runs using a compiled binary executable file, with machine code generated at compile time. The HAIL-CAESAR approach avoids the potential overheads of invoking the dynamic compilation stage in C#/.NET programs.

5.3.3 HPC profiling and speed-up

5.3.4 Performance scaling on HPC compute nodes

Our primary aim in developing this software was to have a code that ran on Unix-based high performance computing systems and could be re-compiled for a variety of different platforms and computer architectures. This section demonstrates the scaling potential and optimal hardware configurations using a single compute node on a typical HPC system. Test simulations are run on the ARCHER supercomputer. Each compute node consists of two Intel Ivybridge Xeon processors, each with 12 cores per physical processor. Simultaneous multi-threading allows each core on the processor to effectively act as if it were two separate processing units, giving a total of 48 possible threads/cores per compute node. A single compute node consists of two NUMA²-regions each with access to 16GiB of memory, a total of 32GiB per compute node.

5.3.5 Strong scaling

Strong scaling describes how the performance increase of the code scales as more processors are used to solve a problem of fixed size. We test two typical use case scenarios for the HAIL-CAESAR model. The first case is an extension of the standard test case using the Swale data. The use of the model in this case is typical of longer duration landscape evolution simulations, with periods of predominantly low hydrological flows interspersed with brief (relative to the simulation time) storm events. The second case uses the HAIL-CAESAR model to simulate a single storm event at higher spatial resolutions, over a much shorter timescale. Two contrasting model applications are chosen to represent the range of timescales the model can be applied to, from short term single storm events to annual and decadal simulations.

Multi-event simulation, 1 year The Swaledale test case is run over a 1 year duration, using an input DEM of 50m resolution. The model is run in hydrology-only and erosion-enabled modes. The input parameters and input DEM are the same for each simulation, as each simulation is repeated on an increasing number of processors

²Non-uniform memory access

Table 5.5: Strong scaling test cases used to benchmark the HAIL-CAESAR model, using Swaledale and Boscastle DEMs.

Strong scaling benchmarking simulations				
Catchment	Swaledale		Boscastle	
Grid cells	124931	720000	4500000	
DEM cell size	50m	5m	5m	2m
Simulation time	8670 hrs	48 hrs	72 hrs	72 hrs
Catchment Size	150km ²	12km ²	12km ²	12km ²
Number of cores	1, 2, 4, 8, 16, 20, 30, 36, 40, 48			

(Table 5.5). At the maximum core count of 48 core, results from this simulation show speed up of up to 700% and 550%, for erosion-enabled and hydrology-only modes, respectively. The speed-up shown for erosion-enabled simulations does not always increase in concert as core-counts are increased, although the overall trend is one of increasing speed-up. For example, erosion-enabled simulations with 16, 24, and 36 processors showed a slow down when compared to the preceding benchmark tests using 12, 20, and 30 processors, respectively (Figure 5.5). The hydrological-only simulations showed a more consistent speed-up between benchmark tests (Fig. 5.5), though speed-up potential begins to decline at thread counts around 30 threads and higher, where it is observed the gains from increasing the thread count further diminish.

Single event simulation, 48–72 hours For this benchmark, two sets of strong scaling experiments are done with two different resolution DEMs. For each set of experiments, the input terrain data and parameters remain the same while increasing the number of processors over which the problem is shared. The strong scaling benchmark tests use a model domain with a) 720 000 grid cells, and b) 4 500 000 grid cells, representing a 12km² river catchment with 5m and 2m resolution DEMs, respectively. At this resolution of DEM, it is possible to resolve the larger channel geometries at the grid-scale, without resorting to techniques of ‘burning in’ the DEM with a channel. Sub-grid parameterisation of the channel shape, a feature of some models, is also not required.

Since HAIL-CAESAR features both a flood-inundation-only and sediment erosion mode, two further sets of benchmarks were done for both problem sizes, one with the model running in hydrology-only mode, and another running in the erosion-enabled

mode.

The results from the single-event scaling benchmark show the greatest speed-up for the 48 hour Boscastle simulation, with a speed-up of almost 16 times at 48 threads/cores, for the hydrology-only simulation. Erosion-enabled simulations also show good speed-up for this problem size (720 000 grid cells), but the linear speed-up trend is lost above around 30 cores, when speed-up gains begin to tail-off. At the larger problem size of 4 500 000 grid cells, the speed-up gains from increasing core counts is much diminished, though still increases linearly through higher core counts. Speed-up here reaches only c. 5 times running on 48 cores, in hydrology-only mode, and 3.5 times in erosion-enabled mode.

5.3.6 Weak scaling

Weak scaling is a parallel performance metric used to assess how the code runtime scales as the amount of work per processor is kept the same, but the total problem size – in our case, the number of grid cells in the model domain – is increased (Gustafson, 1988). This mirrors the practice of using larger numbers of processors to tackle model simulations using higher resolution topographic data, or data covering a larger spatial domain. The weak scaling experiments used a set of hydrology-only simulations and a set of erosion-enabled simulations. A series of 72 hour simulations were done using a range of high-resolution topographic datasets, from 0.5m to 5m resolution, scaling the number of processors used to maintain the ratio of processors/workload as near as possible. The number of model domain grid cells per core was maintained at approximately 1 500 000. The simulations were repeated for hydrological-mode and erosion-enabled modes, but due to the high memory demands of running very high resolution erosion-enabled simulations, not all simulations could be completed on the available hardware, which has a maximum job runtime of 48 hours.

The simulations in hydrological-only mode show positive weak scaling as the problem size increases in tandem with the number of threads (Figure 5.6). Ideal weak scaling would be expected to show a runtime per number of grid cells/cores remaining approximately constant compared with a single-threaded simulation. The slight increase seen in the weak scaling here is likely due to the overheads experienced with creating and synchronising the extra threads as the problem size is increased.

Table 5.6: Weak scaling test cases with the Boscastle DEM resampled at increasing resolutions from 0.5m to 5m. The absence of run-time results for certain erosion simulations is due to memory limitations preventing higher resolution simulations from running.

Weak scaling simulations, Boscastle DEM, 72 hour simulation						
DEM resolution (m)	Grid cells	Ideal no. of threads	No. of threads	Cells per thread	Run time (Hydro)	Run time (Erosion)
0.5	72 000 000	48	48	1 500 000	323.1	—
0.6	49 900 050	33.3	33	1 512 122	190.4	—
0.7	36 808 200	24.5	26	1 472 328	138.8	—
1.0	18 000 000	12	12	1 500 000	84.3	355.2
1.5	7 960 050	5.3	5	1 592 010	49.0	—
2.0	4 500 000	3	3	1 500 000	568.9	1112.1
5.0	720 000	0.48	1	720 000	347.306	432.3

5.3.7 Thread profiling

Further profiling of the parallelised version of the code was carried out using the Intel® VTune Amplifier performance analyser. Thread profiling allows us to see where the bottlenecks in the code area, including functions that are compute intensive, but also where inefficiencies occur due to overheads from parallelisation and load imbalance. For clarity, a single example using the Boscastle test case is presented, with a simulated model time of 48 hours, running on 8 cores/threads. It is expected that different simulations and domain sizes would produce slightly different results, but this simulation gives a general idea of the parallel performance of specific functions in the code.

The break down of major function overheads is shown in Figure 5.7. Only four of the key parallel-implemented functions, *erode*, *flow_route*, *scan_area*, and *depth_update* are assessed, as other functions collectively amounted to a small proportion of the runtime, or were serial functions. Load imbalance accounts for 22% of the execution time of these four key functions, due to certain threads sitting idle while waiting for other threads to complete. Load imbalance is particularly apparent in the *erode* and *depth_update* functions. Overhead time (from the creation and synchronisation of threads) is minimal across all functions. The issue of load imbalance is discussed further in Section 5.4.2.

5.4 Discussion

5.4.1 Parallel scaling

There are four factors that potentially affect the parallel scaling observed in the benchmarking test cases presented.

1. The number of grid cells in a model domain, i.e. the total problem size, which is dependent on the resolution of the DEM input data.
2. The choice of running the simulation as hydrology-only or erosion-enabled model. The erosion option in the model also increases memory demands significantly compared to the hydrology-only mode, as well as computational expense.
3. The hydrological state of the catchment. When the catchment is experiencing flooding due to high rates of rainfall input, the number of cells registered as containing water through the *scan-area* algorithm increases substantially. The location of the water-containing cells in a catchment also determines the amount of load-imbalance in the model. If water-containing cells are concentrated only in a certain area of the catchment, significant load-imbalance will occur.

The most favourable speed-up was observed in the Boscastle 48 hour simulation at 5m DEM resolution (Figure 5.5.) A speed-up of almost 16x was recorded using the maximum number of cores available on the test hardware. Increasing the length and resolution of the Boscastle simulation appeared to negatively affect the speed-up potential of the simulation. Speed-up gains remained approximately linear into higher core counts, with slight tail-off in speed-up gains observed when approaching 48 cores. The longer-term Swale simulation exhibited poorer scaling, with a gradual drop in speed-up gains as the number of threads/cores was increased.

There is not a clear relationship between number of grid cells in the model domain and speed-up potential. The Swale test case (c. 125 000 grid cells), for example, showed almost half as much speed up as the 48 hour Boscastle simulation, despite the latter having 720 000 grid cells in the model domain.

The enabling of erosional processes in simulations did not significantly affect the scaling potential of the simulations. Most benchmark results showed slightly less

favourable speed up at higher core counts for erosion-enabled simulations, but this did not appear to be the controlling factor on scaling. In one case (Swale) the erosion-enabled simulation ran 2x faster than the corresponding hydrology-only simulation. The lack of contrast between erosion-enabled and hydrology-only simulations may be explained by the adaptive time stepping feature used in the *erode* function. In erosion-enabled mode, the timestep increases substantially during periods of low water flow and erosion rates, and this may offset the increased computational demands from the *erode* function.

The Boscastle and Swale test cases represent different types of catchment hydrological state. The Boscastle simulations essentially represent a single intense flooding event, where the catchment is in a period of low flow followed by an intense burst of rainfall and erosion for a few hours. The Swale simulation represents a series of storm events, interspersed with ‘dry’ periods, where the model will be able to increase the timestep, but not make heavy use of parallel sections of the code reserved for water routing and erosion routines.

5.4.2 Load balancing issues

Parallel computing problems in which the distribution of workload between each thread or processor is uneven are said to suffer from *load-imbalance*. In other words, although the model domain is divided into equally sized chunks for the number of threads and processors available, each thread will not necessarily receive an equal amount of work to do (Sakellariou, 1996). Threads with little amounts of work to do will therefore remain idle while waiting for heavily-loaded threads complete their assigned work. Numerical simulation of river catchments is inherently load-imbalanced. The load balance problem arises from the nature of catchment processes, where the most rapid rates of geomorphological activity occur in river channels. In river channels, rates of water flow and sediment transport are greatest, relative to the processes operating on hillslopes. The inherent heterogeneity in the nature of catchment processes translates into computational imbalance during model simulations of river catchments. Within the model domain representing a catchment, the grid cells of the domain representing river channel sections will make the greatest computational demand, due to repeated calls to subroutines that calculate or update cell properties. Highly active river

channel cells will frequently exceed thresholds set for the calculation of water flows and sediment entrainment, whereas cells representing hillslope areas will be active relatively infrequently.

The mechanism in the model that determines which cells will be updated each iteration is the scanning algorithm described in the *scan_area* section. The algorithm was originally designed (Coulthard et al., 2013) to substantially reduce the number of cells in the model domain that had to be updated each iteration, by isolating the cells that contain water and only performing erosion and sediment transport routines on those ‘wetted’ cells. While the algorithm is effective at reducing compute time this way, it creates a load-imbalance problem for parallel implementations as the location of computationally intensive areas of the model domain cannot be easily predetermined. While threads are assigned equal numbers of iterations, each iteration is not necessarily equally load-balanced in terms of computational expense. In algorithms that perform a global scan of all cells in the model domain, such as the parallel implementation of the LISFLOOD-FP algorithm in Neal et al. (2009), load imbalance is minimalised because threads are kept occupied working on every grid cell, rather than focusing on a subset of grid cells with discharge or water depth above a given threshold, as is done in the HAIL-CAESAR model. A global scanning algorithm was initially explored in the implementation of HAIL-CAESAR, allowing the removal of the *scan_area* function, but the increase in run-time was so substantial it negated any potential improvements in load balance among the OpenMP threads.

