



Ecole Nationale Supérieure
Agronomique de Rennes



Institute for Resources
and Environment

Mémoire de fin d'études (septembre 2002):

*DESIGN AND IMPLEMENTATION OF A GIS-BASED HYDROLOGICAL MODEL.
A CASE STUDY USING ARCVIEW 3.2 FOR A SEMI URBANISED WATERSHED IN
CHILLIWACK CITY, BRITISH COLUMBIA, CANADA.*

by

Luc SOREL

A thesis submitted in partial fulfilment of the requirements for the degree of

Diplôme d'Agronomie Approfondie en
Physique des Surfaces Naturelles et Génie Hydrologique

Acknowledgements

First of all, I would like to thank Hans Schreier, my supervisor, without whom, nothing of this great experience would have happened. Beside the guidance, the encouragement, the wise advice and research freedom he always provided me, I would like to underline his trust in a student he did not know beforehand and his great human and scientific ideals. To John Wiley and the City Hall of Chilliwack who provided guidance and data.

I also would like to thank all the people of the Institute for Resources and Environment for their great welcome, their daily smile and good spirit. A special thank you to Gina and Aileen for always helping me in trying times (particularly when I was looking for THE proper word or expression!), to Stefanie (my best –and only- unofficial student in French), to Sandra (remember, don't talk too fast!), to Lea, to Barbara for sharing lunches at the Rose Garden, to Jennifer and Jamie, to Yeshey for the few words of Buthanese I managed to remember and to Les, the coordinator of this crazy small world.

I also would like to thank all my roommates for being so great. Thank you Kris for transmitting me your unicycle virus (though I blame you for all my leg, hand and elbow scratches) and sharing geography discussions, to Sean who started my cinema career by proposing me to act with the Yeti, to Jenny and Chris for their friendly talks, few trips together in the city or in the mountains (I wish we did more), to Shannon and Shannon. To the new roommates who have just arrived: Alexis, Irmela and William (thank you for reviewing parts of this report). And a big bone to Kimo (proper orthography unknown, but this huuuuuge dog comes when I say that). Thank you to all these people. With them, more than a house, I found another home.

Quelques mots en français aussi... pas d'jaloux!

Je voudrais également remercier Christophe Cudennec, mon tuteur de stage pour m'avoir soutenu dans mon initiative et pour m'avoir encouragé à vivre cette expérience unique. Merci aussi au petit monde sympathique du Génie Rural de Rennes pour l'ambiance familiale. Je tiens aussi à exprimer ma reconnaissance pour mon école, l'ENSAR, et l'aide matérielle qui m'a été offerte.

Merci à toute ma famille, Maman, Papa et Stéphane pour leurs encouragements, la chaleur de leur cœur et leur soutien.

Cermi à Tamias roup toutes tes ssssshrouffisseries et tes photos de kiwiland, merci à PJ, Ibaselle, Adel, Thieum et Franck pour vos fréquentes nouvelles.

Merci à Phine d'être là, de donner une nouvelle dimension à mon bonheur, de la couleur à ma vie et un la majeur 11# (pour ceux qui ne savent pas, c'est mon accord préféré, lorsqu'il s'agit de jouer quelque chose de joyeux) à ma musique.

Un dernier mot, mais pas une dernière pensée, à Grand-Mère, qui nous a quittés récemment mais que je revois souvent dans mon cœur, merci pour toute ta générosité et ta tendresse.

Table of contents

ACKNOWLEDGEMENTS	2
TABLE OF CONTENTS	3
LIST OF FIGURES.....	5
INTRODUCTION	6
I) INTERCEPTION DITCH WATERSHED CHARACTERISTICS	7
I.1) GENERAL TOPOGRAPHY	7
I.2) SOILS.....	8
I.2.1) <i>Soils of the mountainous area</i>	8
I.2.1.1) Orthic Humo Ferric Podzols.....	8
I.2.1.2) Rego humic gleysols	9
I.2.2) <i>Soils of the valley area</i>	9
I.2.2.1) Eluviated Eutric Brunisols.....	9
I.2.2.2) Gleyed Gray Luvisols.....	9
I.2.2.3) Rego Gleysols	9
I.2.2.4) Orthic Regosols	9
I.2.2.5) Terric Humisols and Mesisols	10
I.3) LAND USE AND HUMAN ACTIVITIES	11
I.3.1) <i>An agricultural valley</i>	11
I.3.2) <i>A residential and forested mountainous area</i>	11
II) DESIGNING THE MODEL	13
II.1) THE SCALING ISSUE	13
II.2) WHY A RAINFALL-RUNOFF APPROACH?	14
II.2.1) <i>Urbanisation impact on vegetation interception</i>	14
II.2.2) <i>Urbanisation effect on groundwater</i>	14
II.3) RUNOFF GENERATION, THE PRODUCTION FUNCTION	15
II.3.1) <i>Precipitation gradient</i>	15
II.3.2) <i>Vegetation interception</i>	15
II.3.3) <i>Runoff generation</i>	17
II.3.4) <i>Impervious areas</i>	18
II.3.5) <i>Frame of the production function and limits</i>	20
II.4) RUNOFF ROUTING, THE TRANSFER FUNCTION.....	21
II.4.1) <i>The kinematic approach</i>	21
II.4.2) <i>The Unit Hydrograph approach</i>	22
II.4.2.1) What is a Unit Hydrograph	22
II.4.2.2) Cross-summing of the unit hydrograph.....	23
II.4.3) <i>Frame of the transfer function</i>	25
III) IMPLEMENTATION OF THE UNIT HYDROGRAPH MODEL WITHIN ARCVIEW3.2.....	26
III.1) DATA STRUCTURE AND MANAGEMENT IN ARCVIEW3.2	26
III.1.1) <i>Vector data: shapes</i>	26
III.1.2) <i>Raster data: grids</i>	26
III.1.3) <i>Tables: Vtab, Ftab, LUTable</i>	26
III.1.4) <i>Data management</i>	27
III.1.4.1) Data display and identification	27
III.1.4.2) User interaction on data	28
III.1.4.3) Calculation with grids.....	28
III.1.4.4) Conversion between shapes and grids.....	28
III.2) IMPLEMENTATION OF THE PRODUCTION FUNCTION	29
III.2.1) <i>Land use files</i>	29
III.2.1.1) Waterbodies	29
III.2.1.2) Vegetation features	30
III.2.1.3) Urban features.....	30

III.2.1.4) Soil features	31
III.2.2) <i>Implementing runoff generation</i>	34
III.2.2.1) Distributed imperviousness grid	34
III.2.2.2) Distributed interception grid	34
III.2.2.3) Soil water storage capacity distributed grid	35
III.2.2.4) Soil infiltration rate distributed grid	36
III.2.2.5) Distributed runoff grid	36
III.3) IMPLEMENTATION OF THE TRANSFER FUNCTION	37
III.3.1) <i>DEM-derived hydrological concepts</i>	37
III.3.1.1) Cardinal and orthogonal cells	37
III.3.1.2) Flow direction	38
III.3.1.3) Sinks	39
III.3.1.4) Contributing area, watershed delineation	39
III.3.1.5) Flow accumulation	40
III.3.1.6) Flow length	40
III.3.1.7) Slope	41
III.3.2) <i>Runoff routing direction</i>	41
III.3.2.1) DEM data set pre-processing	42
III.3.2.2) Flow direction grid processing	42
III.3.3) <i>Runoff routing speed</i>	46
III.3.3.1) Speed modeling	46
III.3.3.2) Distributed speed grid	48
III.3.4) <i>Drawing the time zone area diagram</i>	49
III.4) FIRST RESULTS	49
III.4.1) <i>Sensitivity tests</i>	50
III.4.1.1) The influence of soil water storage capacity	50
III.4.1.2) The influence of vegetation	51
III.4.1.2) The influence of imperviousness	51
III.4.2) <i>Confrontation with reality</i>	52
III.4.2.1) Origin of the data	52
III.4.2.2) First run	53
III.4.2.3) Second run	54
CONCLUSION	56
APPENDICES	57
APPENDIX 1 AGRICULTURAL LAND RESERVE	58
APPENDIX 2 INTERCEPTION STORAGE CAPACITY	59
APPENDIX 3 LOOKUP TABLES USED IN THE UNIT HYDROGRAPH MODEL	60
<i>Vegetation interception lookup table</i>	60
<i>Urban lookup table</i>	60
<i>Soil texture lookup table</i>	61
<i>Soil moisture lookup table</i>	61
APPENDIX 4 USDA SOIL CONSERVATION SERVICE CURVE NUMBER METHOD	62
APPENDIX 5 THE "AGRID.CON (YESGRID, NOGRID)" REQUEST	64
APPENDIX 6 THE CHILLIWACK TIN	65
APPENDIX 7 UHMODEL USER'S GUIDE	66
<i>Folder and subfolder organisation</i>	66
DataDefault	66
DataUser	67
Workspace	67
SimulationOutputs	67
Resources	68
<i>To run your first simulation</i>	68
<i>How to calibrate the model?</i>	70
REFERENCES	71
LITERATURE	71
ON-LINE REFERENCES	71

List of figures

Figure 1: localisation and decomposition of Interception Ditch watershed.....	7
Figure 2: inflexion between the flat valley (left) and the steep mountainous area (right)	8
Figure 3: typical draining ditch in the agricultural area.....	11
Figure 4: urban impact on soil imperviousness and vegetation cover (urbanised subwatershed)	12
Figure 5: hydrograph separation into “storm runoff” and “baseflow” components (after Hewlett, 1974)...	14
Figure 6: vegetation interception modeling.....	16
Figure 7: frame of the production function.....	20
Figure 8: flow description and parameters for mass and momentum conservation equations.....	21
Figure 9: construction and interpretation of an overland sheet flow unit hydrograph: a) a water sliced into 5 time zone areas (on a time step basis), b) the watershed covered by a unit overland water sheet divided into 5 water volumes (each volume recovering a time zone area)	23
Figure 10: frame of the transfer production.....	25
Figure 11: example of vector data	26
Figure 12: polygon attribute request processes: a) directly through the associated Ftab, b) by the means of a lookup table	27
Figure 13: example of cell-by-cell based grid computation	28
Figure 14: shape to grid conversion with values stored in a lookup table	29
Figure 15: articulation between urban shapes and the urban lookup table	31
Figure 16: soil polygons where the water table can be found (dark blue) across Interception Ditch	32
Figure 17: the Canadian System of Soil Classification textural triangle and texture class definition	33
Figure 18: horizon texture classes represented across Interception Ditch watershed	33
Figure 19: priority order for imperviousness grid combination.....	34
Figure 20: final imperviousness grid	34
Figure 21: cardinal and diagonal cells	38
Figure 22: flow direction values attributed to the processing cell	38
Figure 23: flow direction derivation from an elevation grid.....	38
Figure 24: delineating a watershed with a flow direction grid	39
Figure 25: flow accumulation, drainage network and divide derivation	40
Figure 26: downstream and upstream flow lengths.....	40
Figure 27: slope derivation on the mountainous part of the watershed with 2 methods (a: steepest descent, b: regression plane) and comparison. Dark pixels represent high slope values	41
Figure 28: location and depth of the sinks filled across Interception Ditch.....	42
Figure 29: derived natural drainage network and real ditch network	43
Figure 30: pseudo DEM of the valley based on the distance to the closest ditch	44
Figure 31: flow direction correction for upslope stream neighbouring cells.....	46
Figure 32: forces applied to a volume of water flowing on a sloped plane	47
Figure 33: location and delineation of the subwatershed used for sensitivity tests	50
Figure 34: influence of soil moisture on runoff contribution to discharge	50
Figure 35: influence of vegetation on runoff contribution to discharge	51
Figure 36: influence of vegetation on runoff contribution to discharge	52
Figure 37: runoff contribution hydrograph of the 26-27/10/2002 rainfall event, simulated by UHModel and the measured hydrograph, first run	53
Figure 38: runoff contribution hydrograph of the 26-27/10/2002 rainfall event, simulated by UHModel and the measured hydrograph, second and first runs.....	55

Introduction

This report will be submitted as the thesis requirement for the six-month practical session of the *Diplome d'Agronomie Approfondie en Hydrologie et Géomatique*, which I am currently completing at the *Ecole Nationale Supérieure Agronomique de Rennes*, France. It is an equivalent of a Master's thesis for an agronomical engineering degree in hydrology and geomatics.

My practical session took place within the Institute for Resources and Environment, located on and being part of the University of British Columbia Campus, Vancouver, Canada. The IRE is a teaching unit for graduate students and a research establishment that focuses on interactions between human activities and natural resources in their environmental context. A lot of research activities deal with compared integrated watershed managements in terms of vegetation, soil, water resources and various geographical and cultural backgrounds.

As far as British Columbia is concerned, Great Vancouver Regional District is by far the most dynamic and attractive area on the West Coast, in terms of both professional opportunities and recreational activities. Therefore, a lot of people (Canadians and immigrants) are moving in, but since Vancouver is already congested, urbanisation expands eastwards at an increasing rate in the Fraser River Basin, bringing urban environmental issues in a previously rural or natural background. The project in which I was involved concerns Chilliwack City, a community of approximately 70 000 people, located 100 kilometres east of Vancouver, thus directly concerned with increasing urbanisation. It is located on the south shore of the Fraser River, on the floodplain of Chilliwack River, a tributary of Fraser River. As Chilliwack City is a good option for people who want to live in this active area and still benefit from the mountainous background, the municipality forecasts a population growth of 10 000 people for the next 5 years, causing the building of 5 000 new houses.

But being such a mountainous province implies that British Columbia has only few fertile lands suitable for agriculture, which are located mainly in the flat valleys and are part of the governmental Agricultural Land Reserve (cf. **Appendix 1**), which protects them from being used as residential areas. Therefore, urbanisation in Chilliwack is likely to occur on the slopes of the local forested hills and mountains, causing huge changes in surface perviousness, storm runoff generation and routing processes. As paved surfaces are generally less pervious and smoother than usual vegetated soils, they produce more runoff during rainfall events and route them faster. Such land use leads to bigger stream discharges and faster watershed responses after rainfalls, implying higher flooding risks downstream in the valley, right where farmed lands and other residential areas are located.

The goal of my project was to design a prediction tool to assess hydrological impacts of land use changes, through modeling and predicting stream discharges during rainfall events. Such a hydrological model must be process-based as it has to account for many possible land use scenarios. Whereas many hydrological models offer just very little parameter modification, I wanted to implement the model within a Geographical Information System in order to make it very easy to input user-designed land uses.

After a brief description of the watershed and its hydrological characteristics, I will first explain the concepts used to design the model, then how they were implemented and how urbanisation artefacts on hydrological processes can be accounted for. Due to modeling and programming challenges, the content of my work has been very conceptual and only little calibration of the model has been done. However, I tried build the model in a way that user can easily modify most of the parameters. The last appendix is a user's guide for the Unit Hydrograph Model realised during my stay within the IRE.

I) Interception Ditch watershed characteristics

The case studied is Interception Ditch, a small watershed located south of Chilliwack City. The outlet is the point where Interception Ditch merges with Chilliwack Creek. The catchment of Interception Ditch covers 16,7 km², and its elevation ranges between 8 metres at the outlet and 674 metres for the mountains (cf. **figure 1**).

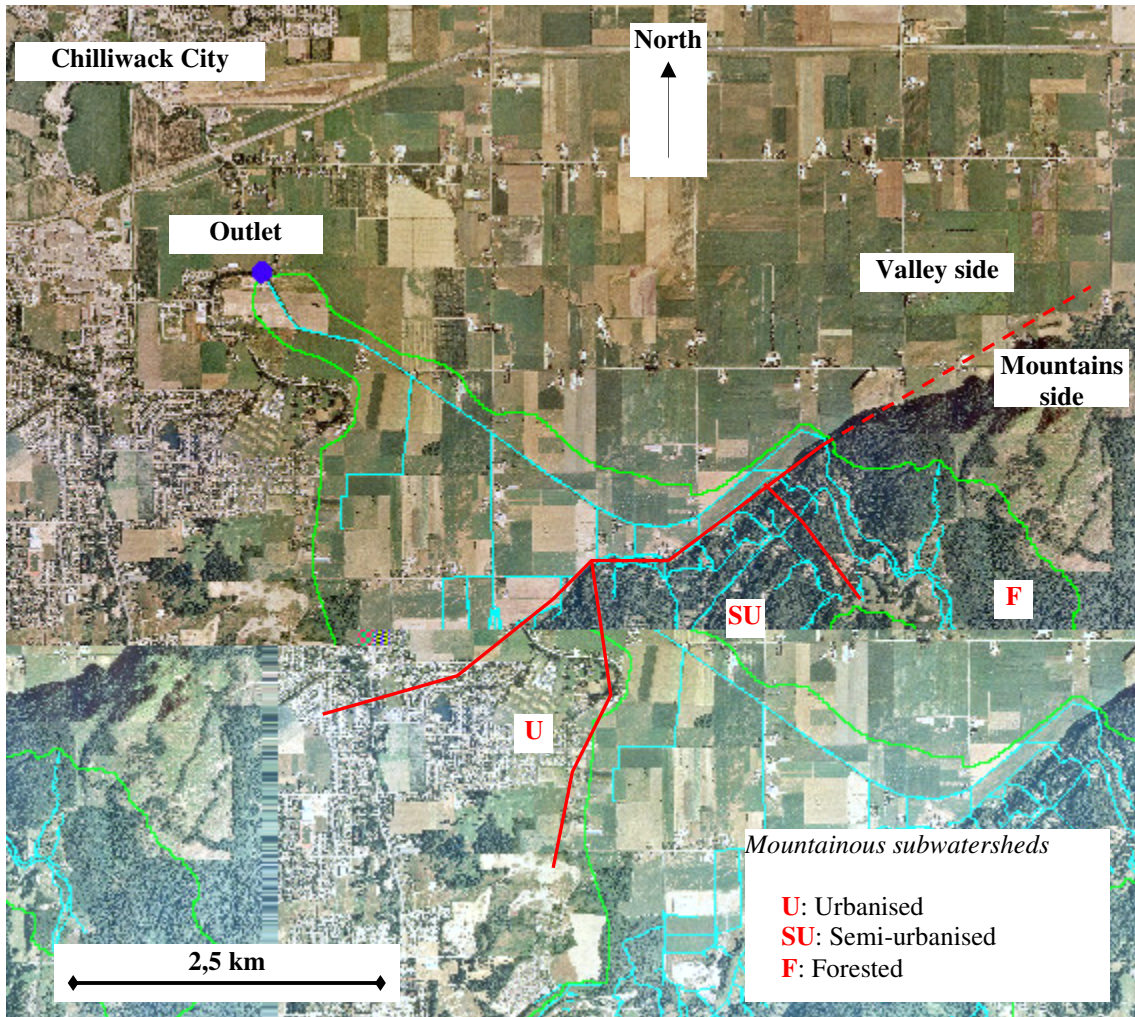


Figure 1: localisation and decomposition of Interception Ditch watershed

I.1) General topography

The watershed has a roughly triangular shape with the outlet as the north vertex, and can be divided into two areas of topographical relevance (cf. **figure 1**):

- the triangular valley north part, where the outlet is, accounts for one third of the whole watershed area (5,4 km²). Through thousands of years, sediments were deposited by Fraser River and have flattened the area so much that the river system is completely artificial and made of ditches dug to drain the soils. The elevation gain is hardly larger than 10 metres, so one can expect very small gravity gradient and slow stream flows.

- the trapezoid mountainous south part, comprising the headwaters, accounts for two thirds of the whole watershed area (11,3 km²). This section is more hilly, with slopes that can reach 62°. Mean slope is 14°, with a standard deviation of 9,4°. The river system is mainly composed of natural creeks, with several road ditches.