5.4.3 Loop scheduling

The OpenMP application programming interface (API) will automatically partition iterations of a loop between the available number of threads. By default these are assigned statically when a `parallel for` block is entered, with threads receiving an equal number of loop iterations to carry out. The default behaviour can be overridden, however, by explicitly specifying a different scheduling type for parallel for loops. For load imbalanced problems such as discussed in this paper, the *dynamic* loop scheduling can improve parallel performance by reducing the amount of idle time that threads spend waiting for busy threads to complete their workload (Willebeek-LeMair and Reeves, 1993; Olivier et al., 2012). The default behaviour of the dynamic scheduling

type is to assign each loop iteration to one thread, and when the thread finishes it is assigned the next iteration that has yet to be executed. Experiments in finding an ideal loop scheduling set-up to reduce load imbalance in HAIL-CAESAR showed mixed results, with no clear indication that using the dynamic scheduling clause improved run-times in all cases. However, in some test cases, dynamic loop scheduling was shown to offer performance improvement. For this reason, the HAIL-CAESAR model provides users with the option to set the loop scheduling type at run time, by specifying the `OMP_SCHEDULE` environment variable before running the model. It is recommended that users experiment with using dynamic scheduling for their own simulations as it may offer performance increases over default scheduling.

5.5 Conclusions

An implementation of a hydrodynamic landscape evolution model is presented, based on the CAESAR-Lisflood model (Coulthard et al., 2013). The new C++ implementation of the model is modular, cross-platform, and scalable to multi-core computing systems through the implementation of shared-memory parallelism with the OpenMP API. Minimal code changes were required to the serial version of the code to deliver a parallel compute code. The model exhibits a performance increase of c.40% compared to the CAESAR-Lisflood implementation in a like-for-like benchmark. The scaling potential of the model is variable, depending on the size of the problem domain and the likely hydrological state of the river catchment being simulated. In most test cases, linear speed up is achieved at moderate core counts of up to 48 cores, on model domain sizes of up to 4.5 million grid cells. The causes of the load balancing issues are discussed, with the key reason being due to the spatially-heterogeneous nature of catchment-scale hydrological and erosional processes, which results in computational load imbalance. Suggestions for minimising load imbalance through dynamic loop scheduling are made, but the success of these methods is highly dependent on the input data to the model, should users wish to explore this option of increasing performance further.

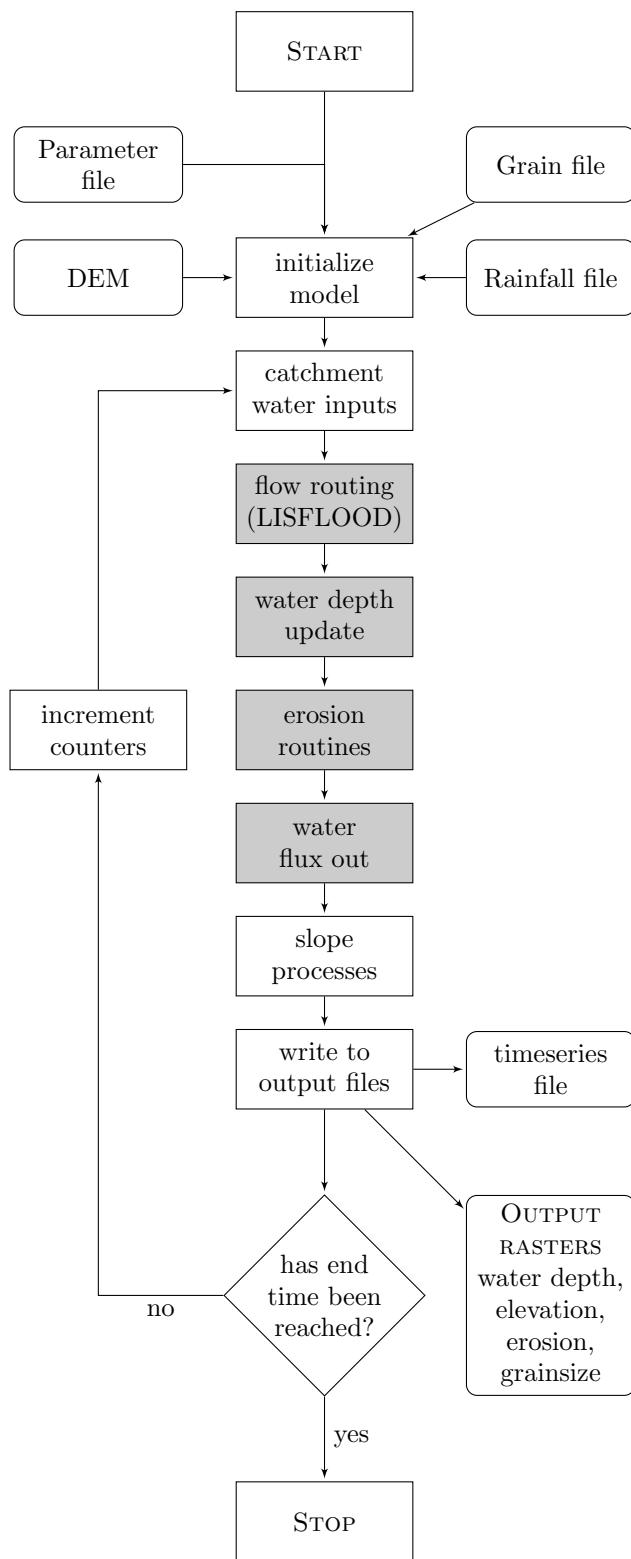


Figure 5.1: Flow chart showing a simplified outline of the HAIL-CAESAR program flow. Grey shaded boxes indicate sections of the code parallelised with OpenMP. Rounded rectangles indicate output and input files.

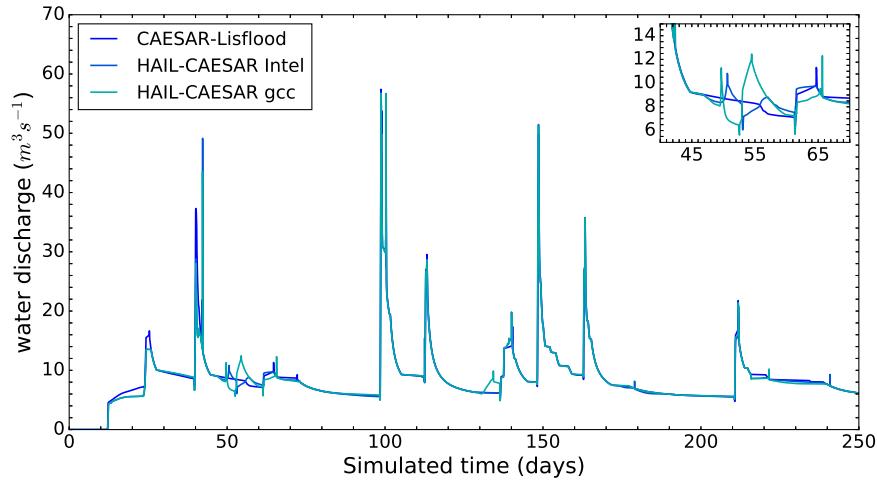


Figure 5.2: Water discharge rates over first 250 days of simulation. Outputs from the Swale 10 year at 50m resolution test case, showing results from the original CAESAR-Lisflood model and HAIL-CAESAR model compiled under two different compilers.

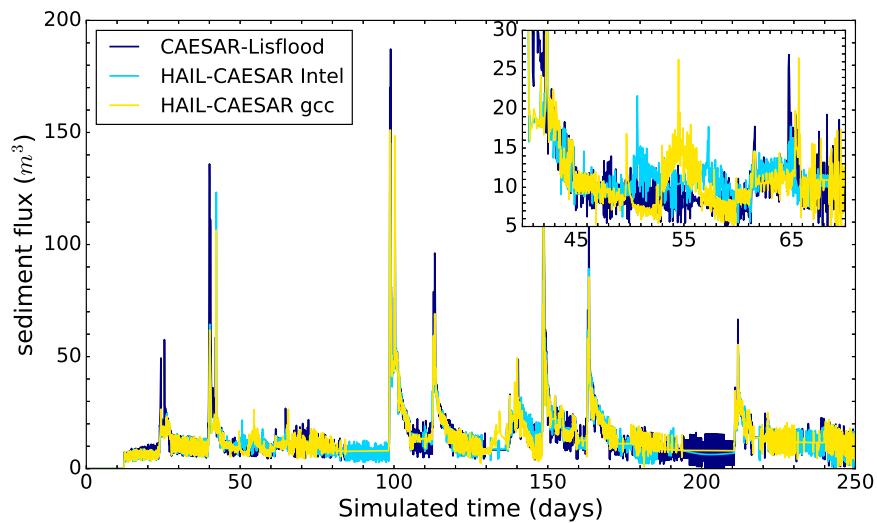


Figure 5.3: Catchment sediment flux over first 250 days of simulation. Outputs from the Swale 10 year at 50m resolution test case, showing results from the original CAESAR-Lisflood model and HAIL-CAESAR model compiled under two different compilers.

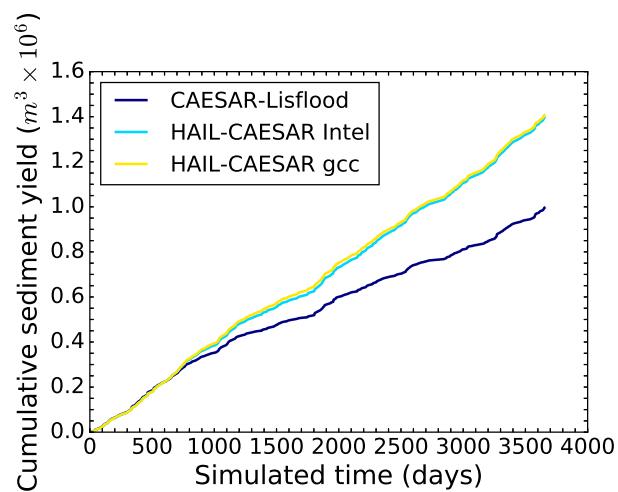


Figure 5.4: Cumulative catchment sediment flux over full duration of simulation. Outputs from the Swale 10 year at 50m resolution test case, showing results from the original CAESAR-Lisflood model and HAIL-CAESAR model compiled under two different compilers.

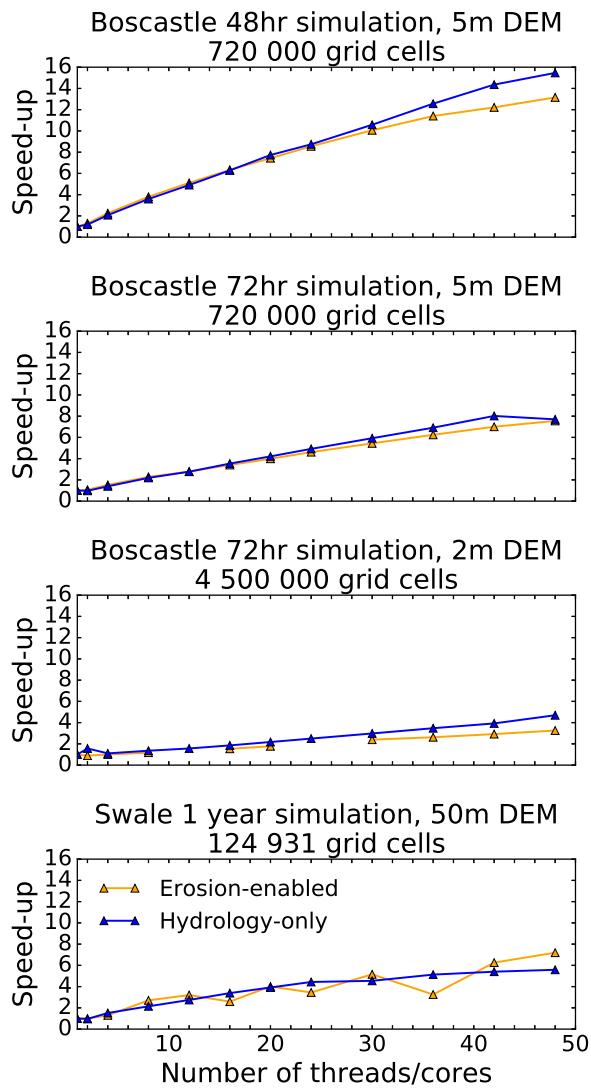


Figure 5.5: Speed-up achieved relative to serial code on 2, 4, 8, 12, 16, 20, 24, 30, 36, 42, and 48 cores. Four sets of simulations are used representing a range of model uses from short, single event episodes (48–72hr) to 1 year simulations.