The transition between those two zones is quite abrupt in terms of slope, and occurs within a couple of metres (cf. **figure 2**). From a dynamical point of view, this slope inflexion means a huge change in terms of flow regime, since fast flowing water from the mountains is suddenly contained in almost flat ditches where it rather accumulates than flows. This detention effect is well known by Chilliwack City and this many floods have been recorded over this area during strong rainfall event.



Figure 2: inflexion between the flat valley (left) and the steep mountainous area (right)

I.2) Soils

Soil origins are very different between the valley and the mountainous parts. Whereas soils in the mountainous area derive from local bedrock and are not very developed, soils in the valley derive from sediments and show a more obvious layering and better-developed horizons. Here are the main soil types with some hydrological characteristics that can be found over the Interception Ditch watershed.

I.2.1) Soils of the mountainous area

There are 5 orthic humo ferric podzolic soil types and one gleyic type over the mountainous subwatershed.

[I.2.1.1) Orthic Humo Ferric Podzols

Orthic humo ferric podzols are typical soils in South British Columbia coniferous coastal areas under humid climate. They are characterised by the amorphous accumulation of humified organic matter and iron and aluminum in B horizon, which gives them a brownish colour. Podzolic soils typically occur in coarse- to medium-textured, acid parent materials.

Under virgin conditions these soils usually have L, F, and H horizons and may have an Ah horizon, of which absorption capacity increases water storage and slows down runoff flow.

Parts of the Podzolic B horizon may be cemented- and the underlying material may be cemented (duric), compact and brittle (fragipan), or friable. Therefore, the B horizon plays an important role in hydrological processes. If duric, it acts as an aquiclude for ground water.

[I.2.1.2) Rego humic gleysols

Though their parent materials are normally calcareous, humic gleysols in the Fraser lowland of British Columbia derive from noncalcareous glacial, marine and outwash deposits, and are often associated with Chernozemic, Luvisolic, Podzolic and Brunisolic soils. They are normally situated in the local depressional areas and pothole basins of moderately undulating to strongly rolling topography soils.

Gleysolic soils are defined on the basis of colour and mottling, which are considered to indicate the influence of periodic or sustained reducing conditions during their genesis. It results in rather impervious soils that are wet and generate runoff quickly due to poor draining conditions. These soils, here, have organic surface horizons derived from grass and sedge, moss, and forest vegetation.

Generally, soils in the mountainous part of the watershed are well- to moderately well- drained, except for the gleysols. Though rather young, most of the soils are developed enough to have a fair B horizon which is usually the most relevant horizon in terms of infiltration rate and storage capacity.

I.2.2) Soils of the valley area

Soil diversity is wider in the valley where Fraser River brought many types of sediment throughout its different regimes. The sediment top layers were deposited mainly during the last glaciation, some ten thousands years ago. Whereas soils in the mountainous area are shallow due to little development and erosion, soils in the valley are very likely to be thicker

[I.2.2.1) Eluviated Eutric Brunisols

These soils are developed mainly on acidic parent materials, and are found in the coastal areas of British Columbia with mild sub-humid to humid climatic conditions under mixed deciduous or coniferous forest. As those soils are more developed than regosols, and often associated with earthworms, they can be used for agricultural, as long as are drained.

[I.2.2.2) Gleyed Gray Luvisols

Gray Luvisols occur under forest-grassland transition zones and further in the valley in humid areas. They are developed mainly on glacial till, and glaciofluvial or lacustrine deposits. They differ from Orthic Gray Luvisols by having either distinct mottles that indicate gleying within 50 cm of the mineral surface, or prominent mottles at depths of 50-100 cm. Intense earthworm activity can lead to a mull type of Ah horizon, that gives the soil a high infiltration rate due to increased macropore structure.

[I.2.2.3) Rego Gleysols

These soils have the general properties specified than the aforementioned Gleysols properties (very poor drainage, quick runoff generation), though their B horizon is thinner than 10 cm. They thus consist of a gleyed C horizon, with or without organic surface horizons, and a thin Ah or B horizon.

As they occupy almost one half of the valley subwatershed, it explains why draining ditches had to be dug.

[I.2.2.4) Orthic Regosols

Regosolic soils do not contain a recognisable B horizon at least 5 cm thick and are therefore referred to as weakly developed. The lack of a developed pedogenic B horizon may result from any of a number of factors: youthfulness of the material, recent alluvium; instability of the

material, colluvium on slopes subject to mass wasting; nature of the material, nearly pure quartz sand; climate, dry cold conditions. Regosolic soils are generally rapidly- to imperfectly drained. They occur under a wide range of vegetation and climates.

[I.2.2.5) Terric Humisols and Mesisols

Humisols are composed largely of organic materials. Decomposition is at the most advanced stage than in any other type of soil. Most of the material is humified with few recognisable fibres, which makes it favourable for a high water storage capacity.

As the other organic soils, humisols are saturated with water for prolonged periods. These soils occur widely in poorly and very poorly drained depressions and level areas in regions of subhumid to perhumid climate and are derived from vegetation that grows in such sites.

Mesisols are at a less advanced stage of decomposition than humisols, for rubbed fibres can still be recognised.

Terric organic soils have a ferric layer (an unconsolidated mineral layer at least 30 cm thick) beneath the surface tier.

Globally, those soils are better suited for agricultural use, as long as they are properly drained. Soil infiltration rates (the speed at which water infiltrates a soil), water storage capacity (often assimilated with porosity, which is the percentage of void in a volume of soil) and soil moisture seem to be very diverse and quite distributed over the whole watershed. Spatial combination of soil characteristics, topography and moisture conditions obviously forms very different soil systems that influence the vegetation cover and the possible land uses.

Moreover, though it is not excluded that some cracks in the natural bedrock can offer subsurface pathways for groundwater, there has been no evidence of karstic systems in this area, which does not offer the geological conditions for such formations. Therefore, subsurface pathways may follow more or less the surface topography.

I.3) Land use and human activities

I.3.1) An agricultural valley

Land use in the valley is dominated by agricultural activities, which are quite intensive, as land reserved for agricultural production is rather scarce in BC. Most of the breeding units are dairy farms for which they grow corn and grass, followed by extremely dense chicken and pig units. Other fields are dedicated to ornamental plants for cities, Christmas trees and blueberries.

Most of the fields and the farms that are next to a ditch do not have any buffer zone at all, which makes the stream network very sensitive to any sort of manure or chemical drainage (cf. **figure 3**). Though there is no impervious area in the valley area such as residential lots, vegetation interception of rainwater is little compared to the one provided by the forest that covers most of the mountainous area.



Figure 3: typical draining ditch in the agricultural area

I.3.2) A residential and forested mountainous area

The mountainous area can be divided into three subwatersheds on the basis of land use (cf. **figure 1**):

- the west part of the mountainous area is urbanised mainly with individual residential lots and really wide roads (some of them are 7 metres wide, cf. **figure 4**). Such paved surfaces are very likely to generate lots of runoff. Between houses, the main vegetation is composed of grass and small bushes. Paved or not, the surface in this area is quite smooth and will not oppose too much friction to surface flow.
- the east part of the mountainous area is still almost completely forested. The forest is a dense mix of hardwood and coniferous trees that can intercept quite a lot of rainwater. The soil underneath is a rugged layer of litter through which runoff must flow rather slowly.
- the centre part is a semi-urbanised subwatershed, mainly forested.



Figure 4: urban impact on soil imperviousness and vegetation cover (urbanised subwatershed)

The expected urbanisation is to occur over the semi urbanised and the forested subwatersheds, causing huge changes in terms of soil perviousness and vegetation interception that can be already seen on the south mountainous subwatershed. Although the main hydrological changes will take place over those two mountainous watersheds, there will be huge consequences on the valley. The detention effect of the east-west transition line (cf. section **I.1**), combined with the increase of upstream soil imperviousness is likely to accelerate and increase the accumulation of storm water during rainfall event, with the risk of flooding fields and units, jeopardising farms, households, animals and crops. On the top of that, it increases the risk to wash out in the river huge amounts of stored milk and manure. However, I will not address the problem of pollution hazard in the model.

II) Designing the model

Though hydrology relies mainly on physical processes like vertical (infiltration) and horizontal flows (surface water and aquifer flows), hydrology is not an exact science due to too many unknowns. Description of water motions using mathematical laws soon becomes complicated because of the spatial variability of the media over or through which water flows. What modeling scale would match the needs of the prediction tool we are looking for? What concepts should it use?

II.1) The scaling issue

Scaling is one of the major issues in hydrological modeling because integration of more and more components has to come with the understanding of their succession and interaction. If it is rather easy to express vertical infiltration rate through a homogeneous layer of soils, it gets really complicated to do it through a discontinuous succession of soil horizons, which may not have the same initial moisture, and may be covered with vegetation and penetrated with roots. The same scaling problem can be found for flow velocity estimation when one switches to a place where we know exactly the local slope, to one where we only have a mean slope, since very different configurations can have the same mean slope.

Beside the theoretical challenges of expressing scaling complexity, lies the problem of data access. Even if we would be able to express an infiltration rate through different soil horizons, how can we access information on soil structure and texture? How do we account for seasonal behaviour of vegetation (crops, deciduous trees)?

It is obvious that assumptions on spatial distribution of hydrological media have to be made before one starts modeling. It is also important to set the right scale according to what phenomena or processes are considered meaningful, and describable.

As said in the introduction, the purpose of the expected model is to output a watershed discharge, knowing its catchment and the precipitation. Among the many rainfall-discharge models, the simplest one is the *Rational Method* single equation (1951 T.J. Mulvaney), which links the discharge peak of a stream (Q_p in m^3/s), with the catchment area (A in m^2) and the catchment-averaged maximum rainfall intensity (I in m of precipitation / s):

$$Q_p = C.A.I$$

Where C is a dimensionless number comprised between 0 and 1 and can be understood as a catchment-averaged *runoff coefficient*, or the percentage of rainwater that contributes in the storm hydrograph.

There are also many equations estimating storm response characteristics such as the *time to peak* (time from the beginning of precipitation until the first stream discharge peak) or the *time of concentration* (time from the beginning of precipitation until every point of the watershed is contributing to the stream discharge with runoff rate equalling rainfall rate) that rely on geometrical properties of the watershed.

Though those equations, often empirical, have some coefficients that reflect some physical processes, like the “ C ” runoff coefficient of the Rational Method, they cannot be easily parameterised for new scenarios with new land uses.

As urban items like roads or houses are small, compared to the size of a watershed, and as they do not a priori follow any spatial pattern, a distributed model has to be considered. A compromise

between the large dimension of a watershed and the narrowness of draining ditches or roads was taken and the watershed area was therefore digitised as a 10 metres by 10 metres square-cell grid for both land use and elevation data. This scale is big enough not to consider very high resolution phenomena, such as the influences of small bumps in the landscape, water routing along the roots or any disturbance due to very local irregularity that would be imperceptible in the storm hydrograph.

II.2) Why a rainfall-runoff approach?

The purpose of the model is to assess the effects of urbanisation on the hydrology of Interception Ditch watershed. I have already mentioned the increase of surface's imperviousness leading to more runoff generation and faster overland flows over smoother surfaces, but there are other effects of urbanisation on hydrological processes.

II.2.1) Urbanisation impact on vegetation interception

Though the clearcut of the forest definitely allows more runoff generation because the vegetation rainwater interception is bypassed (cf. **figure 4**), the effect on water infiltration in the soil is more ambiguous. In one sense the forest layer reduces the amount of rainwater that reaches the ground and returns part of the intercepted volume back to the atmosphere through evapotranspiration. But in the other hand, it also acts as a buffer between the clouds and the soils, as part of the water intercepted by the canopy slowly drips from the leaves or along the trunks. It basically spreads the rainfall event over a longer period of time, thus reducing the rainfall intensity and giving rainwater more time to infiltrate the soil.

However, this effect is highly dependent on the density of the canopy, the nature and the initial moisture of the soil, therefore only vegetation interception will be taken into account during the rainfall event.

II.2.2) Urbanisation effect on groundwater

However, if the surface is made more impervious, beside the fact that more runoff is generated during rainfalls, lies the fact that less water infiltrates the soil. Therefore, less groundwater is obviously stored in the soil, which is the long-term water reservoir for the baseflow of the stream between storm events and rainy seasons. This implies huge long-term consequences since every substance that infiltrates the soil is diluted in a smaller amount of water, causing pollution concentration especially during the summer when rainfalls get rare and tourists abundant.

Though pollution, water quality and water supply are issues as important, if not more, than flooding hazard, groundwater flow has been neglected because:

- it flows much slower than runoff, so that groundwater's hydraulic response usually occurs after the discharge peak. But as the rainfall event gets longer, newly infiltrated water adds pressure on the subsurface sheet of groundwater and sets it in faster motion, and the baseflow is believed to increase as follows:

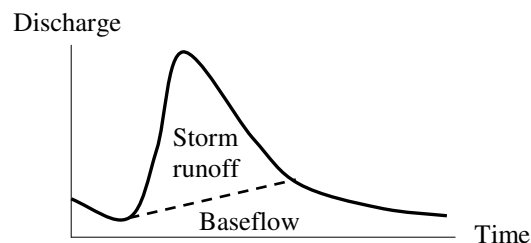


Figure 5: hydrograph separation into “storm runoff” and “baseflow” components (after Hewlett, 1974)

- whereas runoff flows over an observable surface, groundwater flow occurs in a 3-D environment which cannot be accessed and described exhaustively.
- moreover, possible equations to describe groundwater rely on strong assumptions and require a lot of computational resources that may not be available if the model is implemented within a GIS application, which is already running and using a lot of computer resources.

Therefore, the scope of use of the model is restricted to short-term storm events and to the estimation of runoff contribution to stream storm discharge. The stream baseflow is neglected, or assumed being a constant that the user can add to the output runoff hydrograph.

The computation of the runoff contribution to the outlet storm hydrograph can be divided in two logical and independent parts:

- the **runoff generation** from the rainfall input,
- the **runoff routing** to the outlet.

In the next two sections, I describe the concepts I used, their range of reliability and the restrictive assumptions on which they are based. The approach I undertook to design the model follows the “vertical” chronology of events that lead to a runoff hydrograph: precipitation, interception by land use (vegetation interception...), infiltration and runoff generation, overland flow routing to tributaries and the outlet, drawing of the hydrograph.

II.3) Runoff generation. the production function

Though the very place where runoff is generated is obviously the soil’s surface, the model first has to account for any obstacle between the clouds and the ground. Over Interception Ditch, the main cause of rainwater loss is vegetation interception by either forest, crops or grass. Therefore, the first step must be to convert gross precipitation to net precipitation by deducting water intercepted by the vegetation coverage.

II.3.1) Precipitation gradient

On large watersheds, which have several distinct valleys and high ridges between them, the topography can play an important role on spatial distribution of precipitation, mainly because of the rain-shadow effect of the mountains that stop the cloud progression on one side of the mountains, keeping the other one dry.

But Interception Ditch has not such topographical component, and its area is rather small, so I made the simplifying assumption that precipitation is distributed equally across the watershed. However, a precipitation gradient as a function of the topography can be easily added in the frame of the model by multiplying the mean precipitation height by a coefficient, which would be a function of the altitude, for each cell of the gridded watershed.

II.3.2) Vegetation interception

Once a rainfall occurs, the type, the extent and condition of vegetation influences the pattern of deposition and the amount of precipitation reaching the soil surface. Many experimental studies of vegetation interception have been done for forest canopies, and the processes and the rainwater reservoirs can be understood and identified as follow (cf. **figure 6**):

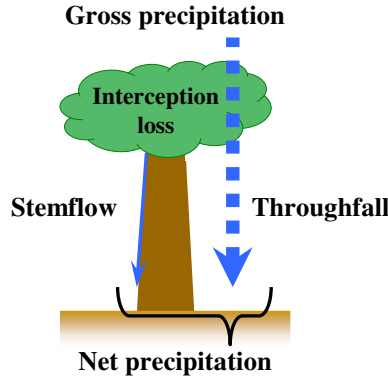


Figure 6: vegetation interception modeling

$$P_N = P_G - I_L = V_{Tf} + V_{Sf}$$

where P_N and P_G stands respectively for the amount of **net** and **gross precipitation** usually expressed in mm of rain and I_L is the amount of canopy **interception loss**. Rainwater actually falls through the canopy as:

- **throughfall**: V_{Tf} is the volume of throughfall (expressed in mm of rainwater), precipitation that passes through the vegetation canopy or drips from the foliage,
- **stemflow**: V_{Sf} is the volume of stemflow (expressed in mm of rainwater), intercepted water that flows down the stems or the trunks to the ground surface

Interception loss accounts for vegetation storage capacity, which is a rather small reservoir ranging from 0,2 to 2,5 mm of rainwater depending on vegetation type (cf. **Appendix 2**), and evapotranspiration. As simple models for throughfall and stemflow already exist, interception loss need not to be calculated.

Stemflow is a small flow that takes time to set up (depending mainly on the spatial organisation of the foliage and the roughness of the bark) and increase over time, but always remains a quantity smaller than 2% of gross precipitation. Though stemflow helps soil infiltration along the roots, its effect on runoff generation is little enough to be neglected in the model.

Throughfall is a faster process, which starts almost as soon as the rainfall event occurs because it implies a shorter contact between rainwater and vegetation. It is very easy to measure and many results have been reported in the form of global regression equations for a whole rainfall event as follow:

$$V_{Tf} = B_{Tf} \cdot P_G - C_{Tf}$$

where V_{Tf} and P_G are the total amounts (expressed either in m^3 or mm) of throughfall and gross precipitation during the rainfall event, and B_{Tf} and C_{Tf} are regression coefficients, the latter accounting for initial throughfall storage capacity and being often smaller than 2 mm of rainwater. This term is therefore ignored, which leaves us with the simple equation:

$$V_{Tf} = B_{Tf} \cdot P_G$$

where B_{Tf} represents the percentage value of gross precipitation that manages to fall through the canopy and actually reaches the soil. This global expression is used in the model at each time step

of the rainfall event to calculate net precipitation. Specific values for B_{Tf} were either found in the literature or estimated for each vegetation type.

Values found for vegetation infiltration rate in the literature (R.P.C. Morgan & R.J. Rickson, 1995) far exceeded soil infiltration rates, therefore any sort of precipitation delay caused by the coming through the vegetation layer was neglected in the model.

II.3.3) Runoff generation

The model used for runoff generation in the model is based on the *Hortonian model* (after R.E. Horton, 1875-1945) in which runoff is basically the rainwater that cannot infiltrate the soil either because the soil is saturated or because it cannot absorb rainwater fast enough. Whereas vegetation interception equation can be understood and used either on rainwater volume or intensity, soil infiltration and runoff generation concepts are dynamic and call for speed comparison (rainfall intensity and soil infiltration rate).

For the logical tests and the calculations that are done at each point of the gridded watershed at each time step of the rainfall event, 3 variables must be known:

- *soilInf*: the soil infiltration rate, which represents the maximum speed at which water can infiltrate the soil. It is expressed in mm/min, as a height of precipitation per minute.
- *soilWSC*: the soil water storage capacity, which represents the amount of water that a soil can store. It is expressed in mm, as a height of water.
- *rainInt*: the rainfall intensity, which is the height of net precipitation divided by the time step duration (in mm/min). *Tsd*, the time step duration is a constant throughout the simulation expressed in minutes.

Here are the logical tests and calculation for a single point at a single time step:

- **if** the soil is at water saturation, **then** the soil is completely impervious because it cannot absorb anything else and the amount of generated runoff equals the amount of rain that falls during the time step (*rainInt.Tsd*).

- otherwise, there can be 2 cases:

- **if** the rainfall intensity is smaller than the soil infiltration rate **then** the soil can absorb all the rainwater as long as the soil storage capacity is large enough. If the soil gets saturated within the time step, which happens if the amount of fallen rain exceeds the soil water storage capacity, then excess rainwater goes as runoff. Therefore, runoff is either 0 or the difference between soil storage capacity and the rainwater amount (*rainInt.Tsd-soilWSC*). This difference becomes positive when *rainInt.Tsd > soilWSC*
- **if** the rainfall intensity is larger than the soil infiltration rate **then** the soil can absorb only part of the rainwater. The difference between rainfall intensity and the soil infiltration rate is the runoff generation rate so the amount of runoff generated is (*rainInt-soilInf*).*Tsd* and the amount of water that infiltrates the soil is *soilInf.Tsd*. But here again, if this amount is bigger than the soil storage capacity (if *soilInf.Tsd > soilWSC*), the difference goes as runoff too.

In every case, the amount of runoff plus the amount of infiltrated water equals the amount of rainwater, which is the amount of net precipitation calculated before. The previous logical tests

can be mathematically expressed as follow (“Max(x,y)” returns the largest number between x and y):

```
if soilWSC=0 then runoff=rainInt.Tsd
elseif rainInt<soilInf then runoff=Max(0, rainInt.Tsd-soilWSC)
elseif rainInt>soilInf then runoff=(rainInt-soilInf).Tsd + Max(0, soilInf.Tsd-soilWSC)
```

At the end of the logical test series, one has to deduct the water infiltrated during the time step from the soil storage capacity before next time step calculation. The infiltrated water is the difference between the rain amount and the runoff amount:

$$newSoilWSC=soilWSC-(rainInt.Tsd-runoff)$$

All those calculations handle runoff and soil storage capacity variables expressed in terms of height and can be easily converted back into real volumes by being multiplied by the area of the square unit used to grid the watershed area.

II.3.4) Impervious areas

Impervious areas are not all totally impervious. Whereas parking lots and paved roads can be considered 100% impervious, a residential lot, which comprises a house and a garden, does allow some infiltration while it generates some runoff too. Such semi-impervious areas are characterised by a number, which represents the percentage of the area being totally impervious, the rest being considered by the model as basic grass. Vegetation interception by grass is taken into account only on the pervious part of the semi-impervious area.

As the rain amount is supposed equal across a cell, the same percentage of rainwater falls over an impervious surface and goes as runoff. Therefore the extra runoff volume due to impervious area equals:

$$extraRunoffVolume=rainInt.Tsd.cellArea.imperviousArea\%/100$$

The rest of rainwater actually reaches the soil and undergoes the logical tests and calculations described above. Though the horizontal area of the soil storage capacity volume is reduced, its depth is not affected. Likewise, rainfall intensity is not modified, neither is the height of runoff produced by infiltration excess. But later on, the height of runoff is converted into a volume by being multiplied by the area of the cell (10 metres by 10 metres), which should be corrected at this stage with the impervious area percentage as follow:

$$runoffVolume=runoffHeight.cellArea.(100-imperviousArea\%)/100$$

Actually, this correction can be done safely for the whole watershed, even over pervious area because their imperviousness percentage equals 0, so the computed runoff height will not be affected.

Those steps can be synthesised in a smaller amount of steps:

i computation of runoff height generated by infiltration excess

```
if soilWSC=0 then runoff=rainInt.Tsd
elseif rainInt<soilInf then runoff=Max(0, rainInt.Tsd-soilWSC)
elseif rainInt>soilInf then runoff=(rainInt-soilInf).Tsd + Max(0, soilInf.Tsd-soilWSC)
```

$$runoff = runoff \cdot (100 - imperviousArea\%) / 100$$

ii computation of runoff height generated by impervious area.

The height is an artificial height corrected by the impervious area percentage, which actually corrects the area when the height is converted back to volume

$$extraRunoffHeight = rainInt.Tsd.imperviousArea\% / 100$$

iii conversion of runoff height to runoff volume

$$runoffVolume = (extraRunoffHeight + runoff) \cdot cellArea$$

II.3.5) Frame of the production function and limits

The following drawing describes the steps to compute distributed runoff generation during one time step.

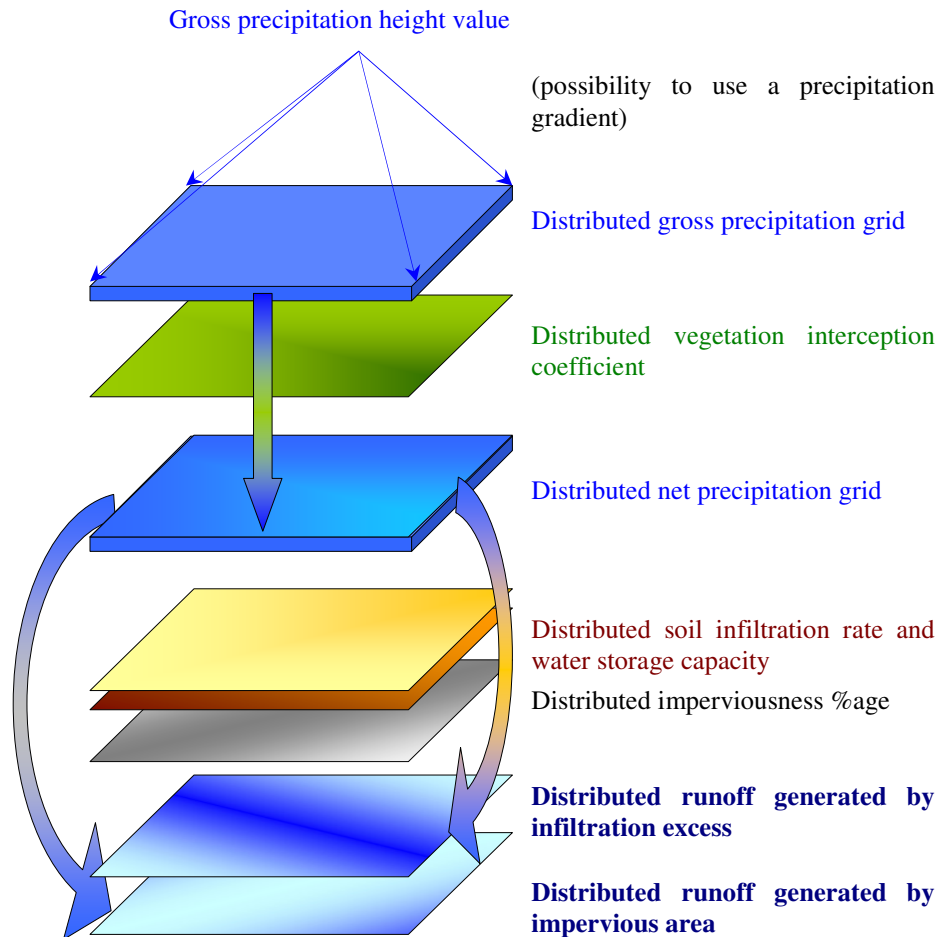


Figure 7: frame of the production function

Though the production function is quite process-based at the soil level, it does not account for other processes that may affect runoff generation and soil conditions:

- slope is likely to play an important role in runoff generation because on a steep slope, water would rather flow overland than stay still and infiltrate. Although the digitising scale (100 m²-cell grid) should be relevant enough to predict global flow routing direction, but probably does not match the soil infiltration processes scale. On the top of that, soil roughness (caused by soil texture and vegetation coverage) may slow down water and help infiltration. Because of scale relevance and the different parameters to consider, slope effect on infiltration was neglected.

- during the rainfall event, it is assumed that soil keeps on absorbing rainwater until it gets saturated. The model neglects the fact that infiltrated water actually moves in the soil as groundwater or subsurface water, thus leaving new room for more rainwater to infiltrate, and probably seeping downhill and contributing to runoff. As mentioned

before, there is a lack of knowledge about Interception Ditch underground flow, and seepage might compensate new soil storage capacity due to drainage.

II.4) Runoff routing. the transfer function

As far as overland and open channel flows are concerned, water moves thanks to the source of energy provided by the gravity gradient, in other terms, according to the local steepest slope. However, water may flow locally uphill due to its momentum, in such place like in tight curbs after a steep slope. This is again an effect dependent on the scale used to observe hydrological processes.

Because of the size of the watershed, the distance unit used here is large enough to neglect such artefacts and makes reasonable the assumption of using the topography as the only information for flow routing direction.

The speed at which water flows overland or in open channel mainly depends on slope and its roughness. The steeper a slope is, the more kinetic energy is given to water because of a higher gravity gradient. Meanwhile, the faster water flows over a rough surface, the more friction occurs and reduces flow speed. Whereas routing direction can be addressed easily, routing speed and water travel times to the outlet are obviously more difficult.

II.4.1) The kinematic approach

The physical concepts usually used to model channel flow routing are based on a mechanistic approach involving gravity, topography, ground roughness, momentum and mass balances. Barré de Saint-Venant (1797-1886) wrote a set of partial differential equations that express flow through local control volumes as a function of cross-sectional area and wetted-perimeter, mean depth, topographic and hydraulic slope, which require the following assumptions:

- water is incompressible and of constant temperature and density,
- velocity distribution across the cross-sectional area is supposed uniform, or is represented by an average velocity,
- the pressure distribution is supposed hydrostatic, which implies a non-turbulent flow,
- the momentum carried to the control volume streamflow by lateral inflow is negligible.

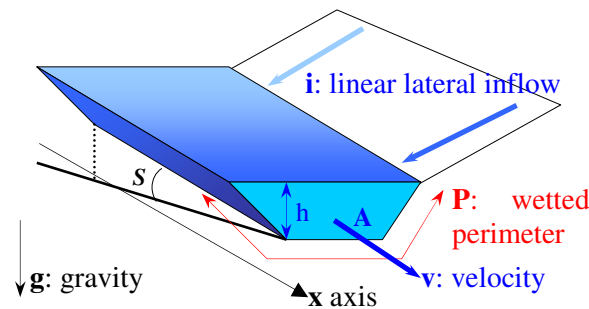


Figure 8: flow description and parameters for mass and momentum conservation equations

And the equations can be expressed as follow (quoted from Beven, 2001):

- conservation of mass

$$\frac{\partial Q}{\partial x} = i - \frac{\partial A}{\partial t}$$

- conservation of momentum

$$\frac{\partial Av}{\partial t} + \frac{\partial Av^2}{\partial x} + \frac{\partial Ag h}{\partial x} = gA(S - S_f)$$

where S_f is the friction slope, usually calculated as a function of f , the Darcy-Weisbach coefficient as:

$$S_f = \frac{P}{A} \frac{f}{2g} v^2$$

But those non-linear partial differential equations, which are already an approximation of the fully three-dimensional processes that occur in a stream channel, have no obvious analytical solutions. Moreover, they are designed for channel routing and further modifications have to be done for overland flow routing.

Although simplified equations and models can be derived from St-Venant's equations (diffusion and kinematic wave approximations), analytical resolution requires a finite element approach and a lot of calculations for more results that we need for our model. Our purpose is not to determine flow speed and flow height across the watershed, but just the travel time of each volume of runoff from their generation location to the outlet.

II.4.2) The Unit Hydrograph approach

Whereas water amount and water speed are combined in the kinematic approach, the so-called *unit hydrograph* theory separates those two issues, which makes the routing easier to do and to implement.

[II.4.2.1) What is a Unit Hydrograph

This theory applies to a sheet of water flowing overland to the watershed outlet. The idea is to split the catchment into zones on the basis of travel time to the outlet. Zone 1, which comprises the outlet itself, is the zone for which overland water reaches the outlet within one time step, zone 2 is the area for which water reaches the outlet between one and two time steps and so on.

Knowing the volume of water over each time zone areas, it is very easy to draw the hydrograph at the outlet, since volume 1, which is the volume of water over time zone 1, contributes to the hydrograph after one time step, volume 2 contributes after two time steps, and so on (cf. **figure 9**).

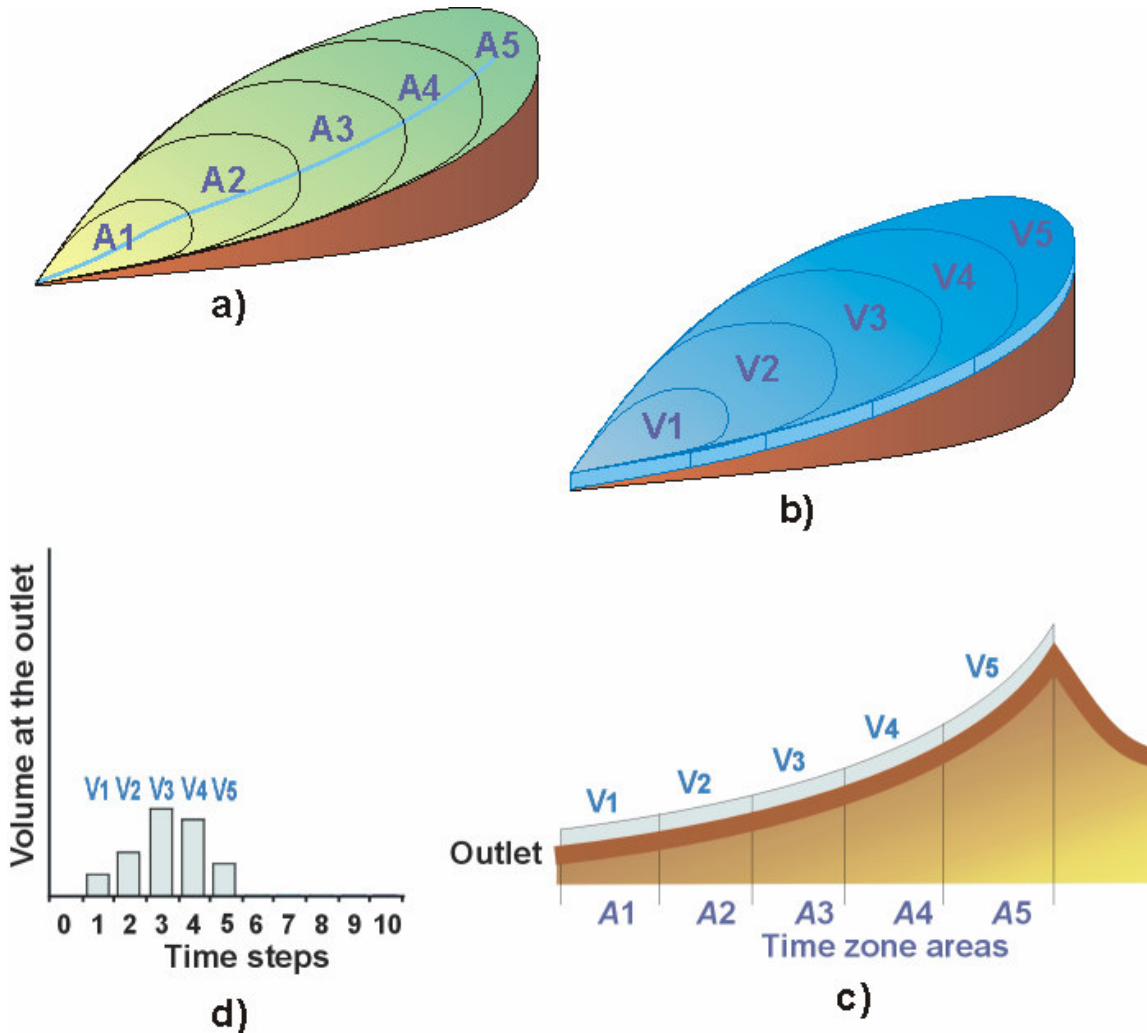


Figure 9: construction and interpretation of an overland sheet flow unit hydrograph: a) a water sliced into 5 time zone areas (on a time step basis), b) the watershed covered by a unit overland water sheet divided into 5 water volumes (each volume recovering a time zone area), c) a uphill-downhill cross-section of the watershed, d) the unit hydrograph

The splitting of the catchment extent into time zone areas can be done in many different ways, according to different kind of approaches. The one I followed requires will be described in the section III.3.4.

[II.4.2.2) Cross-summing of the unit hydrograph

The unit hydrograph theory is used with the runoff volume as input. During the *first time step*, the first layer of runoff is produced and flows one time zone down: TZA1 (time zone area 1) fully contributes to the hydrograph, TZA2 flows down into TZA1, TZA3 flows down into TZA2... and the last TZA flows down into the last but one TZA.

Then, during *time step 2*, the second layer of runoff is produced. V_{21} (volume of runoff produced over TZA1 during *time step 2*) adds to V_{12} (volume of runoff produced over TZA2 during *time step 1*) and both of them contributes to the hydrograph at time step 2, as each TZA flows into the lower one. At the end of *time step 2*, TZA1 is now filled with V_{13} (volume of runoff produced over TZA3 during *time step 1*) and V_{22} (volume of runoff produced over TZA2 during *time step 2*).

The same processes happen during *time step 3*, at the end of which V_{31} , V_{22} and V_{13} contribute to the hydrograph. As we can see with the index numbers, the amount of water that reaches the outlet at the end of a time step is made of different small volumes, and their numbering has the following simple mathematical property:

The volume produced during this time step on TZA1 adds to the volume produced by the next TZA (TZA2) one time step before, which adds to the next TZA (TZA3) one more time step before, and so on. In this way, contributing volume indexes of time decrease as contributing volume indexes of TZA increase.

The cross-summing of double-indexed volumes basically represents concentration of overland water towards the outlet. Moreover, as actual elementary water motions are not computed, cross-summing is very efficient in terms of modeling and computing resources. Nevertheless, it implies:

- generated runoff does not infiltrate downstream while flowing towards the outlet over soils that may not be saturated yet. Considering that upstream runoff arrival in a cell provides an extra water input that would increase rainfall intensity and therefore increase runoff generation can soften this strong assumption and make it more or less self compensating.
- existing runoff does not influence runoff generation for the next time step. The same explanation applies here. However, in some watersheds, it has been observed that runoff flow only a couple of metres overland before infiltrating the soil, and then return to surface again. In reality, overland flow is likely to be a mix of surface and subsurface flows, so concept of runoff generation does not really tell if the water that enters streams is old water or rainwater, but it seems to be good approximation of the stormwater volume reaching the outlet (Beven, 2001).

II.4.3) Frame of the transfer function

The following drawing describes how the runoff hydrograph is computed.