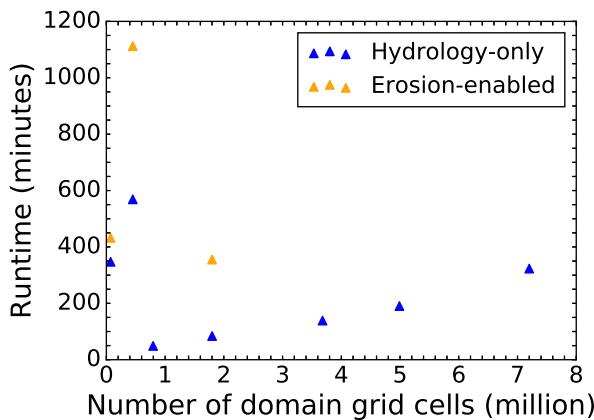


Figure 5.6: Weak scaling with the Boscastle DEM at increasing resolution (see Table 5.6). Hydrological and erosion-enabled simulations shown. Each simulation uses c.150 000 grid cells per CPU. The absence of results for certain erosion simulations is due to memory limitations preventing higher resolution simulations from running.

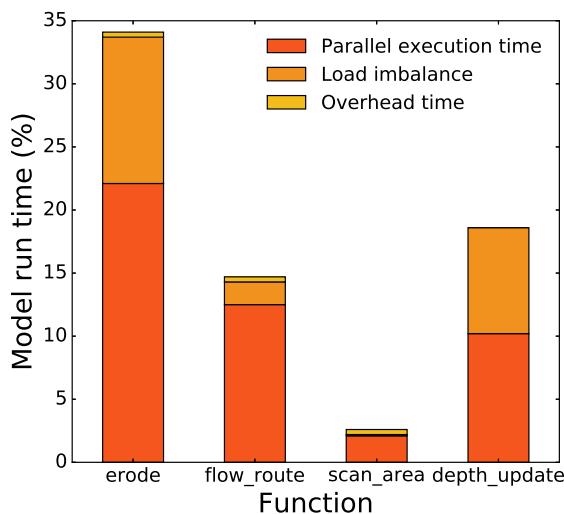


Figure 5.7: Thread profiling of the Swale 1 year simulation with 50m DEM. Data from major functions only displayed.

6

Rain or rock? Sensitivity of a flood-inundation model to rainfall distribution and erosional parameterisation

6.1 Introduction

Patterns of rainfall distribution across a catchment can affect hydrograph response, including the peak discharge and local water levels (Nicotina et al., 2008). As many geomorphic processes are threshold dependent (Schumm, 1979), such as fluvial incision into bedrock (Sklar and Dietrich, 2001; Snyder et al., 2003), there is potential for the spatial distribution of rainfall to control local erosion rates within a catchment. Non-linearity in geomorphic process laws (e.g. Coulthard et al., 1998; Phillips, 2003; Coulthard and Van de Wiel, 2007) should dictate that catchments are also geomorphically sensitive to the spatial distribution of rainfall.

The following questions are explored through the use of numerical modelling simulations;

- Are predicted flood inundation extents during a storm sensitive to the spatial resolution of rainfall inputs to the catchment?
- Are fluvial erosion and sediment transport processes controls on predicted flood

Experiment name	Rainfall input	Erosion law
UNIFORM_HYDRO	Spatially averaged	(no erosion)
UNIFORM_DLIM	Spatially averaged	Detachment-limited
UNIFORM_TLIM	Spatially averaged	Transport-limited
GRIDDED_HYDRO	1km Gridded	(no erosion)
GRIDDED_DLIM	1km Gridded	Detachment-limited
GRIDDED_TLIM	1km Gridded	Transport-limited

Table 6.1: Outline of the ensemble simulations carried for all case studies.

inundation extents in a catchment during a single severe storm.

- Does geomorphic change of the floodplain and channel during a storm event significant enough to affect flood inundation predictions?

6.2 Experiment Design & Method

An ensemble of simulations was designed to address the questions laid out in the previous section. The simulations were parametrised with different erosion law schemes and different rainfall input resolutions to assess which environmental factors exerted the greatest control over catchment response to intense rainfall, in terms of flood inundation and sediment transport. The same ensemble of experiments was carried out for two historic rainfall events that lead to severe flooding in UK river catchments. The summary of the ensemble events is given in the table below:

The numerical landscape evolution model, HAIL-CAESAR (described in Chapter 5) is used to investigate landscape and hydrological response to varying rainfall input resolution. HAIL-CAESAR is a hydrodynamic landscape evolution model that uses a TOPMODEL-based hydrological rainfall-runoff model to generate surface runoff in a river catchment, which is then routed through the landscape according to an adaptation of the LISFLOOD-FP shallow water routing algorithms (Bates et al., 2010). Fluvial erosion and sediment transport is then derived using the velocities and depths of the calculated surface water within the catchment. The equations describing the water routing, sediment erosion, and transport laws are described fully in Chapter 5.

Catchment Name	Ryedale	Valency
Catchment Area	270km ²	18km ²
Catchment Type	Upland, Moor/Peaty	Upland, Pasture
Storm Date	2005-06-19	2004-08-16
Peak Rainfall (mm hr ⁻¹)	125	c.400
Peak Discharge		
Meteorological Setting	Split-front, convective system	Quasi-stationary convective system
3hr Rainfall Return Period	330yr (Wass et al. 2008)	1300yr (Burt, 2005)

Table 6.2: Table showing key characteristics of each storm event.

6.2.1 Study area

Overview

Three upland catchments in the UK were selected to represent a range of catchment sizes and shapes. The catchments were also chosen on the basis that they had experienced a severe rain storm which could be used as a basis for the experiments, such that it could be considered ‘extreme’ in the typical return period of flooding events for each particular catchment. Peak discharges for each of the following flood events exceed the 99th percentile for their respective catchments. The catchments and respective severe rain events chosen were located in: Ryedale, North Yorkshire, 2005; Eden, Cumbria 2012; and Boscastle, Cornwall, 2004. An overview map of their locations is given in Figure ???. A table (Table 6.2) summarises the key features of each catchment and associated storm.

6.2.2 Hydro-Meteorological conditions

Boscastle, Cornwall storm 2004

The Boscastle storm took place on the 16th August 2004 leading to flooding within the River Valency catchment and the village of Boscastle. In the preceding months March–June, the south-west region had been drier than usual, but during July the Valency catchment area experienced average rainfall conditions (Golding et al., 2005). The estimated soil moisture deficit¹ in the area, which had been lower than average due to the dry antecedent conditions, decreased during the period of 1–16th August, due to the return to average rainfall conditions in that month. The soil moisture deficit

¹The amount of water needed to bring the soil back to field capacity, i.e. a state where the soil is holding the maximum amount of water possible against gravity. (bev)

during this period was estimated to have decreased from approximately 80–220mm to 40–180mm.

On the day of the storm, the extreme rainfall accumulations of up to 200 mm in the upper Valency catchment resulted from prolonged rainfall between the hours of 1200 – 1600 UTC. Rainfall rates were thought to have reached almost 400 mm hr^{-1} (?), after correcting for under-reporting from rain gauges in the vicinity of the catchment. (Burt, 2006).

The meteorological conditions that enabled such prolonged heavy rainfall were a combination of large-scale synoptic conditions moving in from the Atlantic, with moist lower atmospheric layers readily forming convective cloud. Repeated initiation of convection along the north Cornish coast lead to what appeared to be relative stationary convective cells over the Valency catchment. Later authors refer to this type of convective storm as a ‘Boscastle-type’ or quasi-stationary convective storm (Warren et al., 2014).

Ryedale, North York Moors storm 2005

The Ryedale storm occurred on 19 June 2005. Intense rainfall throughout the afternoon lead to total accumulated rainfall amounts of up to 89mm in the Ryedale valley, between the hours of 1400 – 1800 UTC. Peak instantaneous rainfall rates were estimated to have been around 32.5 mm hr^{-1} (Sibley et al., 2009) to 59.4 mm hr^{-1} (Hopkins et al. 2010), though one report states they reached as high as 125 mm hr^{-1} (Cinderley, 2005). The antecedent conditions had been dry for a prolonged spell, leading to cracking of the surface peat in the higher elevations of the catchment.

Antecedent conditions before the Ryedale storm of 2005 had been dry over a prolonged period over much of the region (Sibley, 2009). Soil moisture deficit was estimated to be around 60mm in the catchment, higher than usual due to the drier conditions in the preceding months (Wass et al., 2008). With a low soil moisture content, thinner soils in the upper reaches of the catchment would have been dry before the intense rainfall.

The meteorological conditions leading to such heavy rainfall was a combination of a cold, upper-level air mass advecting over a warm moist boundary layer, leading to unstable conditions that enabled a convective thunderstorm to develop in the late

afternoon. The instability was enhanced by a split-frontal system. [More? Too much met here?]. The conditions led to a particularly high amount of precipitable water present in the atmosphere which was subsequently washed out into the landscape during intense rainfall.

6.2.3 Numerical model configuration

Model initialisation

The simulations are set to begin 24 hours before the day of each intense rainfall event. The model is allowed to run for a further 24 hours after the event, giving a total simulation time of 72 hours. The timing of the simulations is given in Table 6.3.

Event	Start time	End time
Boscastle	2004-08-15 00:00	2004-08-17 23:59
Ryedale	2005-06-18 00:00	2005-06-20 23:59

Table 6.3: Start and end times (UTC) for each 72 simulation. The major rainfall event occurs during the 2nd day in each simulation.

Erosion model choice

A variety of erosion laws exist describing how landscapes erode from fluvial incision. The choice of erosion law for a given catchment depends on a variety of factors, such as the characteristic substrate material in the catchment – is it predominantly loose sediment or cohesive, solid bedrock? In reality, landscapes are often a mixture of these two extremes, incorporating loose sediment on top of solid bedrock. Catchments also often exhibit a transition from rockier upland headwaters, to more thickly soil-mantled flood plains. In order to address the uncertainty in choosing which erosion model applies for each catchment (Section ??), two erosion model end-members are used, with each one representing a different conceptual model of fluvial incision and sediment transport. These include: i) a purely sediment transport-limited model, ii) a detachment-limited bedrock incision model. The equations describing the transport-limited and detachment-limited models are discussed in Section ???. Further models were considered, such as a hybrid transport-detachment limited erosion model, but it was deemed beyond the scope of this study, which is to focus on the sensitivity of rainfall resolution, rather than wide range of erosion and sediment transport models.

A set of control simulations parameterising only runoff and surface water routing (no erosion taking place) were also carried out for comparison against the two erosion end-member simulations. (Table 6.1.)

Rainfall spatial resolution

In order to assess the sensitivity of each erosional model to the spatial details of precipitation, rainfall spatial parameterisation is alternated in each simulation between a spatially uniform rainfall input and a 1km gridded rainfall input. Both the spatially uniform and gridded inputs are based on the same original rainfall source data - the UK Nimrod 1km-composite radar data product (Met Office, 2003). To reduce the range of variables within the set of ensemble simulations, only the spatial distribution and resolution of rainfall is assessed in this study – other studies have previously investigated the effects of the *temporal* resolution of rainfall data on discharge and erosion rates (Nicótina et al., 2008; Coulthard et al., 2013; Coulthard and Skinner, 2016). In all experiments, the temporal resolution of experiments is maintained at 5 minute intervals. To summarise, the two types of rainfall spatial input used are:

- Uniform or ‘lumped’ precipitation: Radar-derived rainfall rates across the catchment are spatially-averaged to produce a basin-wide rainfall rate. In other words, every grid cell in the model domain receives the same rainfall rate at each rainfall data timestep.
- Gridded rainfall input. The rainfall rate is input from a overlying gridded mesh of raincells, at the same resolution as the rainfall radar product (1km).

6.3 Sensitivity analysis

There are numerous user defined parameters in the HAIL-CAESAR model, and in landscape evolution models in general, that have a wide range of potential values. Parameter selection in environmental modelling comes with a degree of uncertainty, and resulting outputs from models can be highly sensitive to the user’s choice of input parameters for a given simulation (Pelletier, 2012). Initial testing of the HAIL-CAESAR model, and studies using the CAESAR-Lisflood model that it is based upon

Catchment	TOPMODEL m parameter
Boscastle	
Ryedale	0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.15, 0.2

Table 6.4: TOPMODEL m parameter values used to run sensitivity simulations for each catchment.

show it is particularly sensitive to the TOPMODEL m parameter, a parameter that controls the rise and fall of the soil moisture store, and hence how a river catchment responds to rainfall input (Beven and Kirkby, 1979).