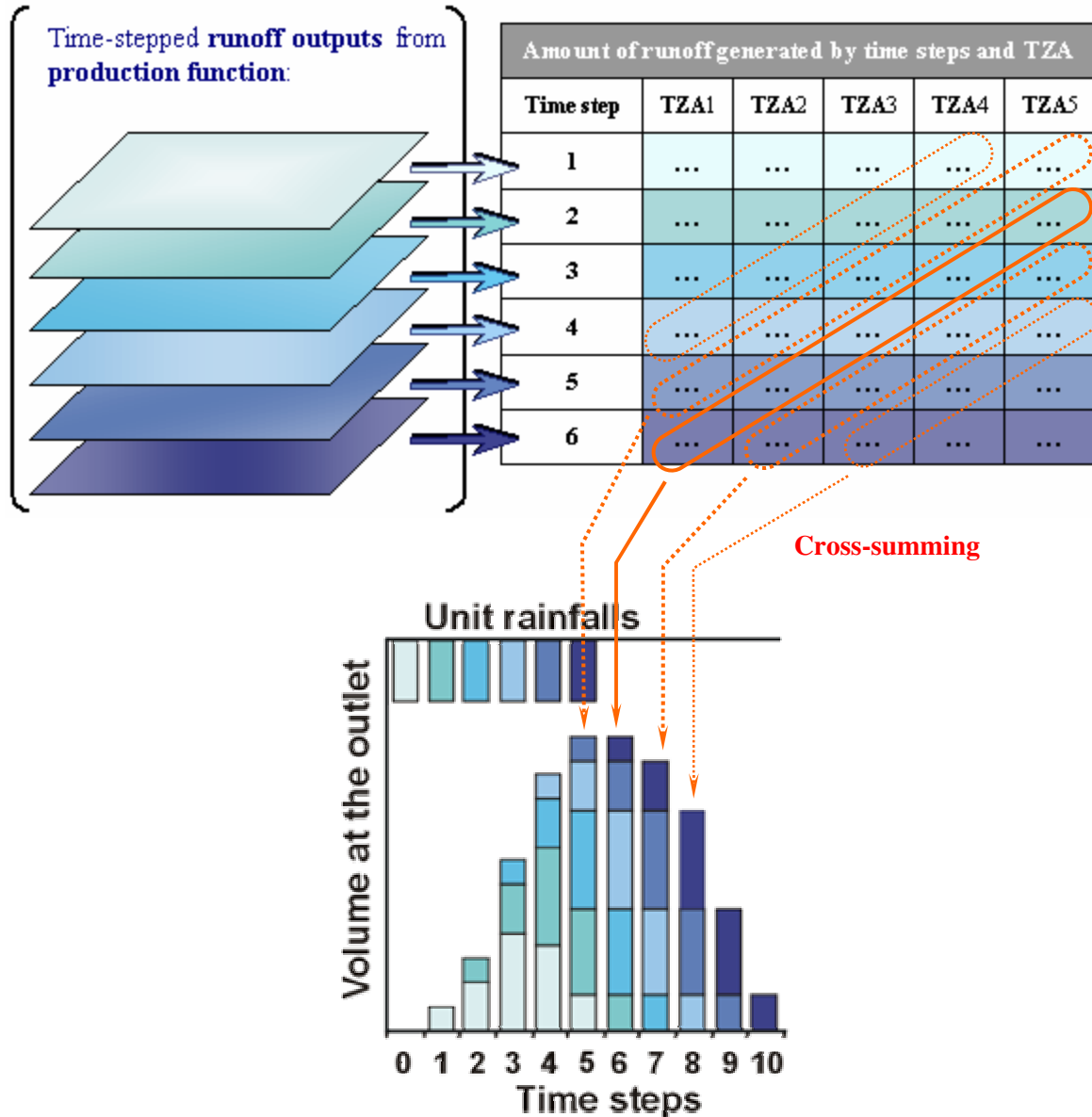


Figure 10: frame of the transfer production

Due to the cross-summing calculation, the **transfer function** cannot operate before all the runoff generation computations are completed by the **production function**.

Now that the modeling concepts are well established, the next section will present how I implemented them in ArcView3.2 GIS environment and how both the user and the model can interact with land use data and initial conditions.

III) Implementation of the Unit Hydrograph Model within ArcView3.2

A Geographic Information System is a hardware and software integrated package designed for the input, the storage, the management, the analysis and the display of data referenced by their spatial or geographic coordinates. Whereas spatial information like areas, lines and points are displayed as geometrical objects on a map, their attributes like size, name or any sort of information.

Although GIS-based modeling presents the obvious advantage that the software environment already provides the modeler with built-in tools and functions to process data, it is necessary to understand how data can be accessed and processed, and how one can develop new functions that fully integrate the model in ArcView.

III.1) Data structure and management in ArcView3.2

ArcView is a GIS software produced by ESRI, and as most of other GIS softwares, it can handle 2 types of geographical information: vector data and raster data.

III.1.1) Vector data: shapes

The basic component of vector data is point, located by its coordinates. Points linked together constitute a line or a polyline, and a polygon is a closed shape delineated by a line (cf. **figure 11**).

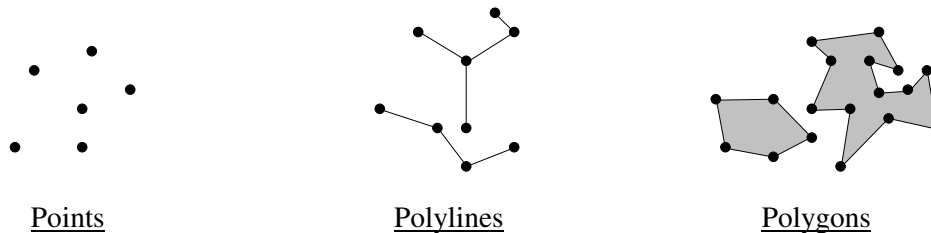


Figure 11: example of vector data

Vector data are easy to create in ArcView since one basically draws them on the map. As only the relevant information is drawn (points or *vertices* - which designates points used in lines and polygons), *shapefiles* (vector data files) are small files, easy to store and modify.

III.1.2) Raster data: grids

Raster data is a grid of cells covering an area of interest. Each pixel, the smallest unit of information in the grid, displays a unique attribute. Raster data can be scanned images or photographs, or a set of values for a distributed parameter such as temperature or elevation (in this case the grid is called DEM for *Digital Elevation Model*). Though ArcView grids can only store integers or decimal numbers, cells that are not attributed any value are given the *null* value, which means “no data” and is different from zero, which is considered as a usual number.

The large amount of information to be stored for one grid results in bigger files. Actually, ArcView stores grids as folders containing different files about the grid data itself and the projection system to be used when the grid is displayed.

III.1.3) Tables: Vtab, Ftab, LUTable

ArcView's tables are spreadsheets stored as standard *.dbf* files that can be opened in Excel too. Each column of a table is called a *field* and is identified by its name. Each row of the table is a *record* and is identified by a number. Fields can be of many types: integers, decimal numbers, text...

Every vector file is also associated with a table, of which the first field is of a special type: it is a *shapefield*, which contains displayable geometric information. A shapefield can be of three types (points, lines or polygons) that are exclusive, which means that one table can exclusively store geometrical object of only one of the three types. Other fields can be of any type. Such tables containing geometrical objects are called *Ftabs* for “feature table” whereas text and number tables are called *Vtabs*, for “value table”. Each record of an *Ftab* is therefore a single geometrical object, of which attributes are accessible through requests on the associated *Ftab*.

Instead of storing values directly in the *Ftabs* associated with the vegetation or urban polygon shapefiles, say for interception or imperviousness, I use lookup tables linked to the *Ftabs*. For example, each vegetation polygon has a “vegetation key” stored as an attribute in the *Ftab* field called “Veg_key”, and the vegetation *LUTab* (LookUp Table) has the same *Veg_key* field, which should contains only one occurrence of all the keys currently and possibly used. Among the other fields of the vegetation *LUTab*, one is the vegetation interception attribute and contains its value. Therefore, the process hereby one gets information about an attribute of a polygon gets more complicated to implement (cf. **figure 12**), but through using vegetation keys, one defines clear vegetation classes on the basis of attributes value, which makes sensitivity tests and calibration much easier since values subject to modification are restricted to the number of classes.

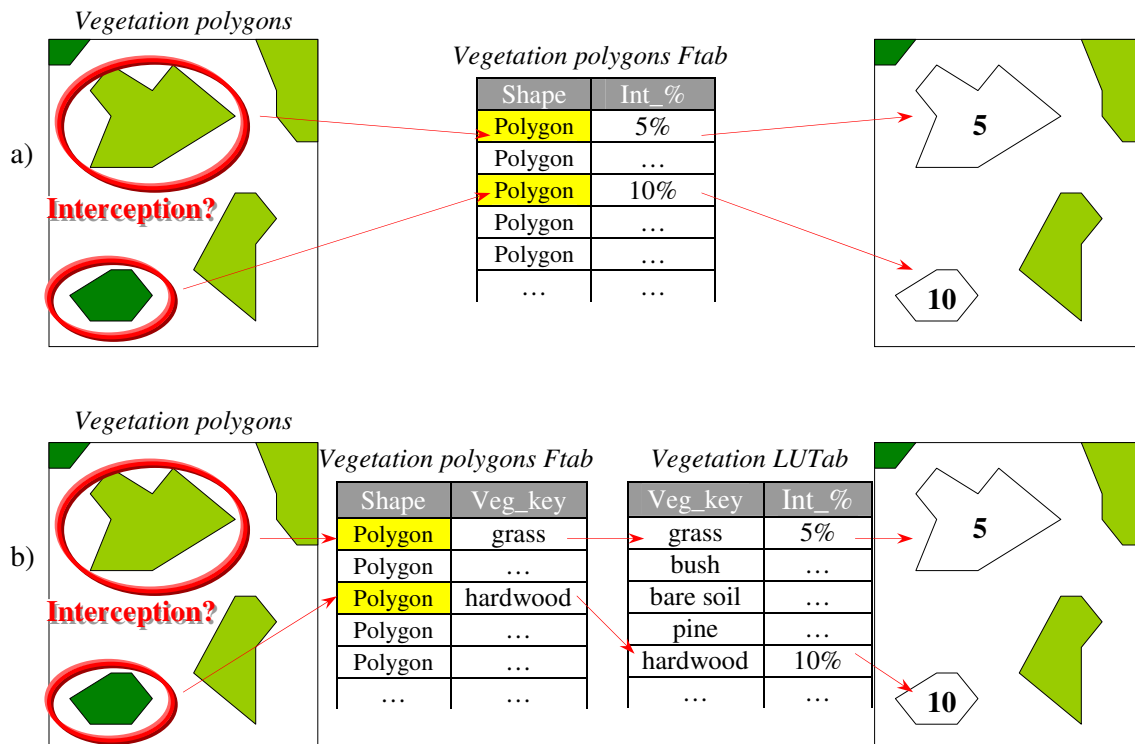


Figure 12: polygon attribute request processes: a) directly through the associated *Ftab*, b) by the means of a lookup table

III.1.4) Data management

[III.1.4.1) Data display and identification

The ArcView window where geographical information is displayed is called a *View* and every displayed layer of information is called a *theme*. The visual representation of raster data on a view is called a *Gtheme*, for grid theme. The shapes (points, lines or polygons) of an *Ftab* are visually represented on a view as one *Ftheme*, for feature theme. Georeferenced images are displayed as *Ithemes*, for image themes.

III.1.4.2) User interaction on data

Whereas other GIS softwares like ArcInfo are essentially command-based, which implies that the user has to type command line for every single operation on geographical data, ArcView is provided with a graphic interface with menus and buttons that activate special tasks on specific objects.

Most of the theme operations (display, basic requests on attributes, layout) are executable from clickable buttons. Behind each button lies a script, which is sort of a small program that runs and processes the selected theme(s). The scripts are coded in a proprietary object-oriented programming language developed by ESRI called *Avenue*.

III.1.4.3) Calculation with grids

Computations with grids of same *cell size* (here, a cell is a square of 10 by 10 square metres) are very easy since one can add, multiply grids as one would handle numbers. The computation occurs on a cell-by-cell basis over the overlapping area of the two grids, therefore, if one wants to multiply grid A cells by grid B cells, and divide the whole result by 3 and then take the square root of it, the following Avenue script is enough (cf. **figure 13**):

```
resultGrid = (gridA*gridB/(3.AsGrid)).Sqrt
```

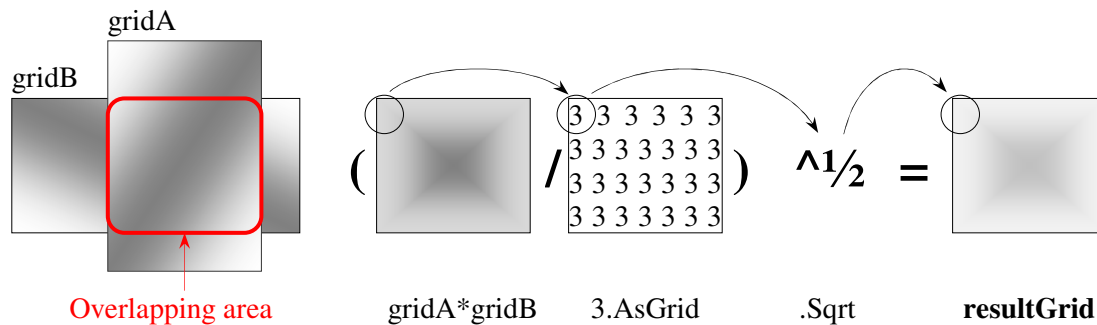


Figure 13: example of cell-by-cell based grid computation

The “.AsGrid” postfix after “3” converts the number 3 into a grid of which all cells store the number 3. The “.Sqrt” postfix at the end takes the square root of each cell of the grid which is just before. As brackets were used before the square root command, all the computations within the brackets are done first and then the square root command is applied (see figure 13blablabla).

III.1.4.4) Conversion between shapes and grids

The expected hydrological model needs both the geographical flexibility of shapefiles (which are very easy to edit) and the computational flexibility of grids (which allow easy but powerful calculations). To accommodate that, conversion from shapefile to grid was necessary. Conversion between grids and shapes is easy both ways, but whereas a shape can have several attributes, each cell of a grid can store only one value. Therefore, only one numbered field/attribute can be selected when converting a shapefile to a grid.

For example, the following Avenue line allows to convert vegetation polygons into a grid, which will contain vegetation interception values (cf. **figure 12**, process a):

```
interceptionGrid = vegetationFTab.AsGrid(viewProjection, Int_%, {cellSize,gridExtent})
```

where *viewProjection* contains the projection system used to display geographical features on the virtual map contained in the View window, *Int_%* is the field containing the interception value for each polygon, and in curly brackets are information for the cell size and the whole extent of the resulting interceptionGrid.

However, as I have chosen to use keys and lookup tables to assign values to polygons, the conversion process is more complex. A script browses the lookup table to find the values corresponding to the keys of each polygon and writes them in a new field added to the Ftab. This new field is used for the shape to grid conversion (cf. **figure 14**).

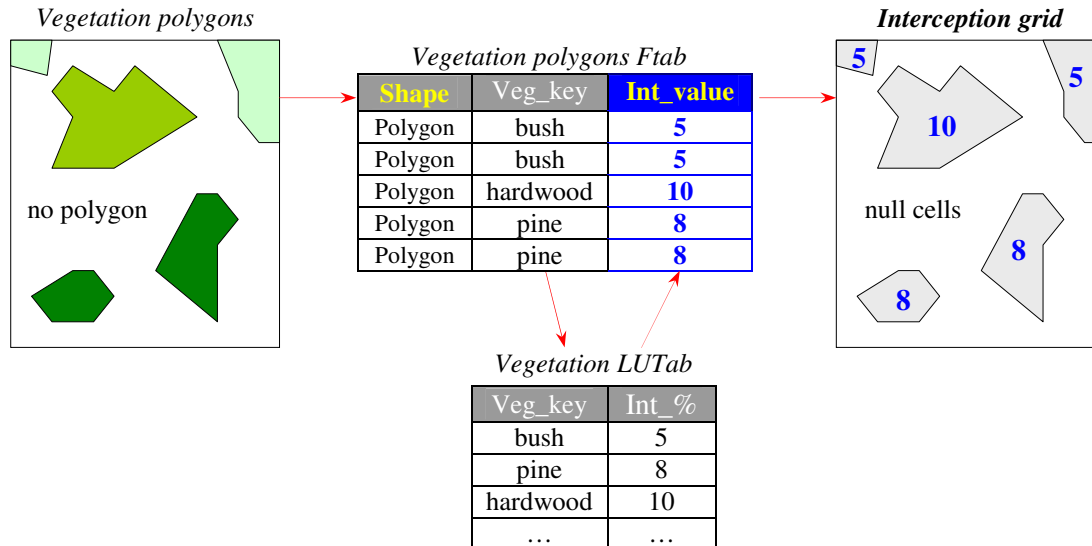


Figure 14: shape to grid conversion with values stored in a lookup table

As on the one hand, Avenue is already designed to handle themes or other objects in a geographical context, and on the other hand, chosen modeling concepts need no huge computational resource, I chose to implement and code the hydrological model in Avenue, so it can be and run on every computer platform where ArcView is installed.

III.2) Implementation of the production function

The purpose of the production function is to derive distributed runoff from a gross precipitation height value using land coverage and soil properties. Here, I want to show how the conceptual model described in part II is translated into a procedural model by the means of Avenue programming language and the advantages due to a GIS environment.

III.2.1) Land use files

Original land use data files, like urban features and waterbodies, were provided by John Wiley, the GIS supervisor of Chilliwack City. However, I had to map or derive rural land use by myself through the means of a set of georeferenced orthophotos, which have a pixel resolution of 1 metre (one pixel represents a 1 by 1 metre square on the ground). Here are the description of the land use files, how they were processed and how they are used in the model.

[III.2.1.1) Waterbodies

Original waterbody data was contained in a polyline shapefile, in which stream network data is not continuous due to diversity of data sources such as aerial, cadastral and observed data from maintenance workers. Therefore, I had to reconnect stream reaches together using the thalwegs and the natural drainage network derived from topographical data contained in the DEM (with the ArcView *FlowAccumulation* request explained in III.3.1.5).

The stream network file used in UHM (Unit Hydrograph Model) is already the modified polyline converted as a grid. Paradoxically, stream cells are considered 100% impervious because all the amount of rain that falls over them goes as runoff. Therefore, the number stored in each stream cell is 1, for 100% impervious.

III.2.1.2) Vegetation features

All vegetation features are considered totally pervious, therefore all vegetation classes accounts mainly for different behaviours in terms of rainwater interception, but also for different nature of vegetation. Each vegetation class has its own *vegetation key*, which is associated with an interception value. These vegetation keys and interception values are stored in the *vegetation LUTab* (cf. **appendix 3**) where they can be easily accessed (cf. **figure 14**) and modified. Vegetation features are classified into 3 polygon shapefiles: natural vegetation, artificial vegetation and bare grounds.

Bare grounds and natural vegetation polygons were drawn over the georeferenced orthophotos. As such, bare grounds are pervious areas over which interception does not occur. They occupy only a very small surface of the catchment and their effect on the whole watershed must be currently insignificant, but as land use is supposed to change a lot, this class is likely to grow, at least temporarily across new building sites.

Here, “artificial vegetation” refers to crops and fields. Instead of drawing the fields contour, I used the urban cadastral polygon shapefile and kept only the cadastral polygons that corresponded with field outlines. I have created just one crop class, called “mean crop” with an interception value of 6% and used it for every crop polygon because crop are subject to rotation every year and interception varies a lot as plants grow. However, the user can easily account for the “growing effect” by modifying the mean-crop-class interception value.

The natural vegetation of the watershed is a mix of hardwood and coniferous trees, which have more or less the same interception value of 9% (K.N. Brooks, 1991), which was associated in the *vegetation LUTab* to the “mean forest” class.

Then, I considered that any area, which is neither natural, artificial vegetation, bare soil nor urban feature, is grass. This reasonable assumption implies that neither the user nor me have to draw all the grass polygons, which saves a lot of time and energy. The “mean grass” class was attributed the interception value of 3%. As explained in **II.3.4**, pervious parts of semi-impervious areas are considered as covered by mean grass, therefore a 3% interception is deducted from gross precipitation there too for excess runoff generation. It is assumed that there is no interception over totally impervious areas, or impervious parts of semi impervious areas. Therefore an extra class for impervious areas was added to the *vegetation LUTab* with an interception value of 0%.

III.2.1.3) Urban features

Urban features participate in 2 different processes: they first influence vegetation coverage and interception, and being impervious, they prevent rainwater from infiltrating the soil. As vegetation polygons refer to a lookup table for interception values, so urban shapes are linked to an urban lookup table for imperviousness percentage (cf. **appendix 3**). The link between shapes and the *urban LUTab* is a key stored in the “Urban_key” field, which is present in both the shapes Ftab and the urban LUTab (cf. **figure 15**).

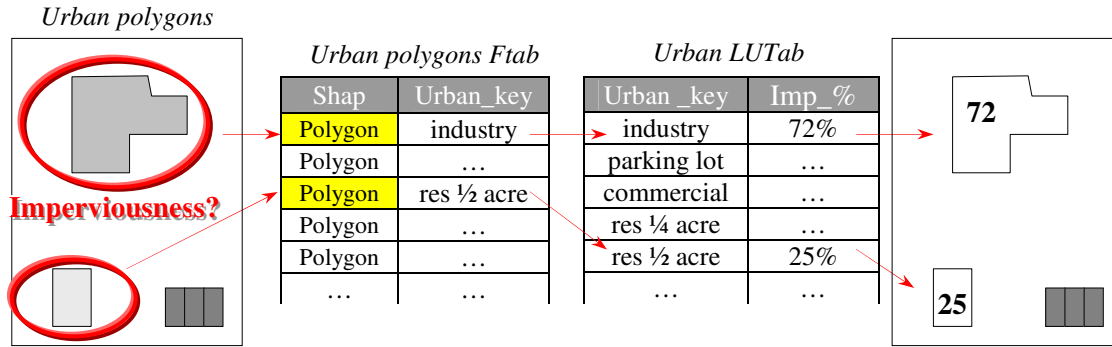


Figure 15: articulation between urban shapes and the urban lookup table

The main class in the *urban LUTab* is the “impervious” class, which has an imperviousness value of 100%. All the other classes account for semi-impervious features. I sorted urban features in 2 files.

All types of road are represented in a polyline shapefile, which was provided by Chilliwack City. Paved roads are considered totally impervious. I have created a class for gravel or dirt road, which has an imperviousness value of 60% due to soil compaction by vehicles and which can be used for any field road, though they are not currently shown to the road polyline shapefile.

Urban areas like houses, farms, parking lots, industrial plants are listed in a polygon shapefile. All the residential lot outlines were derived from Chilliwack City’s cadastral data shapefile, but additional farms were drawn over georeferenced orthophotos used before. As cadastral polygons do not delineate buildings but properties, such areas are semi impervious. Instead of computing impervious area percentage for each polygon, which requires a lot of hi-resolution mapping, I used numbers resulting from the statistical approach used in the SCS Curve Number approach (McCuen 1982) for runoff generation (cf. **appendix 4**).

Its rather empirical formula, which derives from statistical analyses of rainfall-runoff data on small catchments, uses the so-called Curve Number parameter, which reflects imperviousness among other things. As North American cities are newer due to a shorter historical background compared with European cities, their design follows patterns of residential lots where houses are size-standardised and organised in blocks. Relationships between residential lot extent classes and imperviousness (cf. **curve number table in appendix 4**) are therefore relevant and quite accurate, so I used them to attribute imperviousness value to residential polygons. Once again, working in a GIS environment allowed me to easily make request on polygon extents in order to assign them the proper residential class and to control the result visually with the georeferenced orthophoto background.

III.2.1.4) Soil features

Soil information for this project was extracted from the Computer Assisted Aid Mapping Project (CAPAMP, the first Canadian governmental Geographic Information System) and files were provided and converted back to ArcView format by Chilliwack City. It has one polygon file for soil outlines and different databases for soil and horizon physical and chemical characteristics. Soils have been mapped and classified between the late 1960’s and mid 1970’s according to soil order, type and horizon series. CAPAMP original tables were too detailed for the needs of the model, and some of the information, especially chemical properties, had turned obsolete. The next paragraphs describe the structure of the soil data finally used in the model: the soil polygon shapefile, the soil information table and the horizon information table.

The soil polygon shapefile and its associated Ftab contain soil outlines and the polygon soil key, which links each polygon with the soil information table. Interception Ditch catchment comprises 65 soil polygons and 16 different soil types.

The soil information table is a Vtab, which contains general information for each type of soil that can be found in Interception Ditch watershed. Whereas the soil polygon Ftab contains several occurrences of the same soil key since different polygons share the same soil properties, the soil information table has only one occurrence of each soil key. Therefore this table has only 16 records (or rows).

The other fields of the table give information about the water table mean depth, the water table depth standard deviation (which was apparently set to half of the depth) and the soil moisture class. Water table depth information is only available for soils in the valley (cf. **figure 16**). CAPAMP moisture classes are expressed with letters, which reflect the degree and the duration of soil saturation, by the means of the Climatic Moisture Index (CMI). Unfortunately, the CMI uses parameters I could not access, so I was not able to directly assign soil initial moisture values to those classes.

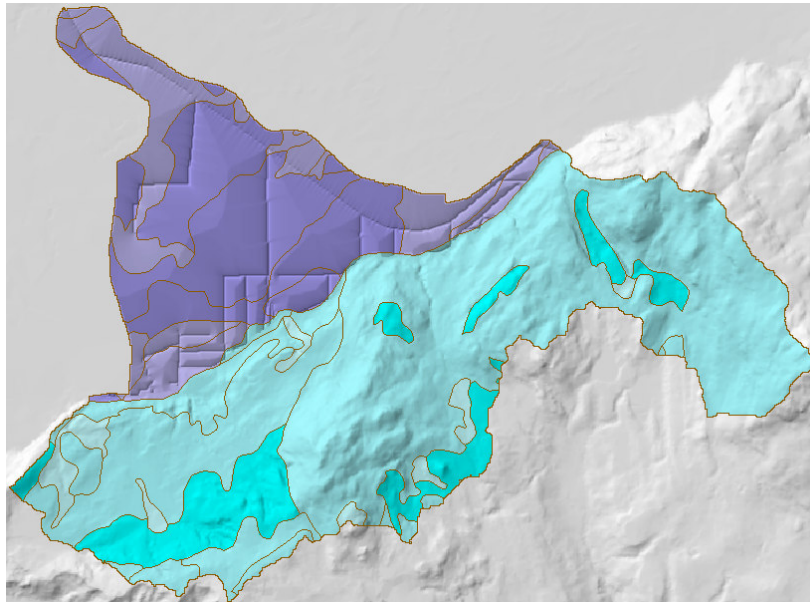


Figure 16: soil polygons where the water table can be found (dark blue) across Interception Ditch

The horizon information table contains information for each horizon of a soil type. A particular horizon is defined by 2 attributes (practically, by 2 fields): the soil key (used in the soil polygon Ftab and the soil information table) and the depth code (horizons are numbered with consecutive integers starting at 1 for the surface horizon and increasing with deeper horizons). The horizon information table has a total of 71 horizons shared among the 16 different soil types, which makes an average of 4 or 5 horizons per soil type.

Other horizon attributes are horizon thickness (in centimetres) and horizon texture. Textural groups used in the Canadian System of Soil Classification (CSSC) were defined with silt, clay and sand percentages (cf. **figure 17**). Silt, clay and sand particles have very different sizes, therefore their relative amount influences the soil porosity and the water storage capacity. In the same way, whereas silt and sand are spherical particles, clay is structured in layers through which water hardly flows, therefore texture composition is likely to influence soil infiltration rate too.

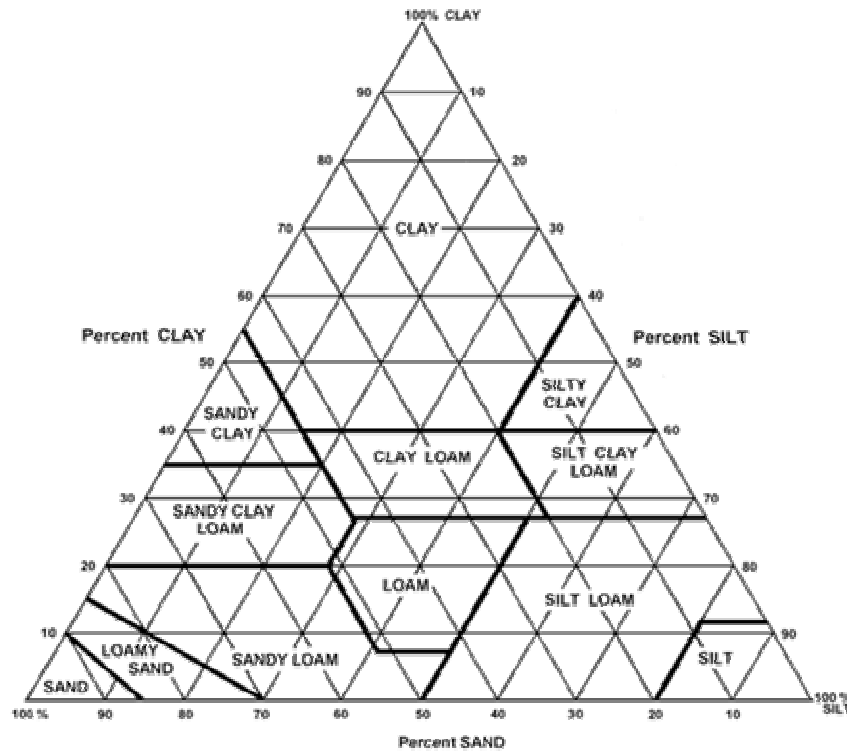


Figure 17: the Canadian System of Soil Classification textural triangle and texture class definition

Some organic horizons were not originally assigned any texture class in the CAPAMP horizon database because their texture could not be expressed in terms of silt, clay and sand relative composition. Therefore, 3 new classes were created: one for peat horizon, another one for humic clay horizon, and one for LFH (litter, fibre and humus) horizon. Fifteen texture keys are represented across Interception Ditch watershed (cf. figure 18):

Texture name	Texture key code
Sand	S
Loamy sand	LS
Fine sandy loam	FSL
Sandy loam	SL
Loam	L
Silty loam	SIL
Sandy clay loam	SCL
Clay loam	CL
Silty clay loam	SICL
Sandy clay	SC
Silty clay	SIC
Clay	C
Peat	P
Humic clay	HC
Litter fiber humus	LFH

Figure 18: horizon texture classes represented across Interception Ditch watershed

Now that we have seen how different data types can be structured and linked together in a GIS environment, runoff generation concepts described in part II must be structured and expressed in Avenue language. Furthermore, we still need to derive values for soil water storage capacity and soil infiltration rate.

III.2.2) Implementing runoff generation

The algorithm for the production function first has to generate grids for imperviousness, interception, soil water storage capacity and infiltration rate. Then, it must handle them to produce the net precipitation grid and the runoff generation grid, which will be used by the transfer production. I have used a modular approach as much as I could, assigning each task to a different script, to allow better understanding of the model structure and further upgrading flexibility.

III.2.2.1) Distributed imperviousness grid

The script, or module, which outputs the imperviousness grid, uses the processes described in III.1.4.4 (cf. figure 14) with the urban cadastral polygon shapefile to produce a first grid. Then, a second grid is produced from the conversion of the road polyline file. When combining the two grids into a single one, in case of overlapping non-zero values from the two grids, road imperviousness values yield to urban polygon values (see figure 19).

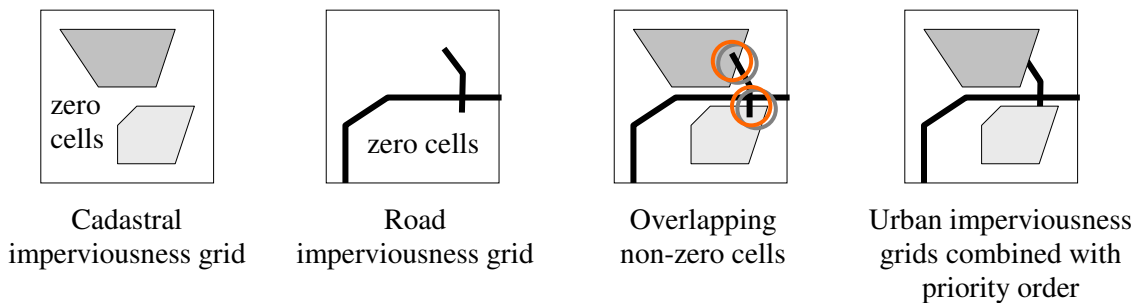


Figure 19: priority order for imperviousness grid combination

The stream network, over which all rainfall turns to runoff, is then superposed over the urban imperviousness grid as completely impervious. Streams are therefore considered prior to urban features because sometimes they cross properties. That constitutes the final imperviousness grid (cf. figure 20).

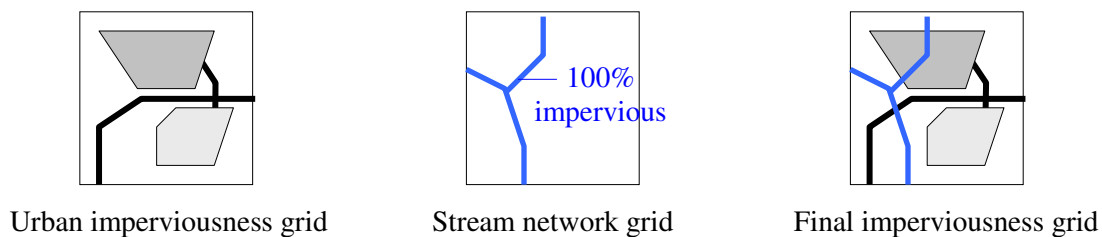


Figure 20: final imperviousness grid

III.2.2.2) Distributed interception grid

Though it is not exactly what happens in the final algorithm (temporary grids are actually created and used at different steps of the algorithm), the priority order between all the themes involved in the interception grid is as follows:

i interception grid cells where imperviousness grid cells equal 100% are set to 0, which transcribes that there is no vegetation interception over totally impervious surfaces like paved roads. All interception grid cells over semi-impervious areas have their value set to default mean grass value, but the excess runoff volume is corrected by the imperviousness value, and then the runoff volume itself is corrected by extra runoff generated by imperviousness (cf. II.3.4).

ii interception grid cells located in bare grounds polygons are set to 0.

iii crop polygons are converted to a grid with interception values according to the process explained in III.1.4.4 and illustrated on figure 14.

iv vegetation polygons are finally converted and added where cells still have no value yet.

[III.2.2.3) Soil water storage capacity distributed grid

From the available soil data, it has been decided that soil water storage capacity and infiltration rate values would be derived from soil texture. From water storage capacity and infiltration rate values found in different bibliographic sources, a *soil texture lookup table* was created in order to assign each texture class with water storage capacity and infiltration rate values (see appendix 3). Water storage values are expressed in percents (which can be understood as a percentage of void in a unit volume of soil, or as a number of precipitation centimetres, which can be stored in one meter of soil) for dry soils. Therefore, they reflect water storage **potential** (WSPot), of which soil initial moisture must be deducted to give the real water storage **capacity** (WSCap).

When the script that computes the water storage capacity is launched, it proceeds through the following steps for each soil polygon:

i it browses through the horizon table, multiplies the water storage percentage (value found in the *soil texture LUTab*) by the horizon thickness for each horizon of the proper soil to produce the horizon WSPot.

ii horizon WSPots are added together as long as there are horizons of the same soil, or as long as the accumulated depth is shorter than the local water table depth. Before a simulation, the user can choose between 5 water table depth modes: mean depth, low depth (mean depth+standard deviation), high depth (mean depth-standard deviation), saturation (water table has reached surface) and custom depth (where the user can set a global depth in centimetres). This applies only for valley soils where the water table exists. However, as horizon information were accessed by augering, the last horizon thickness value of a soil's series could not be set because the horizon was not augered through. Thickness value for valley soils' last horizon was set so the accumulated thickness could reach the lowest water table level (mean depth+standard deviation). For mountainous soils, last horizon thickness was arbitrarily set so accumulated thickness equals 50 centimetres.

iii the accumulated WSPot value represents the soil's WSPot, and soil initial moisture has to be considered now in order to get the WSCap with the following formulae:

$$\text{WSCap} = \text{WSPot} * (100 - \text{initialMoisture\%}) / 100$$

The user can choose between 4 modes to account for soil initial moisture: 0% (all soils are completely dry, WSCap=WSPot), 100% (all soils are completely saturated,

WSCap=0), user percentage and *soil moisture LUTab* mode. *Soil moisture LUTab* mode uses the soil moisture classes described in III.2.1.4 and each class can be assigned a percentage value (cf. **appendix 3**).

Once that each soil polygon has been assigned with its WSC value, the polygons are converted to a grid.

[III.2.2.4) Soil infiltration rate distributed grid

Though infiltration rates are likely to change as soils get wetter, the infiltration rate values written in the *soil texture LUTab* correspond to saturated hydraulic conductivities. As far as vertical infiltration is concerned, an horizon series can be understood as a vertical set of reservoirs connected in series, each reservoir dripping into the lower one with its own speed. But such an approach would require too many calculations and too many chained boolean tests, which cannot be implemented easily with grids, since ArcView grids do not support the typical “if...then...else” statement.

Instead of that, the script browses through the horizon with the same stop conditions (until the end of the horizon series or the water table), lists the infiltration rates, and one is picked up to represent the whole soil. Before the simulation begins, the user can choose between 3 infiltration rate modes: highest (water infiltrates at the highest/fastest infiltration rate), lowest and median. Once that each soil polygon has been assigned with its infiltration rate value, the polygons are converted to a grid.

[III.2.2.5) Distributed runoff grid

Here are the input data used the script extract, which computes the runoff generation:

rawRainHeightGrid is the gross precipitation grid for the time step,
impGrid is the imperviousness grid,
vegIntGrid is the vegetation interception grid,
wscGrid is the water storage capacity grid,
infGrid is the infiltration rate grid,
timeStepDuration is the time step duration expressed in minutes.

Commentaries in Avenue scripts can be added after the ' symbol. The only way to use boolean tests with grids is with the postfix command “.Con(yesGrid,noGrid)” (cf. **appendix 5**). So, for one time step:

```
' Computes the runoff pseudo height due to impervious area  
impRunoffGrid = rawRainHeightGrid*impGrid/100.AsGrid
```

```
' Computes net precipitation rain intensity grid  
rainHeightGrid = rawRainHeightGrid*(100.AsGrid - vegIntGrid)/100.AsGrid
```

```
' Computes rain intensity grid  
rainIntGrid = rainHeightGrid/(timeStepDuration.AsGrid)
```

```
' Creates different boolean grids for tests coming next  
satSoils = (wscGrid = 0.AsGrid)  
unSatSoils = (wscGrid > 0.AsGrid)  
higherRainInt = (infGrid <= rainIntGrid)  
higherInfRate = (infGrid > rainIntGrid)
```

```
' Total runoff generation over saturated soils
```

```
runoffGrid = satSoils.Con(rainHeightGrid,runoffGrid)

' Total infiltration over unsaturated soils until the soil gets saturated
runoffGrid = ((unSatSoils) and (higherInfRate)).Con
((rainHeightGrid>wscGrid).Con((rainHeightGrid-wscGrid), 0.AsGrid), runoffGrid)

' Partial infiltration and runoff generation over unsaturated soils which have a lower
infiltration rate
runoffGrid = ((unSatSoils) and (higherRainInt)).Con
( ((rainIntGrid-infGrid)*timeStepDuration.AsGrid +
((infGrid*timeStepDuration > wscGrid).Con
((infGrid*timeStepDuration-wscGrid),0.AsGrid))), runoffGrid )

' Corrects excess runoff with pervious area percentage and adds runoff due to imperviousness
runoffGrid = runoffGrid*(100.AsGrid-impGrid)/100.AsGrid + impRunoffGrid
```

Of course, the infiltrated water grid is computed too, in order to modify the water storage capacity grid for the next time step. Runoff grids produced by the production function are the input for the transfer production, which derives the hydrograph from them.

III.3) Implementation of the transfer function

Whereas the implementation of the production only required only basic GIS operations such as relational database requests and land use file management in two dimensions, the implementation of the transfer function requires a 3-D approach to derive runoff flow direction and flow speed from a gridded Digital Elevation Model. Furthermore, as mentioned before, grid objects were not designed in ArcView in a way you can access information carried by a particular cell. Therefore, most of the DEM processing requests are already implemented in ArcView, but some of them gave unexpected or inadequate results so I had to recode some and develop others.

III.3.1) DEM-derived hydrological concepts

A DEM is a digitised representation of the topography of an area, projected on the horizontal plan. In this section I want to present a few concepts related to DEM, which I used to implement the transfer function.

[III.3.1.1) Cardinal and orthogonal cells

Most of the requests doable on DEM need to compare elevation change between a processing cell and its 8 neighbouring cells with the distance between cell centres. There are 2 types of neighbouring cells: the 4 cardinal cells (connected by the sides to the processing cell and located north, east, south and west of it) the 4 diagonal cells (connected by the corners and located north-east, south-east, south-west and north-west). Distance between cell centres is not always the same: in the cell size units, distance between the processing cell centre and a cardinal cell centre is 1 whereas it is $2^{1/2}$ ($\approx 1.414214\dots$) with a diagonal cell (cf. **figure 21**).