To assess the sensitivity of the model to the choice of TOPMODEL- m parameter, a series of simulations were carried out with a range of m values for both test cases. Simulations of each flood event were carried out with the m values shown in table 6.4.

6.4 Results

The discussion will be aided by figures showing (for each of the three rainfall input variations for each catchment):

- Total accumulated rainfall maps for each storm (Figure ??).
- Profiles of erosion along main channels in each catchment (Figure ??).
- Plots of the hydrographs and sediment yields for each storm (Figure ??).
- 2D Planform maps of distribution of erosion (and deposition if applicable).

Indicative figures. Note these will change in the final version as I have decided to re-run some simulations after tweaking the model set-up.

6.4.1 TOPMODEL sensitivity

The model exhibited a strong sensitivity to the choice of the TOPMODEL m parameter. In the results presented in figure 6.1 peak river discharge ranged from XX at an m value of 0.003 to XX with an m value of 0.008. The measured peak discharge reported for the 2005 storm at the Ness gauging station was 105 cumecs. In the sensitivity

simulations, a value of $m = 0.005$ produced a flood peak most closely matching the observed value, peaking at XX cumecs (Note the time as well.).

There were differences between the hydrographs of the observed and simulated discharges in terms of peak discharge timing and recession limb shape. The observed hydrograph showed a sharp rise at around 50 hours after the start of the simulation period. (UTC XX XX XX to X). Lower m values ($m < 0.005$) resulted in a prediction of the flood peak being too early compared to the observed timing, with values > 0.005 predicting the flood peak timing too late. Most of the simulations failed to capture the extended duration of peak discharge, which lasted approximately 5–6 hours, before receding back to low flow levels. The simulation with $m = 0.006$ came closest to predicting this hydrograph shape, but failed to predict the magnitude of water discharge correctly, underestimating the peak flow by almost 50%.

For Ryedale simulations, it was decided to use an m value of 0.005, providing the closest possible match to the flood peak discharge, though not the true shape of the hydrograph and the receding limb. As the catchment simulations include a representation of erosion and sediment transport processes, which are often threshold dependent, it was felt necessary to match the discharge peak more closely over choosing to match the hydrograph shape precisely.

6.4.2 Catchment hydrology

At the catchment scale, hydrological response was sensitive to both the rainfall resolution and the choice of erosional model. For both catchments, higher resolution rainfall input data resulted in a greater maximum river discharge. In the Ryedale experiments, simulations using the gridded rainfall input experienced a flashier hydrological response, reaching peak discharge several hours before the uniform rainfall input cases. This was not observed in any of the Boscastle simulations, with all simulations reaching peak discharge within 30 minutes of each other.

The choice of erosion model also influenced the hydrological response. When catchment erosion was modelled using a transport-limited case, peak discharges were higher in all cases, but the timing of hydrograph peaks remained similar for each case. The difference in peak discharges were minimal when comparing the detachment-limited cases to the hydrological-only models.

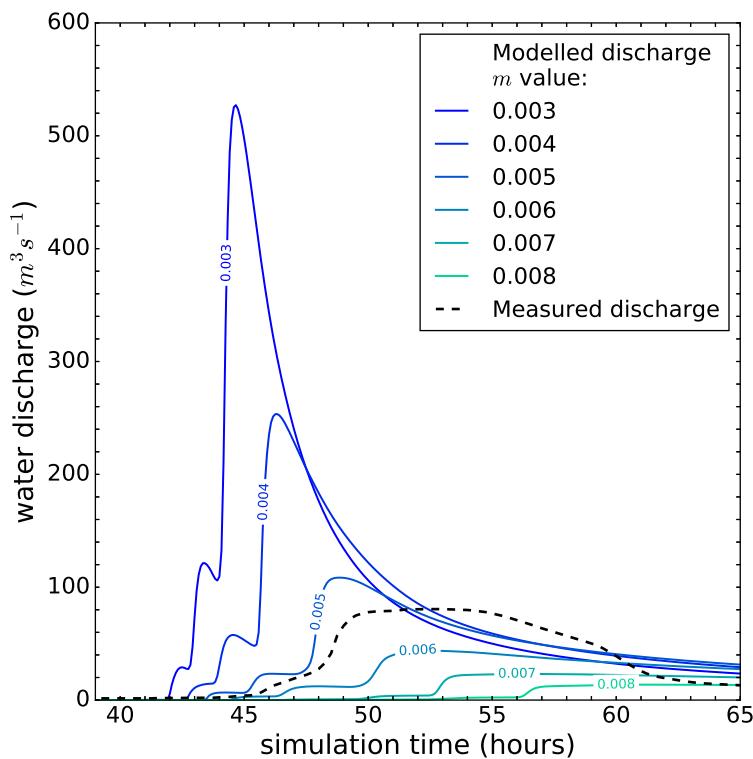


Figure 6.1: Discharge at Ryedale catchment outlet for varying values of the TOP-MODEL m parameter. The measured discharge at the catchment gauging station is overlain in dashed line. The results from the simulations with $m > 0.008$ are omitted for clarity due to the low peak discharges they produced.

6.4.3 Spatial variation in flood inundation

Boscastle

The Boscastle catchment simulations showed minimal variation in flood inundation extent. Simulations with gridded rainfall input did not result in substantially different predictions of floodwater inundation compared to those with uniform rainfall input. Both sets of simulations reflected the general extent of reported flood water extents (Cite Engineers Report XXXX). Simulations that allowed erosion to take place (GRID_TLIM and UNIFORM_TLIM) showed a slight difference in the variation of floodwater depths in the floodplain area, particularly in the vicinity of Boscastle village, where Figure 6.4 is centred on. In hydrological-only simulations, the deepest water depths were predicted to occur in the confines of the river channel, whereas in erosion-enabled simulation, there appeared to be a ‘smoothing’ effect of water depths

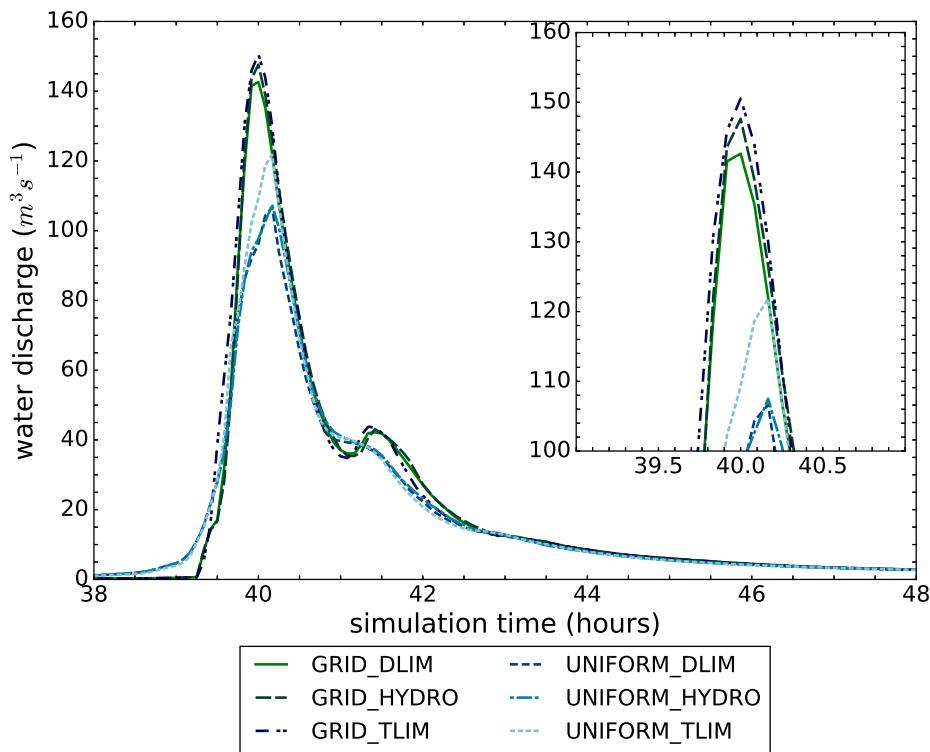


Figure 6.2: Boscastle hydrographs (discharge over time at catchment outlet) for each simulation of the 2004 Boscastle event listed in Table 6.1. Inset shows detail of main flood peaks around hour 40 of the simulation.

between the channel and the adjacent floodplain, suggesting that the channel geometry had altered during the flood event either by infilling from sediment from upstream or collapse of the adjacent river banks. Engineer's reports of the Boscastle flood noted that the river channel in the Boscastle village area had indeed been inundated with debris during the storm, which had potentially contributed to the extent of the flooding within the village.

Ryedale

The Ryedale simulations showed greater variation in flood extent between the gridded rainfall and uniform rainfall inputs. The flood extents were greater in the gridded rainfall input simulations, corresponding with the higher peak discharges predicted in these simulations. The variation in water depths appeared to be less sensitive in comparison to the Boscastle simulations. In the lower reaches of the catchment (Figure 6.5, there appeared to be little indication that flood extents or water depths

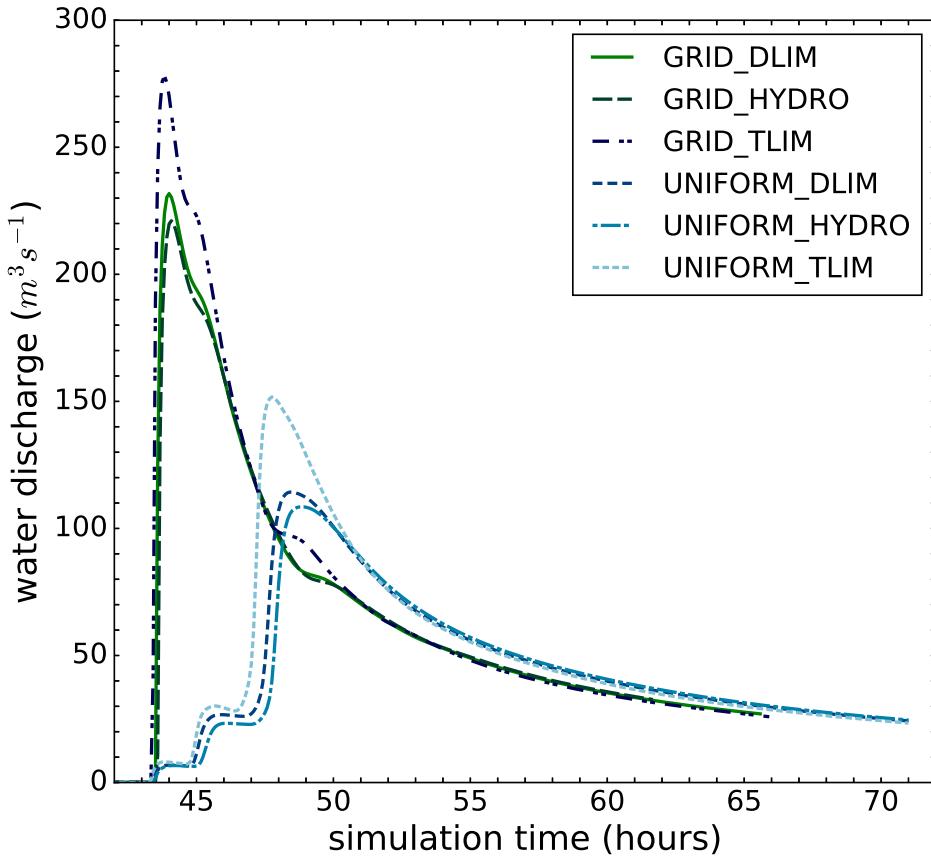


Figure 6.3: Ryedale hydrographs (discharge over time at catchment outlet) for each simulation of the 2005 Ryedale event listed in Table 6.1.

were sensitive to the erosion parameterisation, in contrast to the Boscastle simulations.

6.5 Discussion

The size of the two catchments appears to be a determining factor in how sensitive hydrogeomorphic processes are to the rainfall data input resolution. The Boscastle catchment at 18 km^2 is approximately an order of magnitude smaller than the Ryedale catchment at 270 km^2 . For each simulation using gridded rainfall input, the cell width of the rainfall grid is 1 km. The relative amount of increase in rainfall detail between uniform and gridded simulations is much greater in the larger Ryedale catchment than the Boscastle catchment. By using a 1 km gridded rainfall product as input data, the Ryedale simulation potentially captures 15 times more rainfall heterogeneity compared to the respective Boscastle simulation, by virtue of it simply being a much

larger catchment with a greater number of rainfall input cells. Catchment size is known to be a factor in hydrological studies of rainfall resolution, and larger catchments are reported to show greater sensitivity to rainfall resolution (e.g. Nicótina et al., 2008), in terms of hydrological response.