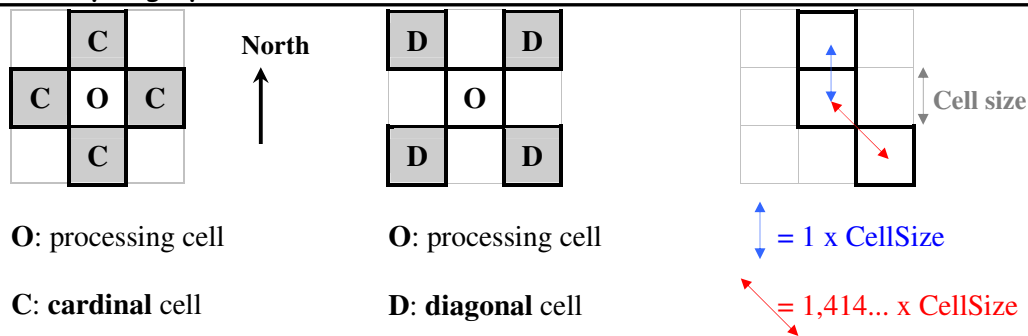


Figure 21: cardinal and diagonal cells

III.3.1.2) Flow direction

The DEM can be used to estimate flow direction at the scale of a cell. The flow direction request available in ArcView relies on the following assumption: water flows from a cell to one and only one of its neighbouring cells according to the steepest descent. The request browses all the cells of the elevation grid (DEM) and the slope between the processing cell and the neighbouring cells contained in the processing kernel is calculated as the elevation raise/drop divided by the distance between cell centres. Depending on the destination cell, the value attributed to the processing cell varies (cf. **figure 22**).

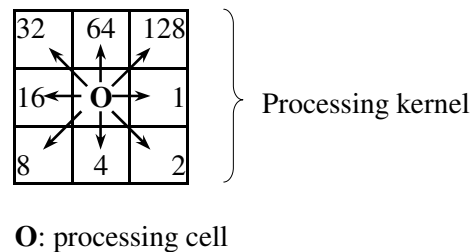


Figure 22: flow direction values attributed to the processing cell

For example, if the steepest descent brings the water down to the south-east cell, then the flow direction is coded 2 (cf. **figure 23**). Flow direction is actually a simple and discretised representation of local (at the cell scale) slope aspect.

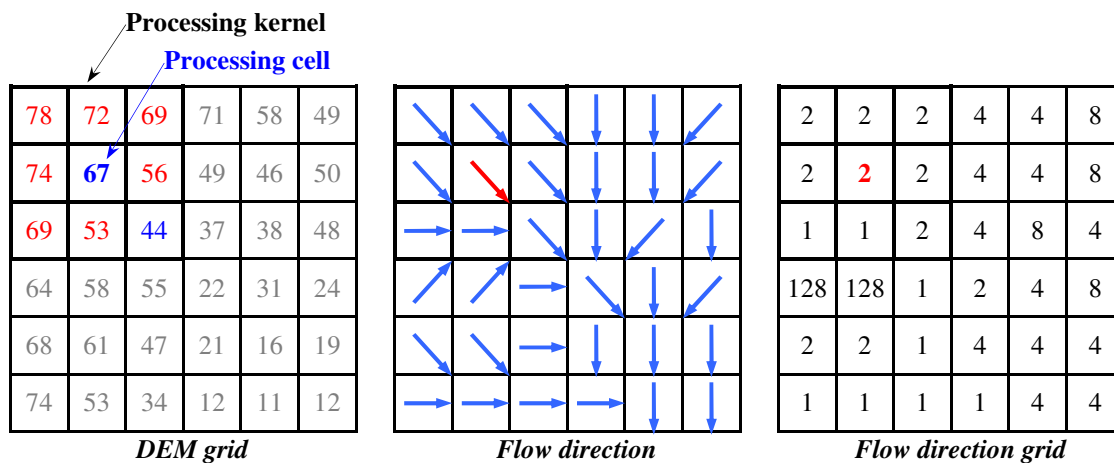


Figure 23: flow direction derivation from an elevation grid

The ArcView request to create a flow direction grid from an elevation grid is:

flowDirectionGrid = elevationGrid.**FlowDirection**(forceEdge)

(if forceEdge is TRUE, then all cells on the edge of the surface will flow away from the surface. If forceEdge is FALSE, then flow is calculated normally for edge cells with the edge being slightly lower than the cell)

There are many interesting requests, which may be done on a flow direction grid computed from a large elevation grid. I used the ones explained in the following paragraphs.

III.3.1.3) Sinks

A sink in an elevation grid is a cell, or a set of spatially connected cells, which is lower than all the surrounding cells. Cells of a sink receive undefined flow direction value. In most cases, sinks are artefacts produced during the sampling of elevation data. Naturally-occurring sinks in elevation data with a cell size of 10 metres or larger are rare (Mark, 1988) except for glacial or karst areas (which is not the case here), and are generally considered as digitising artefacts due to data sampling.

Moreover, real depressions are very likely to be naturally filled by sediments or water, therefore, to create an accurate representation of flow direction, it is best to use a DEM that is free of sinks. An elevation grid, which has been processed in order to remove all its sinks, is referred to as a sinkless (or depressionless) DEM.

The identification and removal of sinks, when trying to create a sinkless DEM, is an iterative process. When a sink is filled, the boundaries of the filled area may create new sinks, which then need to be filled.

III.3.1.4) Contributing area, watershed delineation

Given a flow direction grid and a specific cell and a specific cell chosen as an outlet, the flow direction grid can be used backwards to delineate the contributing area above the outlet. It is an iterative process, which first looks for the cells directly flowing into the outlet cell, then looks for the cells flowing into the cells that flow in the outlet cell, and so on (cf. **figure 24**).

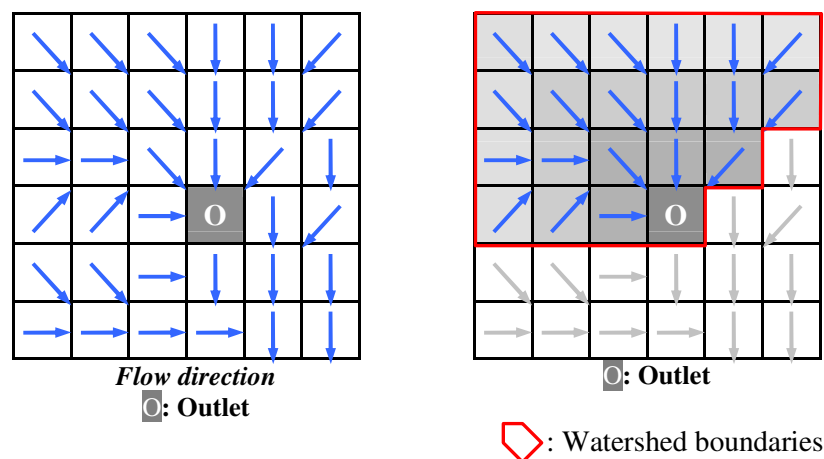


Figure 24: delineating a watershed with a flow direction grid

The ArcView request to create a flow direction grid from an elevation grid is:

watershedGrid = flowDirectionGrid.**Watershed**(outletGrid)

Elevation and flow direction grids can be clipped to the boundaries of the watershedGrid, first of all in order to reduce the size of the data set, but also in order to apply other requests on a specific area only.

III.3.1.5) Flow accumulation

This request browses through all the cells of the flow direction grid, and the accumulated flow is based upon the number of cells flowing into the processing cell. The current processing cell is not considered in its accumulation, but is added to the accumulation of the cell in which it flows.

The results of flow accumulation can be used to create a stream network by applying a threshold value to subset cells with a high accumulated flow. Moreover, internal cells, which have an accumulated flow value of 0, are very likely to be located on local ridges. They can be used to draw drainage divides that split the watershed into subcatchments (cf. **figure 25**).

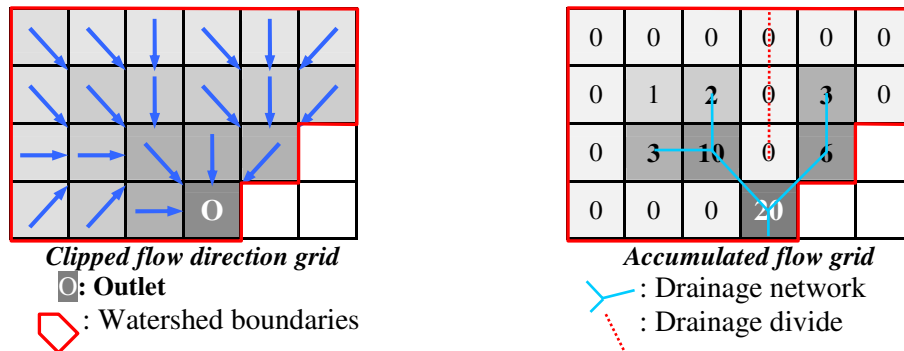


Figure 25: flow accumulation, drainage network and divide derivation

The ArcView request to create a flow direction grid from a flow direction grid is:

flowAccumulationGrid = flowDirectionGrid.FlowAccumulation(weightGrid)

(weightGrid is optional. It can be a nil object or a grid, which contains numbers that will be added during the accumulation calculations)

III.3.1.6) Flow length

The flow length request is can be used either for downstream or upstream flow length (cf. **figure 26**). For each cell of the input flow direction grid, it gives in cell units:

- distance downstream to a sink or the outlet,
- distance to the most upstream divide. The value carried by the outlet cell is the longest flow path of the watershed.

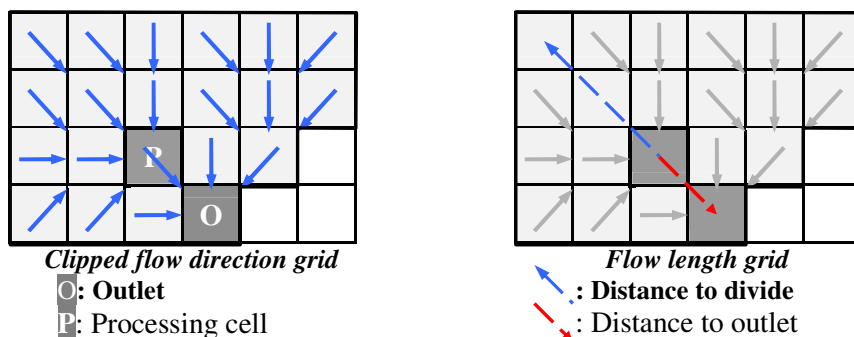


Figure 26: downstream and upstream flow lengths

In the same way as with the flow accumulation, the flow length request supports an optional weight grid, which can be used to convert downstream flow length to downstream flow time. To do so, the weight grid must contain impedance to downstream movement, which is the time required to move from the processing cell to the downstream cell defined by the flow direction grid:

```
flowLengthDivideGrid = flowDirectionGrid.FlowLength(nil,upStream)
    with upStream = true
flowLengthOutletGrid = flowDirectionGrid.FlowLength(nil,upStream)
    with upStream = false
flowTimeOutletGrid = flowDirectionGrid.FlowLength(impedanceGrid,upStream)
    with upStream = false
```

This request can be used to produced a **flowTimeOutletGrid**, which will be classified (i.e. sliced into zones defined by value intervals) in order to draw the time zone areas required by the unit hydrograph theory.

III.3.1.7) Slope

The ArcView built-in slope request fits a regression plane to the 9 elevation values of a 3x3 processing kernel around the processing cell, which is assigned the regression plane slope value. Though powerful because it does not need a flow direction grid, this request tends to smooth slopes of the elevation grid, which is already a flattened representation of the reality due to the elevation sampling.

Moreover, as the flow direction request looks for the steepest descent, I coded a script, which requires an *elevationGrid* and a *flowDirectionGrid* (the latter being automatically generated if missing) and computes the slope in the *flowDirectionGrid* direction. Although the concepts are very simple since one only needs an elevation drop and a distance to compute a slope, the coding of this basic request was made harder because ArcView does not permit single cell value access. However, the difference between the *regression plane method* and the *flow direction method* becomes important at slope inflexion angles where the regression plane method smooths the breaks in the slope (cf. **figure 27**). Over the mountainous area, the maximum difference between slope values generated by the two methods for the same cell can reach 36° , with a mean difference of $2,4^\circ$.

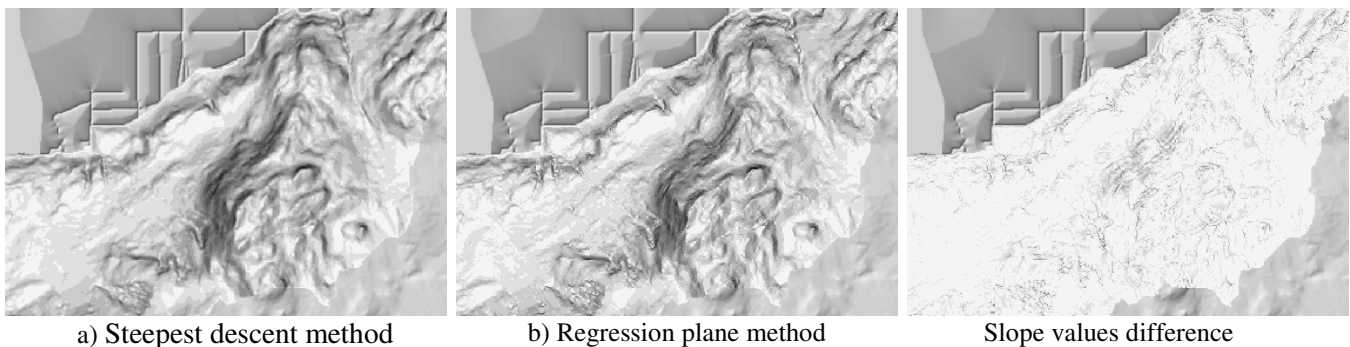


Figure 27: slope derivation on the mountainous part of the watershed with 2 methods (a: steepest descent, b: regression plane) and comparison. Dark pixels represent high slope values

III.3.2) Runoff routing direction

Though the runoff routing direction can be easily and fairly well approximated with a flow direction grid derived from an elevation grid, there are a couple of precautions to take in order to

use a relevant data set. Some of them are related to the DEM itself, but further modifications had to be made to account for disturbing processes due to human activities.

III.3.2.1) DEM data set pre-processing

DEM data set origin

The original DEM data set provided by Chilliwack City was a Triangular Irregular Network wholly covering the area of Chilliwack district. Due to the diversity of materials used by the municipality to create the TIN (see **appendix 6**), its precision is 5 metres for horizontal control and from 1 decimetre (in the valley) to 2 metres (in the mountainous area) for vertical control. As a TIN is a set of specific points, such as peaks, ridges, and breaks in slope, the resulting network is irregularly distributed in space. It was gridded in a 10 by 10 metre square grid for computational efficiency and easy implementation of hydrological concepts.

Sinks removal

The elevation grid was clipped to a 5 510 by 7 970 metre rectangle, which contains Interception Ditch watershed. 5 loops were necessary to fill 2963 cells (29,63 ha) across this area. As we can see on **figure 28**, most of the sinks are located in the valley, which is flat. Moreover, whereas sinks in the valley are rather spread out, sinks in the mountains are more sporadic and made of a couple of cells only, if not just one (except for one location in the semi-urbanised watershed).

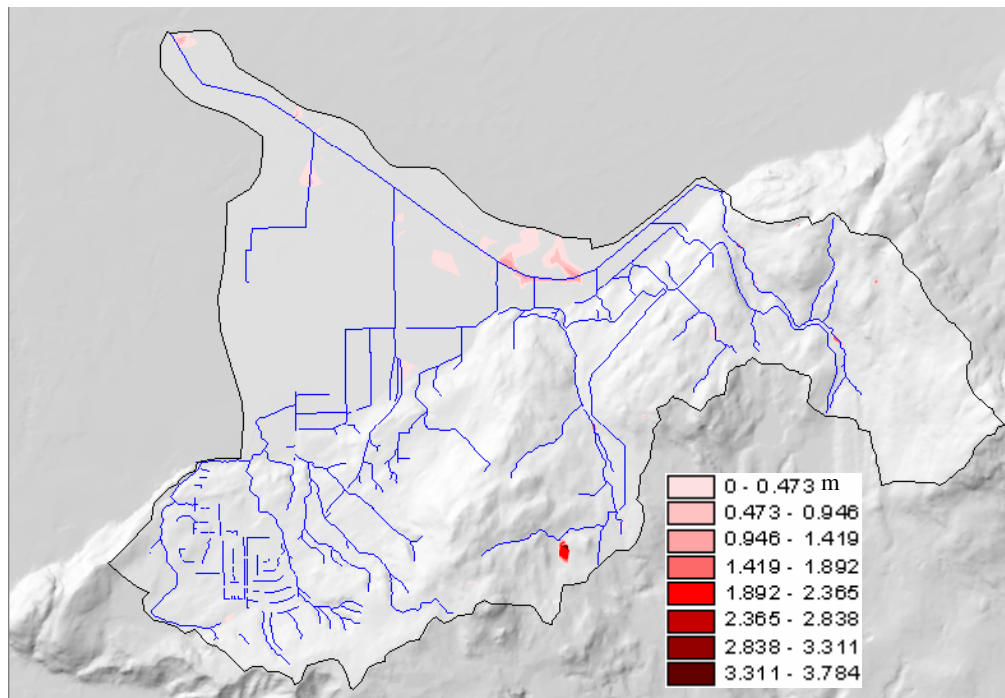


Figure 28: location and depth of the sinks filled across Interception Ditch

The huge amount of sinks in the valley, and the fact that their pattern does not follow any stream direction shows that the gridding of the DEM must have smoothed elevation data too much for a good representation of the valley topography at the hydrological level. Indeed, the elevation change across the valley area is less than 10 meters for an area of 5,4 km².

III.3.2.2) Flow direction grid processing

Hydrological features derived from a DEM reflect typical behaviours of water in a natural environment. However, unusual natural conditions or human activities can heavily disturb basic

hydrological processes, such as flow routing, and make it very challenging for modeling. In this part, I want to show those problems, how they can be solved from a conceptual point of view, but also on the top of that, from a technical point of view with algorithm concepts possible with a GIS software like ArcView 3.2.

Flow direction across the valley

ArcView's FlowDirection built-in function is a very efficient method for deriving flow routing from a DEM, as long the DEM is meaningful. Different facts lead me to realise it was no longer the case in Chilliwack valley area:

- the large number of filled sinks mentioned above,
- elevation fluctuations are much thinner here than over the mountains and ditch width is often 5 times smaller than the cell size of the DEM, so the stream network can no longer be extracted from the DEM because it got blurred when elevation data was being gridded and.
- the natural drainage network derived from the elevation grid and its flow direction grid (cf. III.3.1.5) does not make any sense any sense in the valley (cf. **figure 29**) with the actual network. As this procedure is based on flow accumulation, it is very likely that overland flow routing is meaningless too.

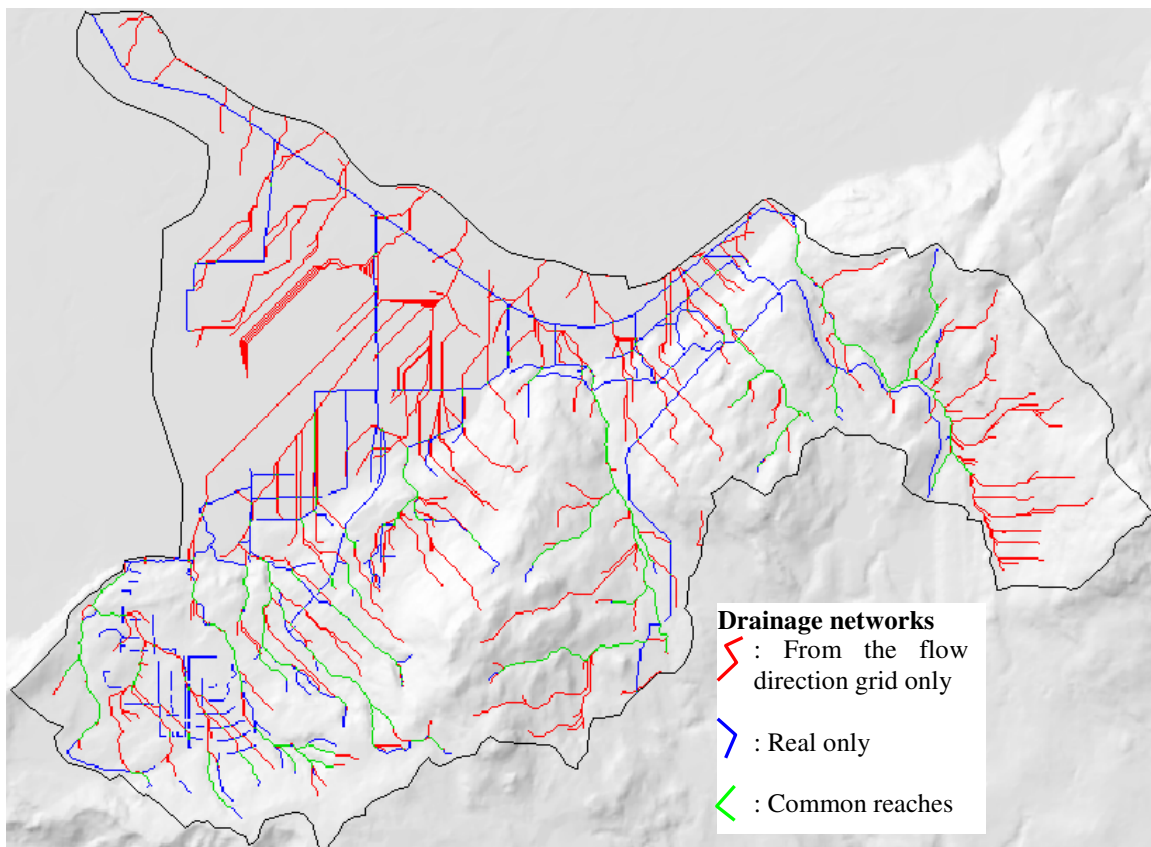


Figure 29: derived natural drainage network and real ditch network

Therefore, I have visually separated the DEM data set in two zones on a visual analysis of the flow direction grid at the foot of the mountainous area. I kept the upstream area data where flow

direction patterns were consistent with the real stream network. Eventually, all the valley area had to be deleted, but as I still wanted the model to be used over this area to model the discharge at the very outlet, the following issue rose: **how can we estimate flow direction where we have no relevant topographical information?**

- Artificial flow direction for the interfluvial area

To accommodate that, I wrote a script, which computes a virtual DEM based on the distance to the closest draining ditch, assuming that, on a flat land, water would rather flow towards the closest ditch (cf. **figure 30**). As each cell is assigned the distance value to the closest ditch, the further a cell is, the higher it gets, but all ditch network cells are therefore given the value 0 because they already are in the ditch network. The ArcView *FlowDirection* request could be successfully applied to this pseudo-DEM, but would work not on the ditch network itself, which remained flat.

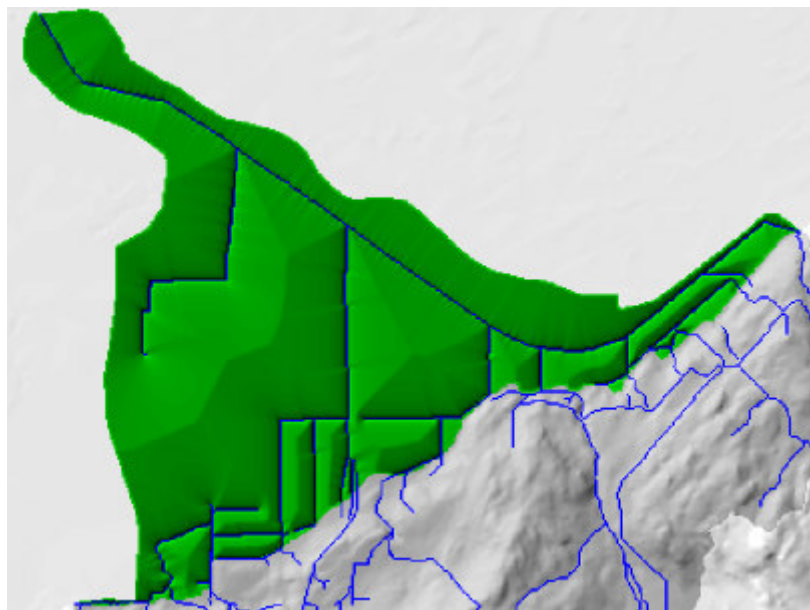


Figure 30: pseudo DEM of the valley based on the distance to the closest ditch

- Artificial flow direction for the ditch network

I wrote another script, which connects together the cells of the ditch network grid. At the beginning, all the stream cells have a value of 0, except the user-defined outlet cell, which has the user-defined flow direction value.

The algorithm starts with the outlet cell and connects the adjacent orthogonal and diagonal stream cells (the adjacent zero-valued cells). It loops on, connecting zero-valued stream cells to valued stream cells, so the algorithm progression is the exact opposite of the natural flow direction. Once again, as ArcView does not allow single cell value access, the script has to compute temporary grids for each loop and run boolean tests to attribute the right flow direction value to the right cell. Nevertheless, the ditch network was finally assigned flow direction values in 559 loops, which required a 13h40 run!

It produced unwanted artefacts where 2 reaches would merge, because through the algorithm looping process, one reach would reach the merging point sooner than the other one, and starts connecting downstream cells of the other reach to this normally higher point. Hence, one has to check the resulting flow direction grid if there are such mergings in the stream network grid and correct them manually. Hopefully, this happened only once for the valley ditch network.

Flow direction across the mountainous area

The original Interception Ditch watershed borders were contained in a shapefile provided by Chilliwack City. They were kept as such in the valley, but I redrew them in the mountainous area with the *flowDirectionGrid.Watershed* request applied to each cell of the separating line between the mountainous area and the valley area. The “new” borders matched quite well the shapefile borders.

In order to check the relevancy of the mountains flow direction grid, I compared the drainage network derived from the DEM with the real stream network to see how they match (cf. figure 29). In the forested area, they match really well, sometimes perfectly, despite a few shortcuts instead of small meanders probably due to elevation data sampling and smoothing.

The biggest differences, and a priori discrepancies, took place again in the urbanised area because of the road ditches. On rough terrain, roads are often built along contour lines so they are not too steep when driven uphill and safe enough when they get slippery during precipitation. Therefore, they often are perpendicular to the steepest slope direction, and from an hydrological point of view, act as a dam for runoff and groundwater. The pressure of accumulated water during big rainfall events can be such that it can push and break the road and trigger a landslide. To prevent that, ditches are dug along roads to collect this water, which can be conveyed to the other side of the road through a culvert, or to detention ponds and rivers.

Therefore, the flow direction grid needed to be modified across the mountainous area too. That required 3 steps:

i first I had to make sure that the flow direction grid cells that belong to the stream network were showing the correct flow direction value. This was done using the “connecting cell flow direction” script (used to compute ditch network flow direction in the valley) with the mountain stream network and the resulting stream flow direction grid was then burnt into the whole mountain flow direction grid.

ii meanwhile, I had to make sure that elevation data was consistent with the new stream flow direction grid. As road ditches are perpendicular to slopes, many times they would flow slightly upslope according to the sampled elevation data set. As I want to use the slope to compute flow speed, I had to modify elevation values to actually *force* the water to flow downstream. Therefore, I lowered elevation values of more than 130 cells while making sure that neighbouring downstream cells were still downstream. Elevation change never exceeded 1 metre and mean change is 23 centimetres, both values being below the vertical accuracy of the DEM.

iii finally, in some cases, I had to modify some flow direction values of the non stream cells that are meant to flow into stream cells. When main flow direction would be diagonal (north-west for example) and road ditch flow direction would be perpendicularly diagonal (north-west), every second upslope neighbouring cells would flow downhill past the ditch (see **figure 31**). Therefore, I modified 232 flow direction values to make sure that such cells flow into the stream network

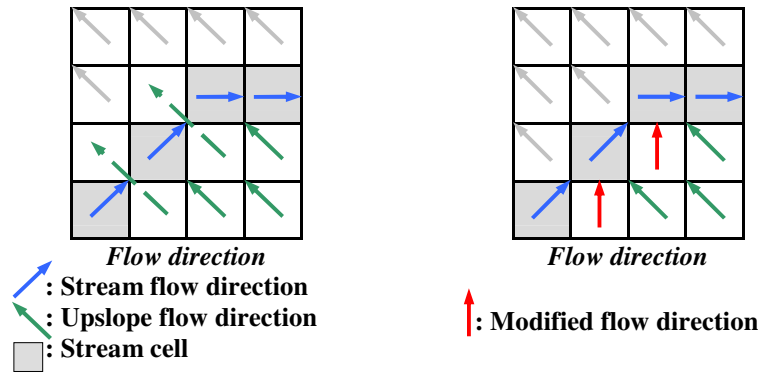


Figure 31: flow direction correction for upslope stream neighbouring cells

The last two steps were done manually because I could not find a reasonable way to implement such corrections with scripts. Because grids are not manually editable as such in ArcView, one has to convert them to a point theme, modify the value attribute of the points, and then convert them back to a grid. Modifying grids is in that sense a very painful task and it is very time-consuming to work at the cell scale. However, once it is done, the flow direction grid is not likely to change unless new road ditches are dug. The user of the model has to be aware that drawing new roads in the road polyline shapefile only affects soil imperviousness and runoff generation but do not modify flow routing direction.

III.3.3) Runoff routing speed

In order to use the FlowLength request on the flow direction grid to draw time zone areas instead of travel length zone areas, it is necessary to compute a travel time cost grid. In such a grid, each cell value would represent the amount of time required to move out from this cell to the destination cell specified by the flow direction grid.

A time can be expressed as a distance divided by a speed. Since we already know the distance through the cell size and the flow direction, an estimation of the speed is required.

III.3.3.1) Speed modeling

I chose the same physical approach to model both channel and overland flow speed. The system used for this demonstration is a volume of 1 kg of water ($m = 1 \text{ kg}$), represented as a point (C: gravity centre) flowing on a sloped plane (slope = α) and submitted to 3 forces (cf. figure 32):

- its weight, $\mathbf{W} = m \cdot g = 1 \cdot g = g$, where g is the acceleration of gravity and assumed constant,
- the normal force, \mathbf{N} , which is the support force exerted by the plane upon the volume of water. \mathbf{N} is orthogonal to the plane surface,
- the friction force, \mathbf{F} , which is the force exerted by the plane surface as the water moves across it. In this case, I chose to express \mathbf{F} in a simple way: $\mathbf{F} = -k \cdot \mathbf{v}$ where k is a positive friction coefficient partially depending on the plane texture, \mathbf{v} is the speed of the water volume and the minus at the front shows that friction is a resistant force opposed to motion,

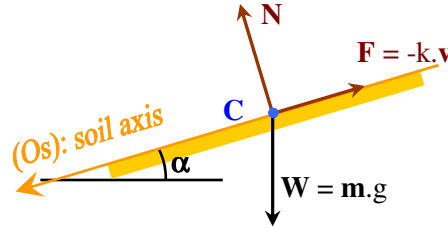


Figure 32: forces applied to a volume of water flowing on a sloped plane

According to *Newton's second law of motion*, the relation between applied forces and the derivative of momentum is:

$$(m.v)' = \mathbf{N} + \mathbf{W} + \mathbf{F} \text{ where } \mathbf{v}, \mathbf{N}, \mathbf{F} \text{ and } \mathbf{W} \text{ are vectors.}$$

The mass of water is supposed constant and at the scale of a cell, I assume that the volume of water has reached its speed limit so \mathbf{v} is assumed constant too. Therefore, $(m.v)'$ is the null vector. The equation becomes:

$$\mathbf{N} + \mathbf{W} + \mathbf{F} = \mathbf{0}$$

As such, normal force \mathbf{N} is orthogonal to the plane surface and perpendicular to soil axis (Os) , therefore its value is zero when the previous vector relation is projected on (Os) :

$$0 + W.\sin\alpha - F = 0 \text{ where } W \text{ and } F \text{ are the magnitudes of } \mathbf{W} \text{ and } \mathbf{F} \text{ vectors}$$

If we express W and F , we obtain the following relation between gravity, slope, friction coefficient and V_s (speed of the water volume across the soil axis):

$$m.g.\sin\alpha - k.V_s = 0$$

So V_s can be expressed as:

$$V_s = \frac{m.g.\sin\alpha}{k}$$

Nevertheless, as the distance between cell centres is not the real distance, which also accounts for elevation change between centres but the distance between their projections on the horizontal plane, we must consider V_x , the water speed projected on the horizontal plane, instead of V_s :

$$V_x = \frac{m.g.\sin\alpha.\cos\alpha}{k}$$

$$V_x = V_s.\cos\alpha$$

The following trigonometric identity:

$$\sin\alpha.\cos\beta = \frac{1}{2}(\sin(\alpha + \beta) + \sin(\alpha - \beta))$$

Here, $\beta = \alpha$ simplifies the expression of V_x :

$$V_x = \frac{m.g}{2k} \sin 2\alpha$$

Comparisons with speed values estimated with Manning's formula showed the $\sin 2\alpha$ factor in V_x expression tended to decrease speed values too much as the slope would diminish. Moreover, friction coefficient k is likely to vary with the slope, increasing when the speed would increase and decreasing. Therefore, I decided to express k without more physical justification as follows:

$$k = C \cdot \sqrt{\sin 2\alpha}$$

where C is a constant. The final detailed expression of V_x is as follows:

$$V_x = \frac{m \cdot g}{2C} \sqrt{\sin 2\alpha}$$

As a function of α , V_x hits its highest for $\alpha = 45^\circ$ and $\sin 2\alpha = 1$. In that case, $V_{x_{45}}$ (for $\alpha = 45^\circ$) can be expressed as follows:

$$V_{x_{45}} = \frac{m \cdot g}{2C}$$

Finally, V_x can be expressed in the following rather intuitive form:

$$V_x = V_{x_{45}} \sqrt{\sin 2\alpha}$$

where $V_{x_{45}}$ is the speed (projected on the horizontal plane) at which water flows on a certain type of terrain sloped at 45° .

III.3.3.2) Distributed speed grid

Though V_x expression is theoretically a function of m (the mass of the water volume), or a function of the height of channel and overland water, I decided to keep it as such because the height of water contained in each cell would change every time cells flow into each other. Computing height, would require a finite-element approach, which is typically the kind of calculation I tried to avoid since GIS are not meant for that. The speed grid results from the produce of the $V_{x_{45}}$ grid by the square root of twice the slope grid.

As $V_{x_{45}}$ is obviously a function of surface roughness, a new field was added to the vegetation LUTab (which is the lookup table used to account for land use characteristics) to store $V_{x_{45}}$ values for each class of land use. The priority order between the land use polygon shapefiles is the same as the one for the vegetation interception grid:

- stream network cells are first attributed their V_{45} value,
- then the cells contained in totally impervious or semi-impervious areas (urban shapes). The assumption here is to compute V_{45} value linearly with the imperviousness percentage and 2 extreme values: $V_{45paved}$ for totally impervious surface and $V_{45meanGrass}$.
- bare grounds,
- crops,
- vegetation,
- likewise, the rest is filled automatically with the $V_{45meanGrass}$ value.

Then, where slope information is available (across the mountainous area), the distributed speed grid is computed with the following Avenue command:

$$\text{speedGrid} = \text{V45Grid} * (\text{slopeGrid} * 2 * \text{AsGrid}) * \text{Sin.Sqrt}$$

Across the valley area, speed values are arbitrarily set to 0,5 m/s for the ditches and to 0,05 m/s. As the inverse of *speedGrid* values will be used as the impedance grid to compute the time-to-the-outlet-grid (cf. the **flowTimeOutletGrid** Avenue command below), I had to make sure that the speedGrid do not contain any zero (over filled sinks for example). Therefore, all the speed cell value below 0,05 m/s were set to 0,05 m/s.

flowTimeOutletGrid = flowDirectionGrid.FlowLength(1.AsGrid/speedGrid,upStream)
with upStream = false

III.3.4) Drawing the time zone area diagram

The flowTimeOutletGrid is then sliced into time zones on the basis of the time step duration used in the rain series table. Though production and transfer functions are conceptually independent, they are actually implemented together in order to save RAM (Random Access Memory) resource. Here are basically the steps actually followed during a simulation:

- i* the watershed boundaries are drawn for the user-defined outlet,
- ii* all necessary grids are computed (land use, soil characteristics, speed...),
- iii* time zone areas are drawn and a new *Runoff Vtab* is created: it contains one field for time step numbers, one field for each time zone area, one field for gross precipitation mean height, one field for net rain mean height and a last one for generated runoff mean height,
- iv* the first record is filled with time zone area surface expressed in ha,
- v* and then for each rainfall time step:
 - a* calculation of the distributed runoff grid,
 - b* the mean height of runoff is computed for each time zone area and written in the respective time zone field (see *iii*),
 - c* the mean gross precipitation height, the mean net precipitation height and the mean runoff height values are written in the last three field of the *Runoff Vtab*.
- vi* the runoff contribution hydrograph is drawn by cross-summing the amount of runoff (each volume being the mean runoff height times the time zone area surface).

The iterative process described in *v* allows the script to use the same runoff grid at each iteration by writing the relevant information in the *Runoff Vtab*. See **appendix 7** for a more detailed manual, which explain how to use UHModel application and its graphic interface.

III.4) First results

Before I tested the model on real rainfall data, I did a couple of sensitivity tests to check how the model reacts to different initial conditions and land uses.

III.4.1) Sensitivity tests

Among the different parameters, which are likely to influence the hydrological response of a watershed, I focused on soil water storage capacity, vegetation interception and imperviousness. As I also wanted to test the influence of roughness on flow speed (with the V45 coefficient), I used the south-east subwatershed where the DEM is relevant and the slope thus accessible (cf figure 33).

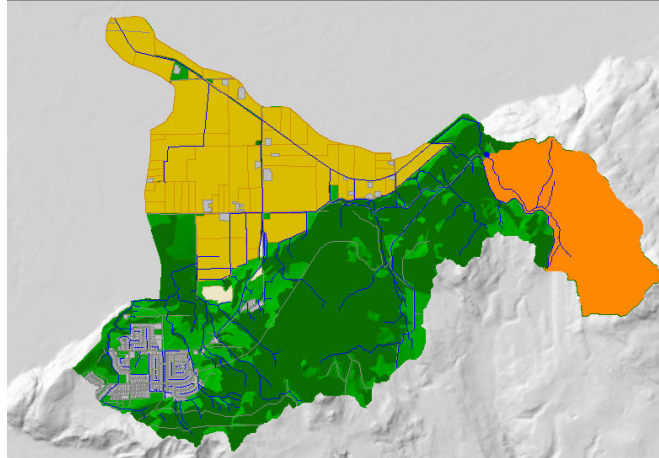


Figure 33: location and delineation of the subwatershed used for sensitivity tests

All the sensitivity tests were run with the same rainfall series: the virtual rainfall table used has 11 records of precipitation, each one lasting 15 minutes (it is the time step duration) and accounting for 0,5 mm of rainwater. The soil infiltration rate mode was set to “median”, resulting in soils having infiltration rate values ranging from 0,7 cm/h to 2 cm/h.