Increasing the resolution of rainfall input data may not be enough to observe sensitivity in smaller catchments, as rainfall features themselves may not exhibit the necessary heterogeneity in structure to benefit from being resolved at finer scale. Rain cells or bands equal to or greater in size than the catchment over which they rain upon, may well be homogeneous enough in spatial extent and rainfall rate that a ‘uniform’ approximation of their rainfall rate is sufficiently precise enough to represent the rainfall rate at all points in the catchment. As seen in the Boscastle catchment simulations, using a detailed rainfall input data source did not notably alter the outcome of the hydrological predictions (Figures 6.4, 6.5). In the Ryedale catchment simulations, the hydrological predictions were notably different based on the choice of rainfall input data resolution, affecting both the time and magnitude of the resulting flood peak.

Talk about size of convective cell features. What is typical storm cell size and heterogeneity (decay from centre?).

6.5.1 Implications for longer-term landscape evolution

The experiments presented in this chapter have focused on the hydrogeomorphic response to single severe storm events, events which produced floods with return periods of 1 in 330 years (Ryedale) and 1 in 1300 years (Boscastle). The amounts of river channel incision predicted as a result of these storms is comparable to that predicted by studies of landscape evolution on scales of 1000 years, for example a study of a similar upland river basin by (Coulthard and Skinner, 2016) predicted channel incision amounts of 0.5–5m over 1000 years. The experiments presented here, using the same sediment transport law, predicts comparable incision amounts of channel incision during a single storm. If the flood return periods are assumed to be broadly correct, these simulations suggest that the majority of sediment erosion occurs during rare but high magnitude flood events, rather than through gradual processes or more frequent but lower magnitude events.

6.6 Conclusions

Over long term landscape evolution, erosional processes in catchments are known to be sensitive to the spatial distribution of rainfall input. Catchment hydrological response on the short term scale – such as during the course of single storm event – is also sensitive to the spatial input pattern of rainfall, with the simulations carried out in this study also supporting similar work of (Nicótina et al., 2008) AND OTHERS. This study has shown that catchment erosional processes are also sensitive to rainfall spatial distribution during the course of a single severe storm, and it is suggested that this is due to the spatial variation in shear stresses required brought on by heterogeneous rainfall inputs to the catchment system. As sediment transport and erosional process are highly threshold dependent, this leads to erosional patterns that differ according to the pattern of rainfall input. In other words, in the simulation of landscape evolution processes at the catchment scale, the choice of whether to use a uniform value representing the rainfall input, or to use a spatially heterogeneous gridded rainfall data, can have a marked impact on the predicted sediment yields from the catchment, as well as slight variations in the prediction flood inundation extents and the timing of peak flood levels.

In terms of sediment yields, however, these experiments have shown that it is the choice of erosional parameterisation in the model, and not necessarily the resolution of rainfall input data, that has a first-order control on the total sediment yields from a catchment and the magnitude of river channel incision during a simulated event.

Stuff amount rarity of events. Catastrophism.

6.7 Fixed parameters

A table showing the other parameters used in the simulations (All of which remain fixed for each simulation)

The following table lists the parameters that were held constant for all simulations.

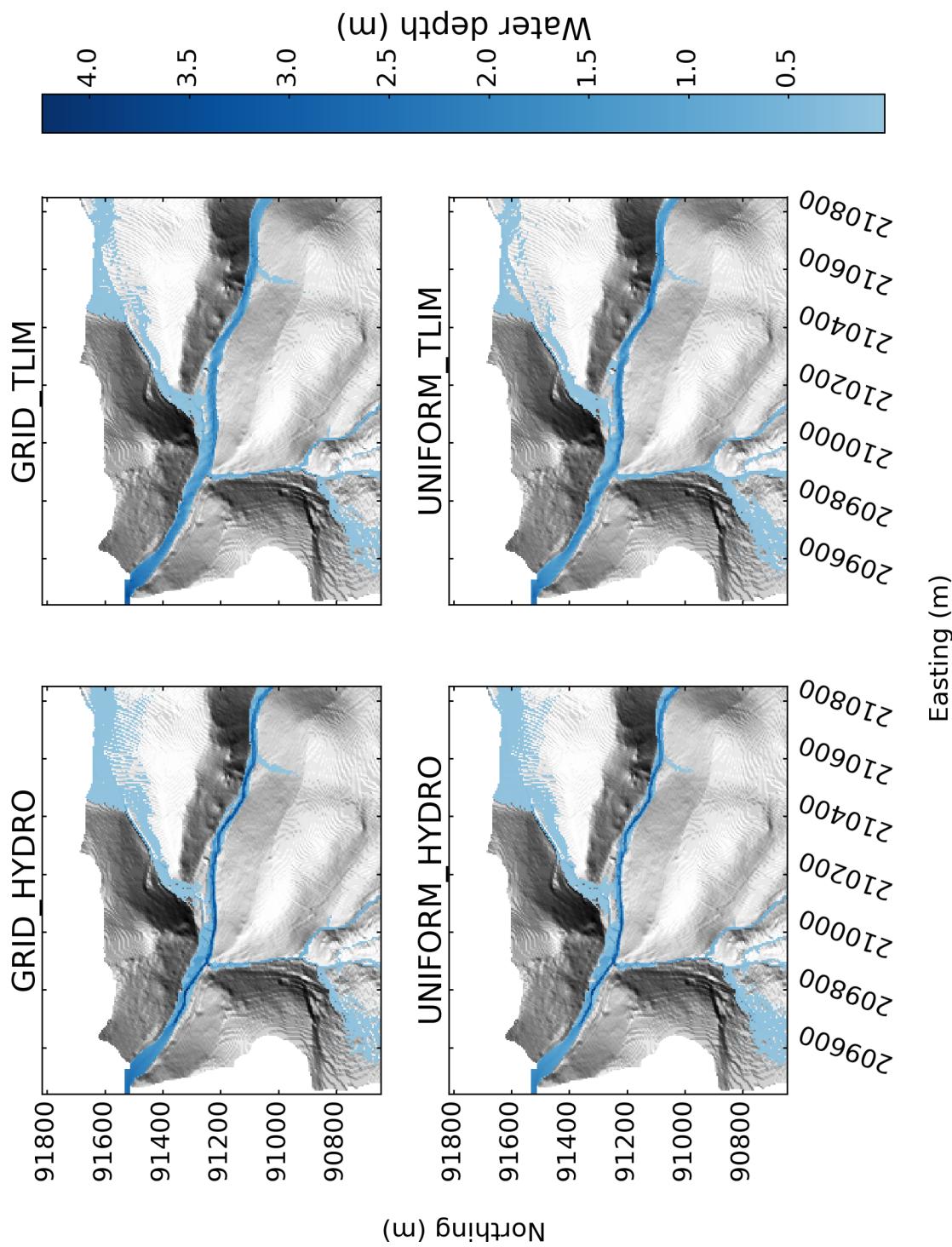


Figure 6.4: Flood extents in the Boscastle catchment at the time of maximum river discharge for each simulation.

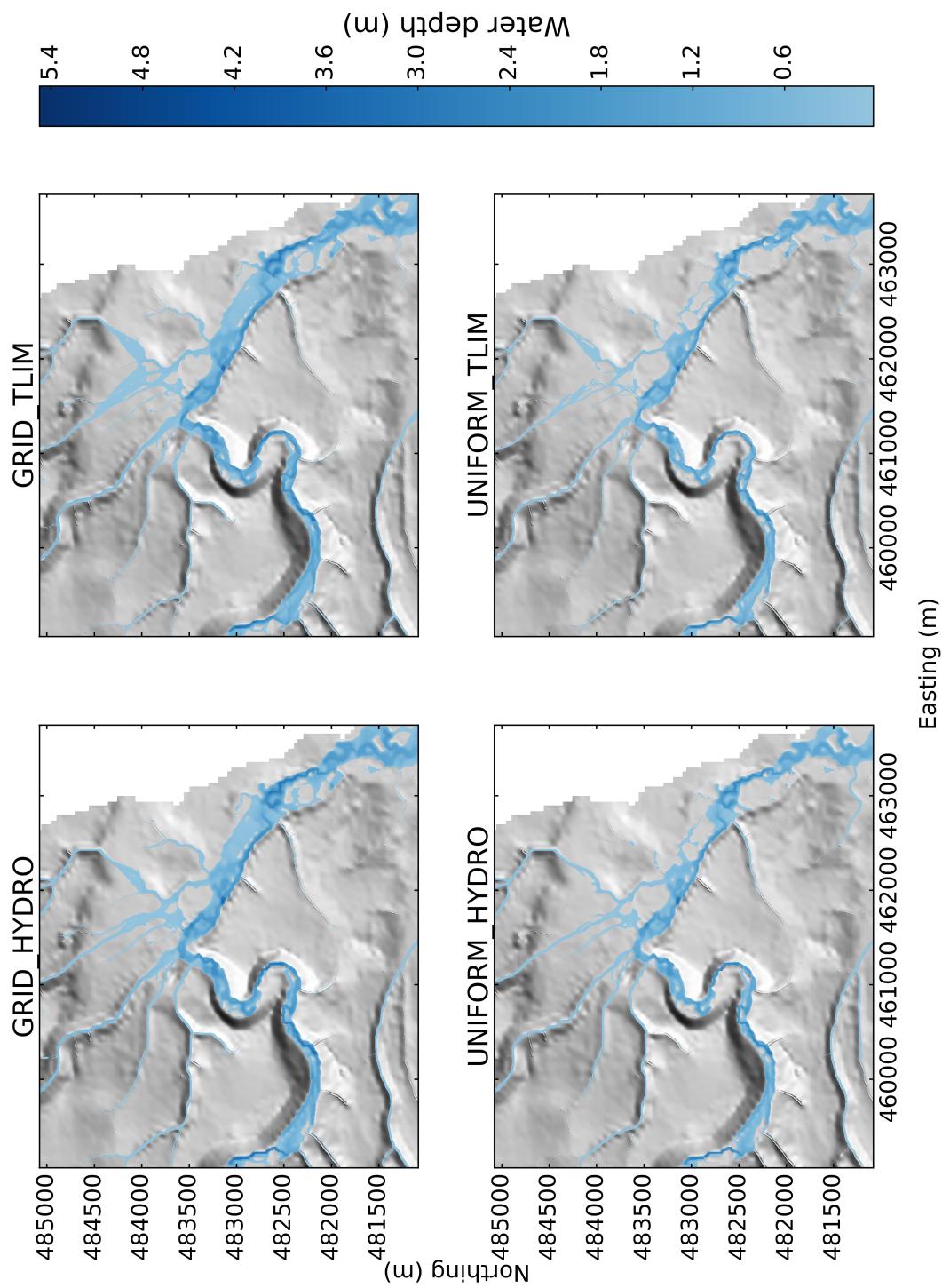


Figure 6.5: Flood extents in the Ryedale catchment at the time of maximum river discharge for each simulation.

7

Hydrometeorological controls on landscape erosion and evolution

7.1 Introduction

Describe how sediment and erosional processes are sensitive to rainfall spatial inputs.

Landscape evolution at the catchment scale is punctuated by intense erosive episodes driven by flood events (Wolman and Miller, 1960; Newson, 1980; Costa and O'Connor, 1995) interspersed with periods of relative calm and little geomorphic change, an idea that harks back to the early ideas of geological ‘catastrophism’ (Cuvier, ?). It is these erosive events, driven by intense rainfall in temperate climates, separated by long periods of stasis, that cumulatively sculpt the landscape over geological time. The importance of these rare but formative events has been revisited by recent work such as Huang and Niemann (2006), looking at the long term implications of different geomorphically effective event discharges on fluvial incision; Gupta et al., (2007); Lamb and Fonstad (2010); and Baynes et al., (2015) where the amount of bedrock erosion during a single large flood event was quantified. Still, our understanding of catchment scale landscape evolution is far from complete - the role of individual events is highly variable. It has been reported that rapid gorge formation can be driven primarily by small to moderate sized floods (Anton 2012), rather than floods of extreme magnitude. The relationship between flood magnitude and erosion rate is not always observed in case studies (Anton 2012), suggesting that the links between precipitation, hydrological response, and sediment transport are not fully understood. The behaviour of

certain catchment processes is non-linear (Coulthard, XXXX). Turowski et al. (2011) report that streams within catchments can exhibit different behaviour in response to the same flood event – some streams may erode during high flows, whereas others may deposit during high flows. During small–medium flows their respective behaviour is reversed. Wong et al. (2015) establish through numerical modelling that geomorphic changes in channel geometry during severe flood events are substantial enough to change hydrological response of a river catchment. Catchment-scale erosional dynamics are complex, and except in the simplest cases depend on other forcings other than the magnitude of single flood events alone. The understanding of hydrogeomorphic processes during single storm events is not only important for the long-term evolution of landscapes, but also for prediction of how catchments will respond to changing hydro-meteorological conditions that may accompany climate change (Kendon et al., 2014).

The focus in this chapter is to quantify the sensitivity of catchment-scale erosional processes to the spatial distribution of rainfall during flood events. The assumption of uniform rainfall over a river catchment is argued to hold true for small catchments (Solyom and Tucker 2004; Tucker 2010), but even over small areas, mesoscale rainfall features, such as localized convective storm cells, can result in spatially and temporally uneven input of precipitation into the catchment. In the case of intense convective precipitation, individual storm cells can be as small as 10km^2 in areal extent (Weisman and Klemp, 1986; Von Hardenberg et al., 2003). Over larger catchments, or those with steep topographic gradients, precipitation is almost certain to vary spatially, due to orographic enhancement of rainfall (Roe, 2005). As such, rainfall-runoff generation, local river flow, and erosion rates may vary considerably within individual drainage basins.

Rainfall resolution has been demonstrated to exert a control over sediment yields over seasonal and decadal timescales (Coulthard and Skinner, 2016). In a numerical modelling study of the River Swale catchment, Coulthard and Skinner show that local as well as catchment-wide sediment yields were predicted to increase by orders of magnitude as rainfall data resolution increases. The study looked at the effects of rainfall spatial data resolution as well the temporal resolution of data, showing both to demonstrate a control over sediment fluxes in a catchment.

Numerical models of landscape evolution usually omit a realistic distribution of rainfall input in favour of uniform, homogenised precipitation across the landscape. When precipitation is ‘lumped’, either spatially or temporally in a catchment, local minima and maxima of precipitation are lost, and with discharge being a function of rainfall rate, this uncertainty propagates through to local discharges and erosion rates. The uncertainty in erosion rates is potentially exacerbated by the non-linearity and threshold dependence of erosive processes. The variability of precipitation is considered in many cases to be as important as total precipitation amount in determining erosional effectiveness (Tucker and Bras, 2000; Tucker, 2010). What is currently lacking in landscape evolution studies is a fuller understanding of how landscapes erode during individual storms, and in particular how erosional processes are sensitive to the details of precipitation across a catchment.

In numerical models of landscape evolution, resolving the precise temporal and spatial details of rain storms and the hydrological response is often computationally prohibitive, especially over long timescales, and as such modellers have taken to using simpler parametrisations of storm characteristics, such as using simple stochastic models to generate rainfall inputs and rainfall timeseries (Eagleson, 1978; Tucker et al., 2001). In studies of long term landscape evolution, the sensitivity of landscapes to the spatial distribution of rainfall has been investigated to some extent – particularly the imprint of orographic precipitation on landscapes (e.g. Roe 2002; Anders 2008; Han and Gasparini, 2015).

The following questions are explored through the use of numerical modelling simulations;

- Are fluvial erosion and sediment transport processes sensitive to the details of precipitation distribution at the catchment scale during single storm events?
- Does the choice of erosional model operating within the catchment influence sensitivity to rainfall patterns?
- What are the implications of this for longer term landscape evolution?

In contrast to previous studies, this paper looks at the effects of individual severe rainfall events using a range of erosional end-member models. The study investigates

the sensitivity of catchment-scale erosion to the spatial details of severe rain storms – the agents of long term landscape evolution. Landscape response is investigated using a numerical landscape evolution model that incorporates a dynamic (non steady-state) water-routing component and a range of fluvial incision and sediment transport laws. A series of model experiments is presented to test how sensitive real landscapes are to the catchment-scale details of precipitation during intense rainfall events. The simulations are each based on selected severe storms in the UK occurring in the past decade, which left significant flooding, damage, and geomorphic change in their wake.

7.2 Method

7.3 Results

7.3.1 Catchment sediment flux

In contrast to water fluxes, sediment flux from the catchments were most sensitive to the sediment erosion parameterisation, rather than the spatial detail of rainfall inputs.. For all catchments and events simulated, sediment flux from the catchment was higher in the simulations using higher resolution rainfall input data. The patterns of peak sediment discharge also mirrored that of water discharge, with sediment flux peaking earlier in the simulations with higher resolution rainfall inputs.

7.3.2 Spatial variations in sediment and bedrock erosion

The main spatial variation seen in sediment erosion is seen in river channels, where shear stresses require to initiate erosion and sediment transport are greatest due to the flow of water. The amount of erosion in each simulation was highly sensitive to parameterisation choice of the sediment erosion and transport law, with the choice of rainfall input data (gridded vs uniform) being only a secondary controlling factor on erosion amounts, all other factors being equal. This behaviour was seen on all simulations, in Figure 7.3 and 7.4.

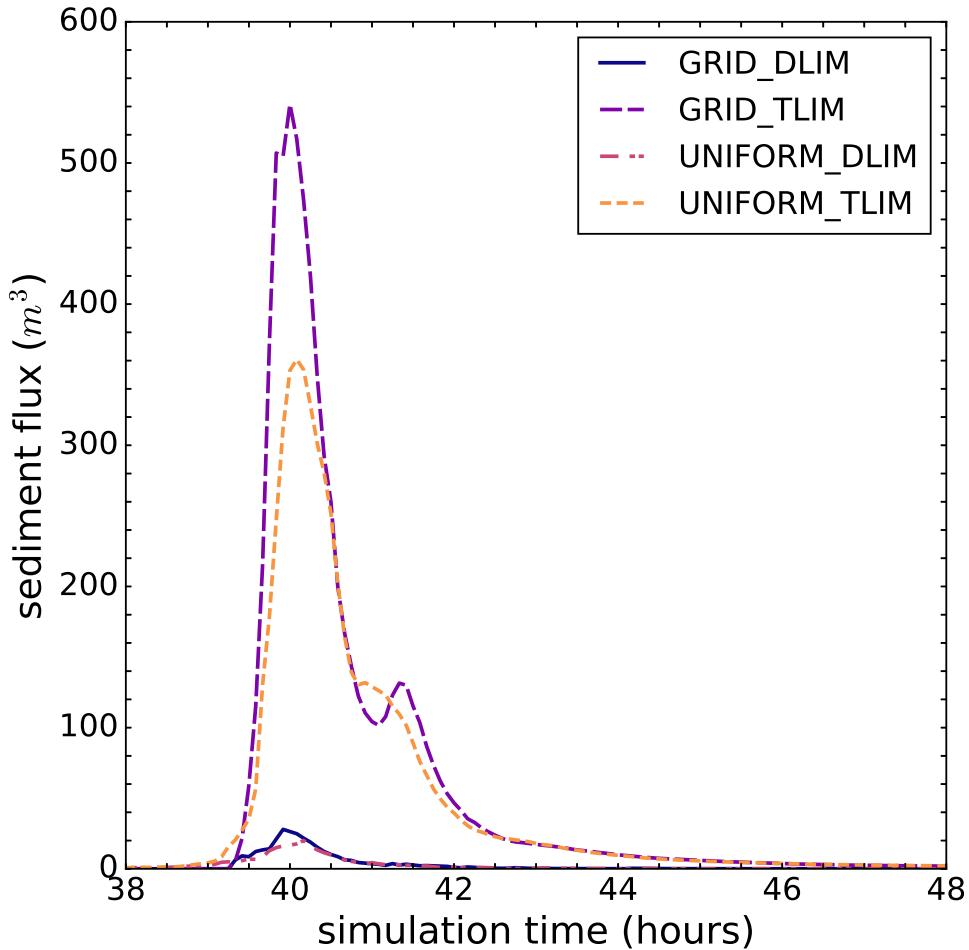


Figure 7.1: Boscastle sediment flux (total sediment volume output per hour at catchment outlet for each erosion-enabled simulation of the 2004 Boscastle event listed in Table 6.1.

7.4 Discussion

As with hydrology and flood extent (Chapter 6), sediment flux exhibits sensitivity to the input rainfall data resolution, but the more dominant sensitivity is to the sediment erosion parameterisation choice in the model. The set of erosion-enabled simulations in these experiments represented two end-members of sediment transport and erosion laws. The *TLIM*-suffixed simulations representing a purely transport limited environment and the *DLIM*-suffixed environment representing a purely detachment limited environment – with the transport limited law simulations predicting much greater sediment flux and erosion than the detachment limited counterparts. In terms of sediment flux and erosion, the role of rainfall input data spatial resolution played only a

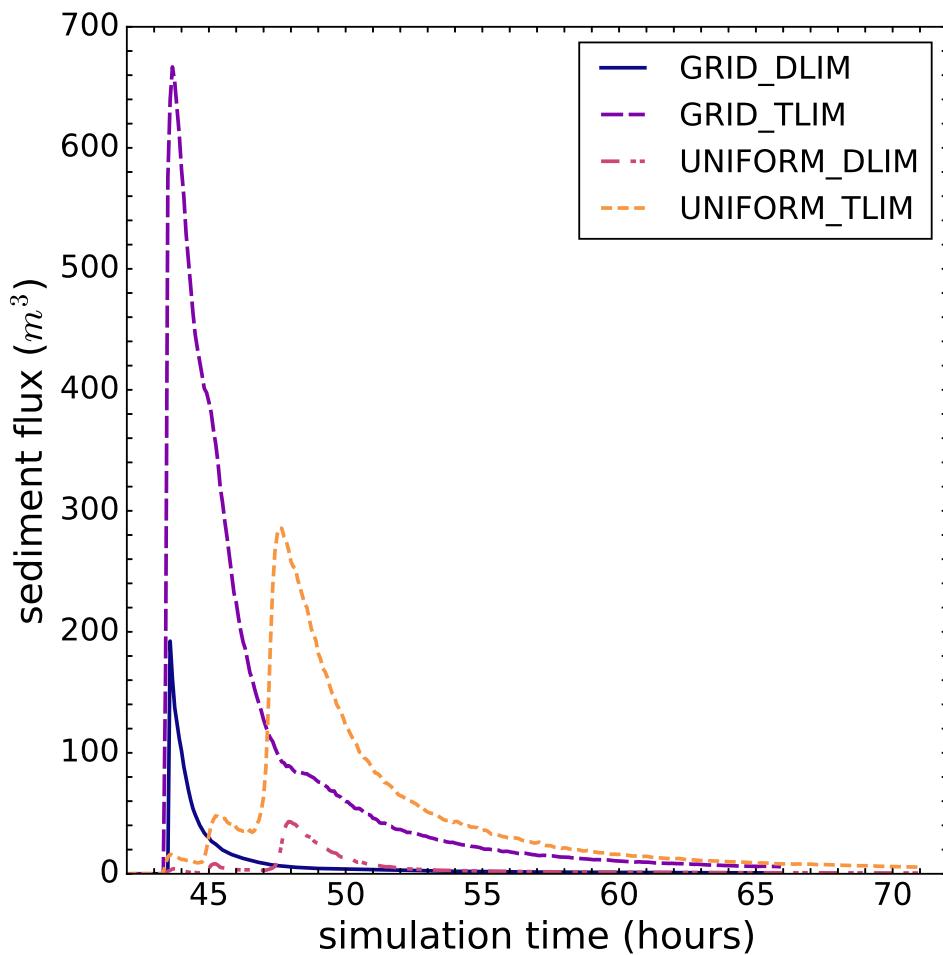


Figure 7.2: Ryedale sediment flux (Total sediment volume output per hour at catchment outlet) for each erosion-enabled simulation of the 2005 Ryedale event listed in Table 6.1.

secondary role in determining erosion amounts. The size of the catchment was less important in this respect, with even the smaller Boscastle catchment showing a marked sensitivity to the choice of erosion law.

7.5 Conclusion

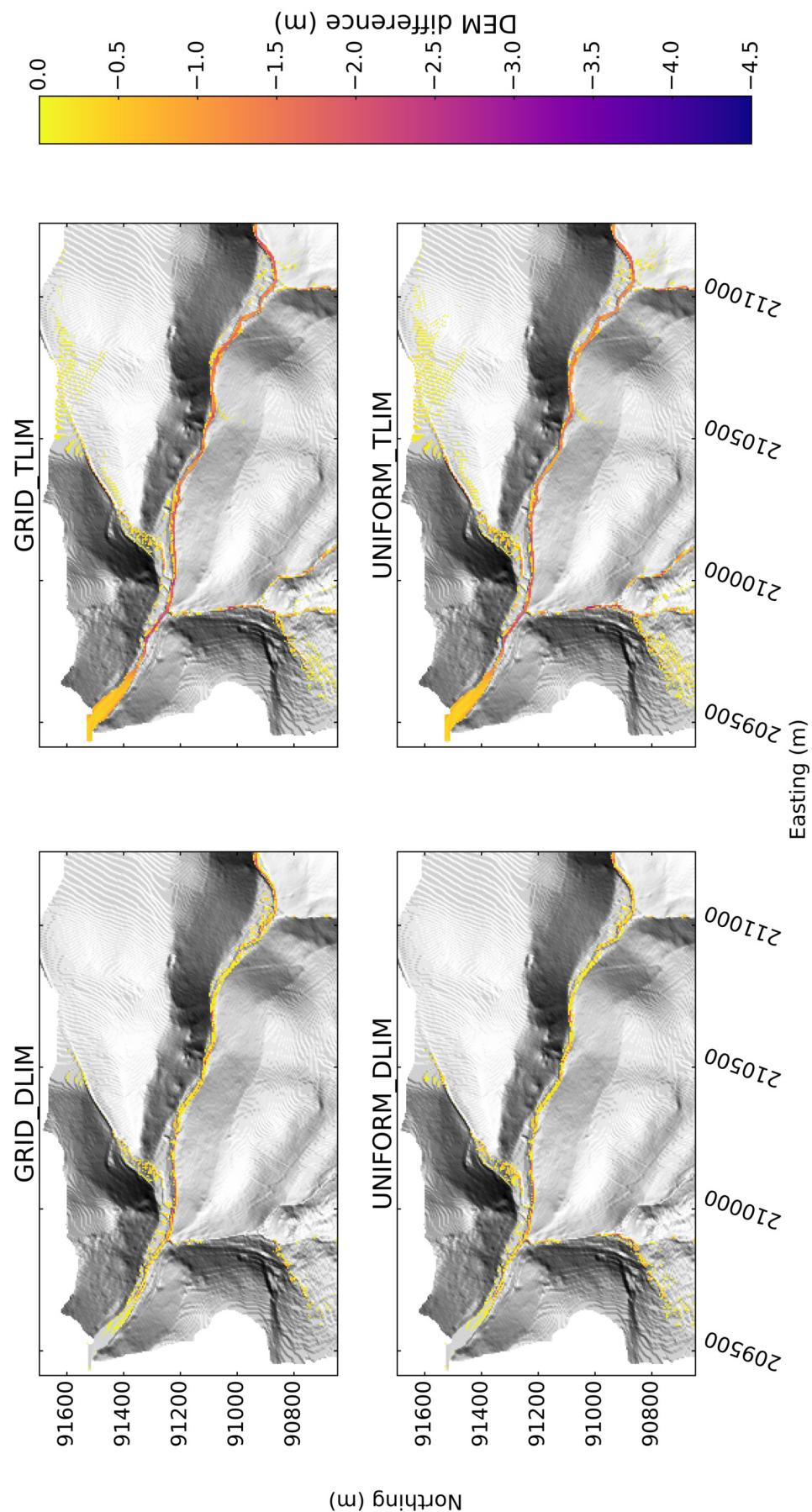


Figure 7.3: Boscastle. Erosion.

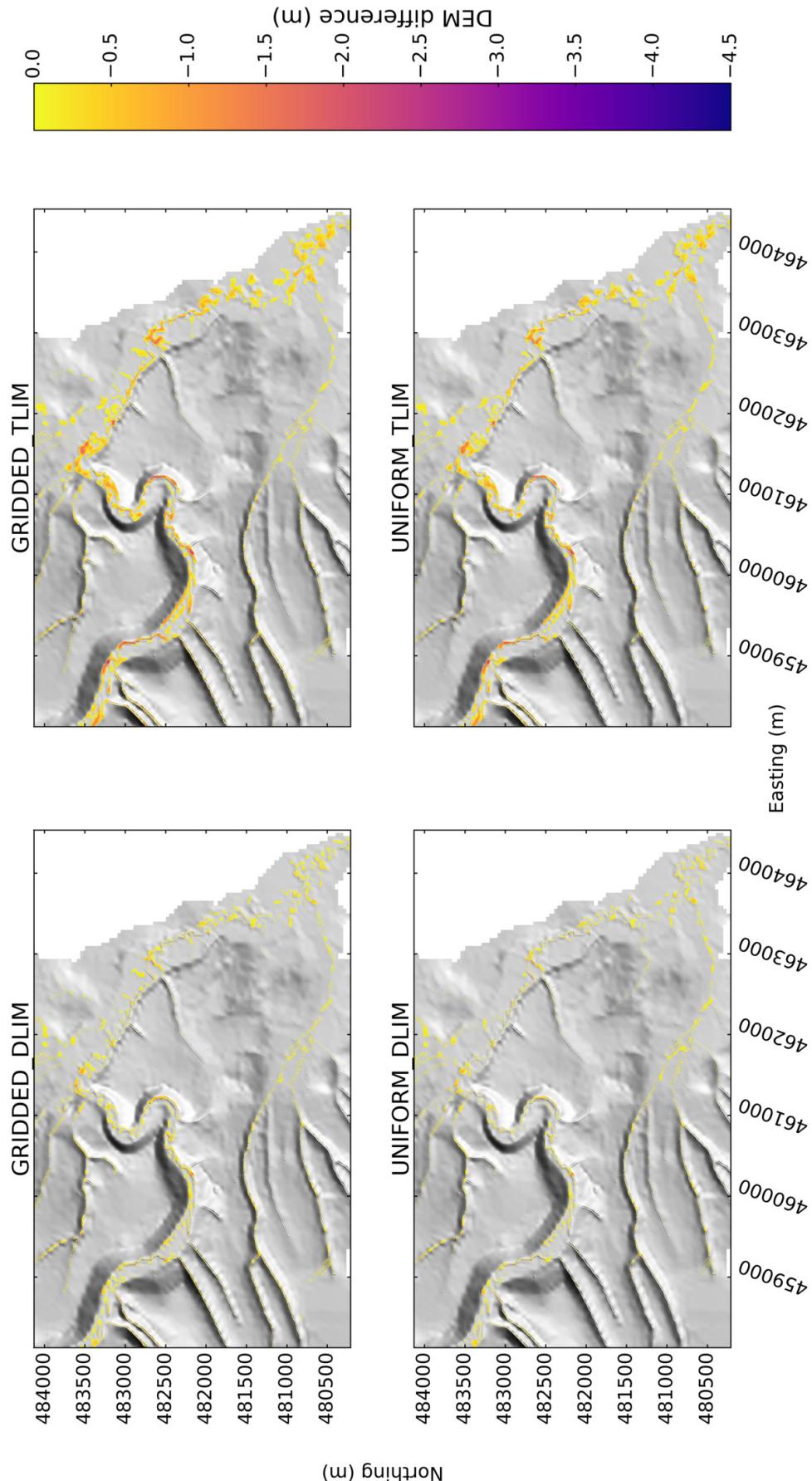


Figure 7.4: Ryedale. Erosion.

8

Conclusions

The spatial resolution of rainfall input in a river catchment simulation has a demonstrable effect on both hydrological and sediment fluxes, which is seen in both catchment-wide fluxes (Figures 6.2, 7.1, 6.3, 7.2). Though the magnitude of the sensitivity varied between experiments, there was a general increase in water fluxes, and hence sediment flux, in simulations with gridded rainfall data used as the rainfall input, compared to simulations that used a single, uniform value for rainfall. Existing studies investigating landscape evolution model sensitivity to rainfall resolution (Coulthard and Skinner, 2016) have noted similar increases in water discharge and sediment yields when using spatially heterogeneous rainfall data. The results from the experiments presented in this chapter are broadly in agreement with the general findings of Coulthard and Skinner (2016) although the timescales involved in each study are of different magnitudes. The results presented here suggest that the effect of rainfall resolution sensitivity applies to hydrogeomorphic processes at the single-event time scale, as well as landscape evolution over periods of hundreds and thousands of years.

Bibliography

- F Ahnert. Brief description of a comprehensive three-dimensional process-response model of landform development. *Z. Geomorphol. Suppl.*, 25:29–49, 1976.
- Gene M Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In *Proceedings of the April 18-20, 1967, spring joint computer conference*, pages 483–485. ACM, 1967.
- John Acock. A brief history of just-in-time. *ACM Computing Surveys (CSUR)*, 35(2):97–113, 2003.
- Paul D Bates, Matthew S Horritt, and Timothy J Fewtrell. A simple inertial formulation of the shallow water equations for efficient two-dimensional flood inundation modelling. *Journal of Hydrology*, 387(1):33–45, 2010.
- PD Bates, KJ Marks, and MS Horritt. Optimal use of high-resolution topographic data in flood inundation models. *Hydrological Processes*, 17(3):537–557, 2003.
- KJ Beven and Michael J Kirkby. A physically based, variable contributing area model of basin hydrology/un modèle à base physique de zone d'appel variable de l'hydrologie du bassin versant. *Hydrological Sciences Journal*, 24(1):43–69, 1979.
- Rafael L Bras, Gregory E Tucker, and Vanessa Teles. Six myths about mathematical modeling in geomorphology. *Prediction in geomorphology*, pages 63–79, 2003.
- A Casas, Gerardo Benito, VR Thorndycraft, and M Rico. The topographic data source of digital terrain models as a key element in the accuracy of hydraulic flood modelling. *Earth Surface Processes and Landforms*, 31(4):444–456, 2006.

- Michael Church. What is a geomorphological prediction? *Prediction in Geomorphology*, pages 183–194, 2003.
- Fiona J Clubb, Simon M Mudd, David T Milodowski, Martin D Hurst, and Louise J Slater. Objective extraction of channel heads from high-resolution topographic data. *Water Resources Research*, 50(5):4283–4304, 2014.
- Robert Colwell. The chip design game at the end of moore’s law. In *2013 IEEE Hot Chips 25 Symposium (HCS)*, pages 1–16. IEEE, 2013.
- T J Coulthard. Landscape Evolution Models: a software review. *Hydrological Processes*, 173(15):165–173, 2001.
- TJ Coulthard, MG Macklin, and MJ Kirkby. A cellular model of holocene upland river basin and alluvial fan evolution. *Earth Surface Processes and Landforms*, 27(3):269–288, 2002.
- Tom J Coulthard and Christopher J Skinner. The sensitivity of landscape evolution models to spatial and temporal rainfall resolution. *Earth Surface Dynamics*, 4(3):757, 2016.
- Tom J. Coulthard, Jeff C. Neal, Paul D. Bates, Jorge Ramirez, Gustavo a. M. de Almeida, and Greg R. Hancock. Integrating the LISFLOOD-FP 2D hydrodynamic model with the CAESAR model: implications for modelling landscape evolution. *Earth Surface Processes and Landforms*, 38(15):1897–1906, dec 2013. ISSN 01979337. doi: 10.1002/esp.3478. URL <http://doi.wiley.com/10.1002/esp.3478>.
- Leonardo Dagum and Ramesh Menon. Openmp: an industry standard api for shared-memory programming. *IEEE computational science and engineering*, 5(1):46–55, 1998.
- William E Dietrich, Dino G Bellugi, Leonard S Sklar, Jonathan D Stock, Arjun M Heimsath, and Joshua J Roering. Geomorphic transport laws for predicting landscape form and dynamics. *Prediction in geomorphology*, pages 103–132, 2003.
- Dean Gesch, Michael Oimoen, Susan Greenlee, Charles Nelson, Michael Steuck, and Dean Tyler. The national elevation dataset. *Photogrammetric engineering and remote sensing*, 68(1):5–32, 2002.

Brian Golding, Peter Clark, and Bryony May. The boscastle flood: Meteorological analysis of the conditions leading to flooding on 16 august 2004. *Weather*, 60(8):230–235, 2005.

Susan L Graham, Peter B Kessler, and Marshall K Mckusick. Gprof: A call graph execution profiler. In *ACM Sigplan Notices*, volume 17, pages 120–126. ACM, 1982.

John L Gustafson. Reevaluating amdahl’s law. *Communications of the ACM*, 31(5):532–533, 1988.

Mark D Hill and Michael R Marty. Amdahl’s law in the multicore era. *Computer*, 41(7):33–38, 2008.

Daniel EJ Hobley, Jordan M Adams, Sai Siddhartha Nudurupati, Eric WH Hutton, Nicole M Gasparini, Erkan Istanbulluoglu, and Gregory E Tucker. Creative computing with landlab: an open-source toolkit for building, coupling, and exploring two-dimensional numerical models of earth-surface dynamics, earth surf. *Earth Surface Dynam. Discuss.*, 10, 2016.

Alan D Howard and Gordon Kerby. Channel changes in badlands. *Geological Society of America Bulletin*, 94(6):739–752, 1983. doi: 10.1130/0016-7606(1983)94;739.

Valeriy Y Ivanov, Enrique R Vivoni, Rafael L Bras, and Dara Entekhabi. Catchment hydrologic response with a fully distributed triangulated irregular network model. *Water Resources Research*, 40(11), 2004.

Stefan J Kollet, Reed M Maxwell, Carol S Woodward, Steve Smith, Jan Vandeborgh, Harry Vereecken, and Clemens Simmer. Proof of concept of regional scale hydrologic simulations at hydrologic resolution utilizing massively parallel computer resources. *Water resources research*, 46(4), 2010.

Sriram Krishnan, Christopher Crosby, Viswanath Nandigam, Minh Phan, Charles Cowart, Chaitanya Baru, and Ramon Arrowsmith. Opentopography: a services oriented architecture for community access to lidar topography. In *Proceedings of the 2nd International Conference on Computing for Geospatial Research & Applications*, page 7. ACM, 2011.

Qiuhua Liang and Luke S Smith. A high-performance integrated hydrodynamic modelling system for urban flood simulations. *Journal of Hydroinformatics*, 17(4):518–533, 2015.

Charles C Mann. The end of moores law. *Technology Review*, 103(3):42–48, 2000.

Met Office. 1 km resolution uk composite rainfall data from the met office nimrod system, 2003. URL <http://catalogue.ceda.ac.uk/uuid/27dd6ffba67f667a18c62de5c3456350>.

Gordon E Moore. Cramming more components onto integrated circuits. *Proceedings of the IEEE*, 86(1), 1998.

Jeffrey Neal, Timothy Fewtrell, and Mark Trigg. Parallelisation of storage cell flood models using openmp. *Environmental Modelling & Software*, 24(7):872–877, 2009.

L Nicótina, E Alessi Celegon, A Rinaldo, and M Marani. On the impact of rainfall patterns on the hydrologic response. *Water Resources Research*, 44(12), 2008.

Stephen L Olivier, Allan K Porterfield, Kyle B Wheeler, Michael Spiegel, and Jan F Prins. Openmp task scheduling strategies for multicore numa systems. *International Journal of High Performance Computing Applications*, page 1094342011434065, 2012.

Paola Passalacqua, Tien Do Trung, Efi Foufoula-Georgiou, Guillermo Sapiro, and William E Dietrich. A geometric framework for channel network extraction from lidar: Nonlinear diffusion and geodesic paths. *Journal of Geophysical Research: Earth Surface*, 115(F1), 2010.

Frank J Pazzaglia. Landscape evolution models. *Development of Quaternary Science*, 1:247–274, 2003. doi: 10.1016/S1571-0866(03)01012-1.

Jon D. Pelletier. Fluvial and slope-wash erosion of soil-mantled landscapes: detachment- or transport-limited? *Earth Surface Processes and Landforms*, 37(1):37–51, jan 2012. ISSN 01979337. doi: 10.1002/esp.2187. URL <http://doi.wiley.com/10.1002/esp.2187>.

Jon D Pelletier, A Brad Murray, Jennifer L Pierce, Paul R Bierman, David D Breshears, Benjamin T Crosby, Michael Ellis, Efi Foufoula-Georgiou, Arjun M Heimsath,

Chris Houser, et al. Forecasting the response of earth’s surface to future climatic and land use changes: A review of methods and research needs. *Earth’s Future*, 3(7):220–251, 2015.

Bernhard Rabus, Michael Eineder, Achim Roth, and Richard Bamler. The shuttle radar topography mission—a new class of digital elevation models acquired by spaceborne radar. *ISPRS journal of photogrammetry and remote sensing*, 57(4):241–262, 2003.

Rizos Sakellariou. *On the quest for perfect load balance in loop-based parallel computations*. PhD thesis, University of Manchester, 1996.

Robert R Schaller. Moore’s law: past, present and future. *IEEE spectrum*, 34(6):52–59, 1997.

JM Schoorl, MPW Sonneveld, A Veldkamp, et al. Three-dimensional landscape process modelling: the effect of dem resolution. *Earth Surface Processes and Landforms*, 25(9):1025–1034, 2000.

Andrew M Sibley. Analysis of the north york moors storms—19 june 2005. *Weather*, 64(2):39–42, 2009.

Luke S Smith and Qiuhsia Liang. Towards a generalised gpu/cpu shallow-flow modelling tool. *Computers & Fluids*, 88:334–343, 2013.

Luke S Smith, Qiuhsia Liang, and Paul F Quinn. Towards a hydrodynamic modelling framework appropriate for applications in urban flood assessment and mitigation using heterogeneous computing. *Urban Water Journal*, 12(1):67–78, 2015.

Noah P. Snyder. Importance of a stochastic distribution of floods and erosion thresholds in the bedrock river incision problem. *Journal of Geophysical Research*, 108(B2):2117, 2003. ISSN 0148-0227. doi: 10.1029/2001JB001655. URL <http://doi.wiley.com/10.1029/2001JB001655>.

Xian-He Sun and Yong Chen. Reevaluating amdahl’s law in the multicore era. *Journal of Parallel and Distributed Computing*, 70(2):183–188, 2010.

Paolo Tarolli, J Ramon Arrowsmith, and Enrique R Vivoni. Understanding earth surface processes from remotely sensed digital terrain models. *Geomorphology*, 113(1):1–3, 2009.

Gregory Tucker, Stephen Lancaster, Nicole Gasparini, and Rafael Bras. The channel-hillslope integrated landscape development model (child). In *Landscape erosion and evolution modeling*, pages 349–388. Springer, 2001a.

Gregory E Tucker. Drainage basin sensitivity to tectonic and climatic forcing: Implications of a stochastic model for the role of entrainment and erosion thresholds. *Earth Surface Processes and Landforms*, 29(2):185–205, 2004.

Gregory E Tucker and Gregory R Hancock. Modelling Landscape Evolution. *Earth Surface Processes and Landforms*, 50:28–50, 2010. doi: 10.1002/esp.

Gregory E. Tucker, Stephen T. Lancaster, Nicole M. Gasparini, Rafael L. Bras, and Scott M. Rybarczyk. An object-oriented framework for distributed hydrologic and geomorphic modeling using triangulated irregular networks. *Computers & Geosciences*, 27(8):959–973, October 2001b. ISSN 00983004. doi: 10.1016/S0098-3004(00)00134-5. URL <http://linkinghub.elsevier.com/retrieve/pii/S0098300400001345>.

Declan A Valters. Modelling geomorphic systems: Landscape evolution. In *Geomorphological Techniques*, volume 6, pages 5–12. British Society for Geomorphology, 2016.

Marco J Van De Wiel, Tom J Coulthard, Mark G Macklin, and John Lewin. Embedding reach-scale fluvial dynamics within the caesar cellular automaton landscape evolution model. *Geomorphology*, 90(3):283–301, 2007.

Robert A Warren, Daniel J Kirshbaum, Robert S Plant, and Humphrey W Lean. A ‘boscastle-type’quasi-stationary convective system over the uk southwest peninsula. *Quarterly Journal of the Royal Meteorological Society*, 140(678):240–257, 2014.

P Wass, D Faulkner, and A Curini. An investigation into the north yorkshire floods of june 2005. *JBA Consulting, Skipton*, 2008.

Kelin X Whipple and Gregory E Tucker. Dynamics of the stream-power river incision model: Implications for height limits of mountain ranges, landscape response timescales, and research needs. *Journal of Geophysical Research: Solid Earth*, 104(B8):17661–17674, 1999.

Peter R Wilcock and Joanna C Crowe. Surface-based transport model for mixed-size sediment. *Journal of Hydraulic Engineering*, 129(2):120–128, 2003.

Marc H Willebeek-LeMair and Anthony P. Reeves. Strategies for dynamic load balancing on highly parallel computers. *IEEE Transactions on parallel and distributed systems*, 4(9):979–993, 1993.

Garry Willgoose. Mathematical Modeling of Whole Landscape Evolution. *Annual Review of Earth and Planetary Sciences*, 33(1):443–459, may 2005. ISSN 0084-6597. doi: 10.1146/annurev.earth.33.092203.122610. URL <http://www.annualreviews.org/doi/abs/10.1146/annurev.earth.33.092203.122610>.

W Eric Wong, Joseph R Horgan, Saul London, and Hiralal Agrawal. A study of effective regression testing in practice. In *Software Reliability Engineering, 1997. Proceedings., The Eighth International Symposium on*, pages 264–274. IEEE, 1997.

Appendix A

Linking high-resolution NWP model output with a landscape evolution model

A.1 Introduction

An introduction to the need for better forecasting of flash flooding events. Societal impacts etc. References to recent events in the UK. Rapid geomorphological change can accompany such events, leading to unexpected hydrological outcomes as most models assume static terrain surface during flood-inundation prediction.

Weather forecast models allow high resolution simulations to forecast spatial details of precipitations. Potential forecasting capability when coupled to hydrodynamic landscape evolution model. Relate to summer convective events which tend to be much more spatially focused (mesoscale) and hence rainfall inputs can vary significantly across a catchment. NWP offers greater lead times than radar ‘nowcasting’.

Can cite Pitt review (2008).

Need to investigate better linking of environmental models. E.g. NWP to Land Surface Flooding and Erosion models.

This chapter presents a framework for driving a landscape evolution model with output from NWP simulations. The framework is based upon further enhancements made to the HAIL-CAESAR model, described initially in Chapter 5. The enhancements are described in the following section and then tested by the application of

the model framework to test cases from intense rainfall events in the UK, previously investigate in Chapter ??.

The work investigates whether a one-way coupled model can provide better forecasting ability over a model driven with coarser resolution rainfall radar data.

A.2 Modelling framework

A.3 Method

Two WRF simulations.

- Boscastle (2004)
- Ryedale (2005)
- Both events triggered flash floods, and the flooding was exacerbated by the mobilisation of sediment during the flood, leading to blocked river channels/channels with reduced capacity.

Model set up:

- Four domains, with the innermost domain centred on the river catchment of interest with a horizontal grid spacing of 200m. Outer domains have resolutions of 25km, 5km, and 1km.
- Initialised with ECMWF data. (ERA-20C).

Figure: WRF domain set up

Table: WRF parameters for each simulation

A.3.1 Modifications to the CAESAR-Lisflood model

Description of model modifications to enable ingestion of high resolution rainfall data. I.e. describe the new rainfall runoff generation model. How it differs from the standard sem-distributed model (TOPMODEL) found in CAESAR-Lisflood.

A **ffigure** would probably be useful here to aid the description of the model component.

A.3.2 Landscape evolution model set up

No need to repeat the general model description in Chapter 5, just refer back. List the parameters for each simulation in a table though.

A.3.3 Modelling framework

Figure: flowchart showing the set up of the two models and their integration. Brief description here.

A.4 Results

A.4.1 NWP modelling results

Show a figure of the rainfall outputs over the inner domain and overlain over the river catchment(2 figures, one for each case)

A.4.2 Landscape evolution/hydrology modelling results.

Similar figures as to previous chapter of flood inundation and erosion spatial distribution.

A.5 Discussion

Discuss differences with the rainfall radar chapter previously. Does 200m NWP simulations produce any noticeable differences with the 1km gridded radar inputs and the uniform rainfall inputs.

Appendix B

Code availability

Appendix C

Key code and algorithms for the cellular automaton LEM

Appendix D

Configuration of the WRF model for the simulation of the North York Moors storm

APPENDIX D. CONFIGURATION OF THE WRF MODEL FOR THE SIMULATION OF THE

.....