[III.4.1.1)The influence of soil water storage capacity

Although porosity is an inherent property of a soil resulting from its composition and its compaction, its initial water storage capacity before a rainfall event can be subject to strong variations due to initial soil moisture. For this first series of tests, I set the land use of the whole catchment to bare ground to avoid any vegetation interception. Five simulations were launched with soil initial moisture values ranging from 60 % to 100 % (cf figure 34).

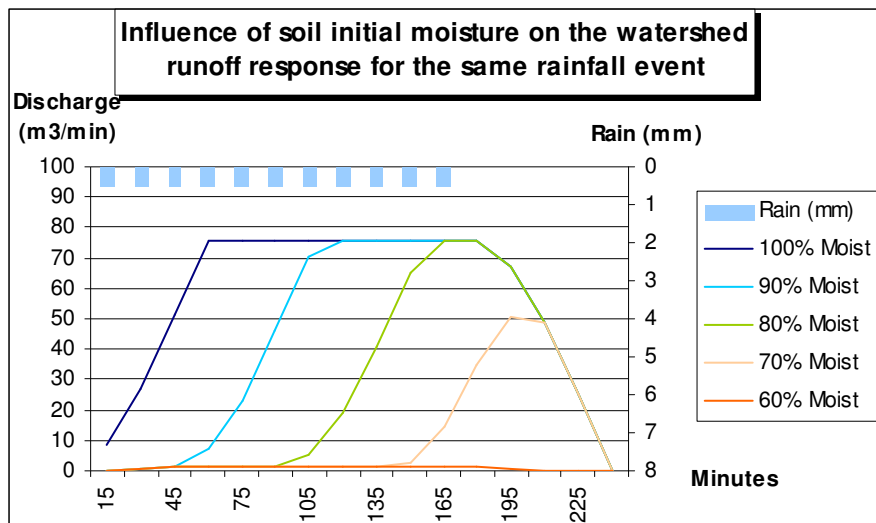


Figure 34: influence of soil moisture on runoff contribution to discharge

The 100 % Moist curve represents the behaviour of a saturated soil, which cannot absorb any water, therefore all the rainwater goes as runoff. As I used a constant rainfall as an input, the curve soon reaches a plateau, which depicts full runoff generation. The next curves show the retention effect of soil storage capacity. As the field of speed is modeled constant in time, the lags between the times to speak of each curve is linear with initial soil moisture: the bigger the soil buffer gets, the longer the soils get saturated and generate full runoff. During the last two simulations (70 and 60 % of initial moisture), the rainfall event was not long enough to fully saturate the soils, therefore, there is no plateau but a peak for the 70 % Moist curve and almost no runoff generation for the 60 % Moist curve.

[III.4.1.2) The influence of vegetation

As far as UHModel is concerned, vegetation is characterised by 2 parameters: vegetation interception and overland water flow speed (with the V45 value). A series of 3 simulations were done to test the effects of three vegetation covers (cf **figure 35**): bare ground (absence of vegetation), grass and forest (the latter two ones being used in the model for any forest and for any undefined land use –cf section III.2.1.2).

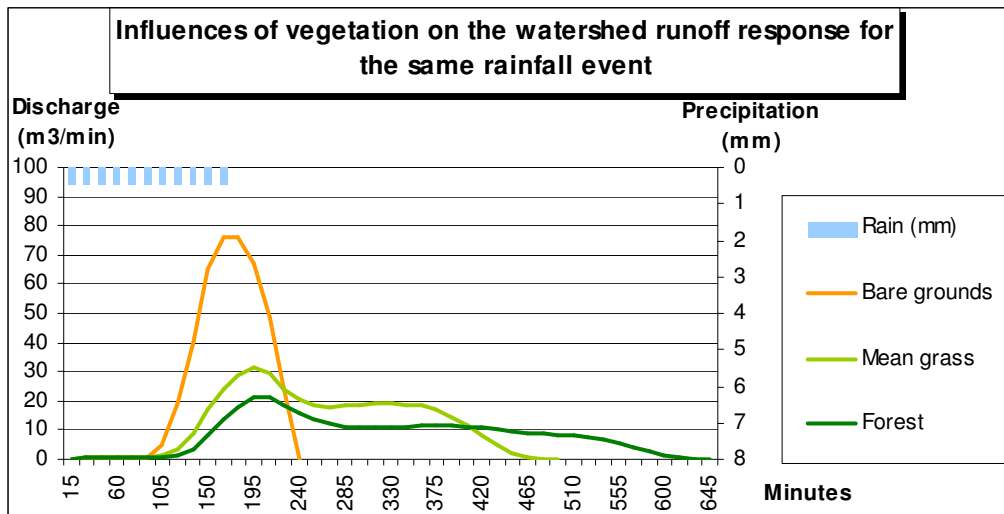


Figure 35: influence of vegetation on runoff contribution to discharge

The three simulations were conducted with an initial soil moisture of 80 %, which leaves only very little soil water storage capacity. However, as the rainfall event is rather short, the *bare ground curve* only reached a short full runoff generation plateau. The unexpected shape (a peak and a plateau) of the mean grass and forest curves can be partially explained by the distribution of infiltration rate values across this specific subwatershed (roughly, there is one set of soils, which have an infiltration rate value of 2 cm/h and may account for the peak since they will get saturated earlier, whereas the other set has the much lower infiltration value of 0,7 cm/h). However, hydrograph peaks generated over vegetated land have a lower intensity due to interception and are spread over a longer period because of a slower overland flow. The lag to discharge peak for vegetated cover is not very longer than the one for bare soil, but it must be just because the water storage capacity available is low.

[III.4.1.2) The influence of imperviousness

Two more tests were done with a longer rain series and compared with the *bare grounds curve* to assess the effect of soil imperviousness (bare grounds are considered totally pervious): one test was with a soil imperviousness of 50 % and the other one was considered totally paved and

impervious. At its current stage, speed values on bare grounds are considered the same than over paved surface, therefore we only assess the effect of imperviousness here (cf **figure 36**). Here again, soil initial moisture values were all set to 80 %.

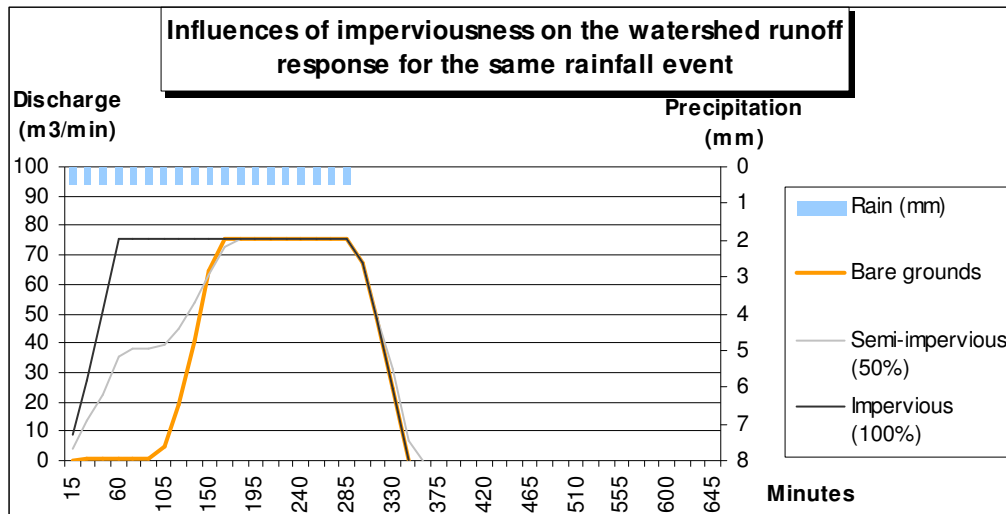


Figure 36: influence of vegetation on runoff contribution to discharge

The same final plateau discharge value for all the curves shows the time when full runoff generation affects the discharge at the outlet. Nevertheless, the *semi-impervious* curve shows a first plateau, which has value exactly 50 % of the higher plateau. This can be explained by how the model interprets semi-impervious areas (cf **section II.3.4**). Basically, if an area is 50 % impervious, then 50 % of the rain, which falls on it, goes as runoff, the other 50 % can infiltrate the soil or produce excess runoff. Therefore, in this case, the runoff hydrograph is the result of the addition of 2 different hydrographs: one from a watershed, which would be 100 % impervious, and another one which would be totally impervious, but each one of them would receive only half of the precipitation. Therefore, there is this first plateau from the impervious watershed while the pervious one is absorbing rainwater until saturation.

III.4.2) Confrontation with reality

Now that we have checked that the model reacts in a logical way to different watershed settings, it is necessary to observe the model output when it is given real precipitation data. Both rainfall and hydrometrics data were provided by Chilliwack City as spreadsheets.

III.4.2.1) Origin of the data

Rainfall data were extracted from a tipping bucket rain gauge located in the south-west urbanised subwatershed with a time step duration of fifteen minutes and provided as such to me. The rainfall event chosen for the simulation occurred between Saturday, 27th October 2 PM and Sunday, 27th 10:15 AM, resulting in a table of 82 precipitation records. I assumed that those values can be considered as the gross precipitation value inputs to the model and that this value can be extrapolated to the simulation subwatershed, which is not unrealistic since there is only 5 kilometres between the rain gauge and the watershed, and the rain gauge elevation (156 m) fits in the elevation range of the watershed (132-670 m).

Within Interception Ditch watershed, there are 8 hydrometric stations, which derive flow rate data from pressure probes located in normalised culverts. I chose one, which matched in time and in time step duration with the rainfall data. Moreover, I did not want a watershed too large or too complex in terms of land use for a first simulation. Actually, the chosen hydrometric station is

located at ElkView Creek and Wincott Road and also is the outlet of the watershed used for the test series (cf. **figure 33**). The resulting catchment area is 227,5 ha wide.

III.4.2.2) First run

The parameters used for the first run were:

- infiltration mode: median
- soil moisture mode: 50 %
- water table mode: low (but this parameter is meaningless in this simulation since there is no water table in this sub watershed)

Any other parameter values were the ones set in the lookup tables (cf. **appendix 3**). The simulation run took about 3 hours on a Pentium II 266 MHZ with 256 MB of RAM and here is the resulting hydrograph:

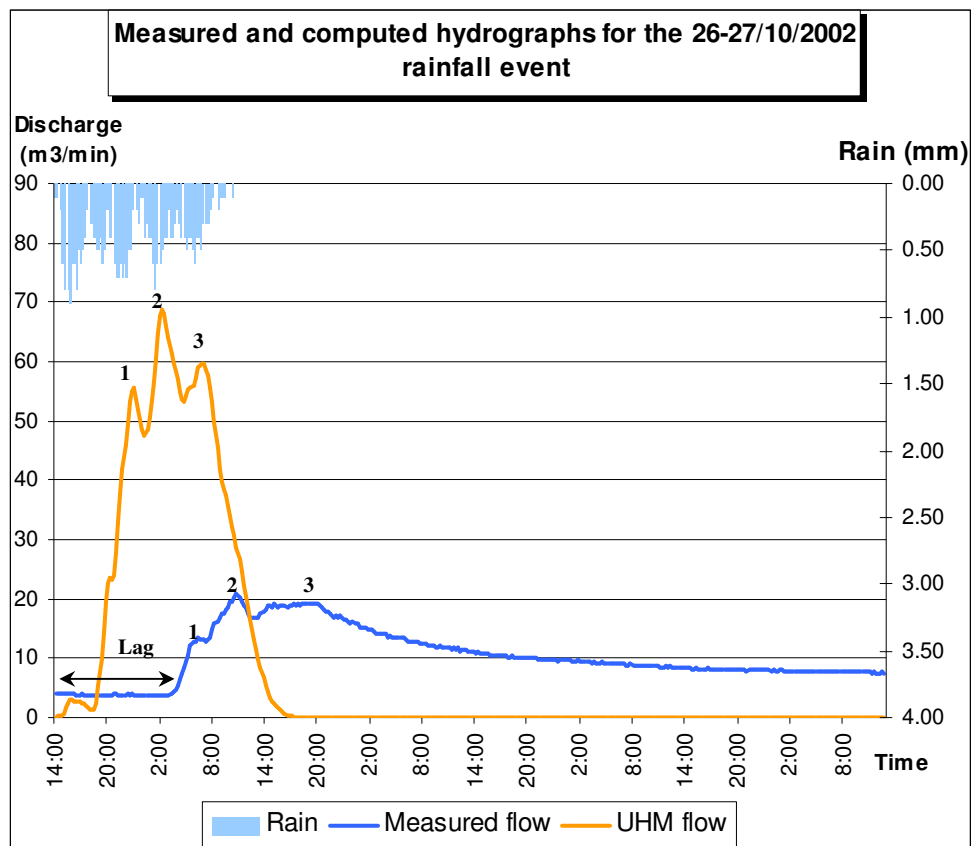


Figure 37: runoff contribution hydrograph of the 26-27/10/2002 rainfall event, simulated by UHModel and the measured hydrograph, first run

The first observations are that the simulated hydrograph is much bigger and in advance in comparison with the real discharge hydrograph. Since the simulated hydrograph should only be the runoff contribution to the stream discharge, it should be even smaller than the measured hydrograph. As the UHM flow curve only represents runoff contribution, its recession limb falls steeper than the measured flow curve recession limb, which falls much slower because of increased base flow contribution.

However, despite the scale discrepancies, the shape of the simulated hydrograph does have some characteristics in common with the measured flow hydrograph: 3 peaks can be seen on both curves with the same relative intensity (the 1st peak has the lowest intensity, the 2nd has the highest one and the 3rd is in between). A more detailed analysis (cf **figure 38**) shows:

- whereas the hydrograph lag (time between the beginning of the rainfall and the beginning of the raising limb of the hydrograph) of the simulated hydrograph is around 4,75 hours, which is consistent with what one can expect for such a small and mountainous catchment, the measured hydrograph lag only occurs 13,75 hours after the beginning of the rainfall, which seems very late. The nine hours difference could not be explained by a delay in the start of the rainfall between the rain gauge watershed and the simulation watershed.
- the time difference between apparently similar peaks of simulated and measured hydrographs increases slightly between 7,75 to 10,5 hours. The 3 simulated peaks being too close to each other can be interpreted by an overestimated flow speed. Decreasing V45 values in the lookup table may split the peaks further apart and increase the hydrograph lag.
- the simulated peaks intensity is about 3 or 4 times bigger than the measured ones. Vegetation interception might be underestimated, but it would never compensate such a difference. However, as I had to set an arbitrary thickness to the deepest accessed horizon of each soil, I may have underestimated the total soil thickness and therefore, the soil water storage capacity. Furthermore, one of the model outputs is a distributed grid of accumulated infiltrated water (cf. **appendix 7**) and I noticed that all the soils were totally saturated at the end of the simulation. Therefore, the peaks were generated by total runoff generation on saturated soils rather than by rainfall intensity exceeding infiltration rate. Last but not least, the soil moisture was set to 50 %, which may have been too much.

Event	UHM Time	Real Time	Time difference	UHM Value	Real Value	UHM/Real
Hydrograph lag	18:45	3:45	9:00			
1st peak	23:00	6:45	7:45	55.52	13.30	4.17
2nd peak	2:30	11:00	8:30	67.98	20.08	3.39
3rd peak	7:15	17:45	10:30	58.77	19.33	3.04

Figure 38: occurring times and intensities of simulated and measured hydrographs.

A second run on this rain data set was launched with modified parameters.

[III.4.2.3) Second run

The parameters used for the second run were:

- infiltration mode: median
- soil moisture mode: **10 %**, instead of 50 %
- (- water table mode: low)

And a few lookup table values were modified too:

- V45 speed values were lowered: from 0,07 to **0,03 m/s** for **forest** overland flow speed and from 0,14 to **0,1 m/s** for **grass**,
- **forest interception** rate was increased from 9 to **18 %**,

The second hydrograph (cf. **figure 39**) is a little bit better in terms of hydrograph lag since the time difference has been reduced by half. However, the discharge peak intensity ratio is still too high (around 2,5) although the cumulated effect of initial soil moisture and vegetation interception are visibly considerable.

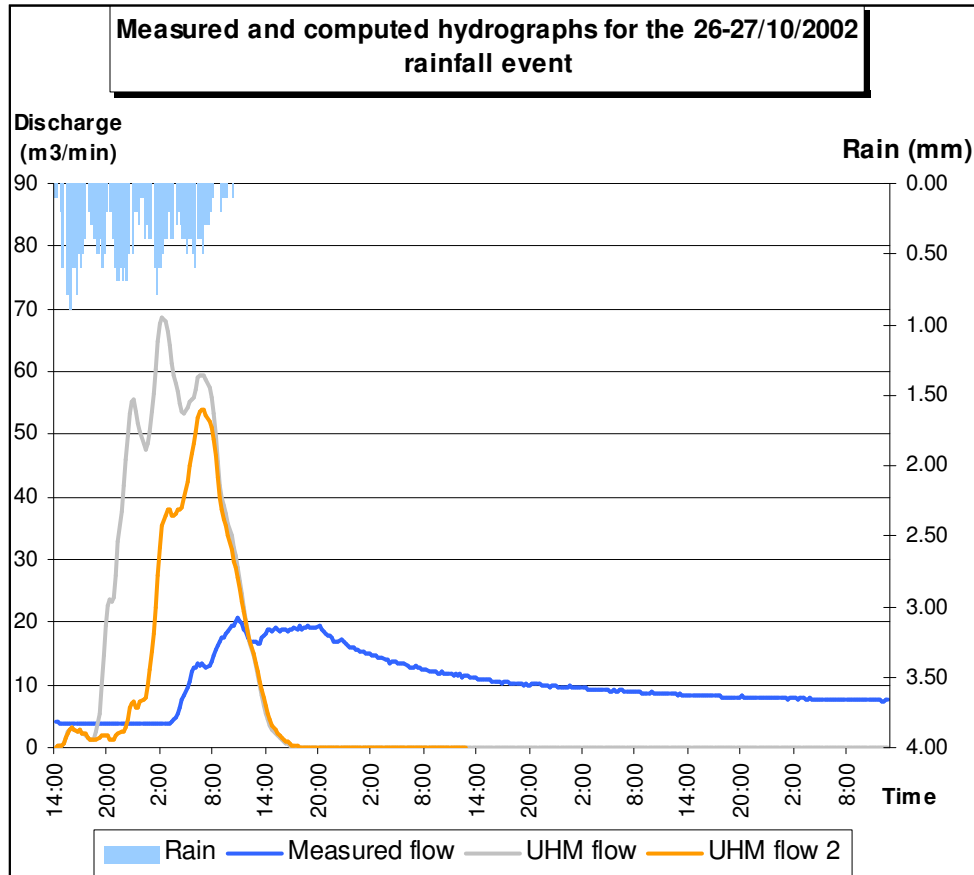


Figure 38: runoff contribution hydrograph of the 26-27/10/2002 rainfall event, simulated by UHModel and the measured hydrograph, second and first runs

Unfortunately, I haven't had more time to do more simulations on other rainfall data sets or with other modifications. However, indepth calibration could be done in many ways:

- by modifying speed values and comparing various time differences, such as hydrograph lag times, occuring time difference between similar peaks of simulated and measured peaks and so on. Easy experiments can easily be done in the laboratory to attribute better values for the V45 coefficients, since I represents the speed (projected on the horizontal plane) at which runoff flows over a plane sloped at 45 °.
- by modifying infiltration rate modes and values. Experiments on field or in the lab could be done, such as infiltration rate measurements.
- vegetation interception: rain gauges could be set inside and outside the forest to measure throughfall and interception loss.
- newer soil augering and analyses could be done to have a better description of the soil distribution, but I think that the knowledge gained would be very little compared with the effort it requires.

Conclusion

Throughout my training period at the Institute for Resources and Environment, I managed to prove that simple, yet powerful hydrological concepts could be implemented in a GIS originally not designed to support hydrological modeling. Moreover, on the top of that, I tried to show that GIS might be the key hardware and software environments for future development of hydrological distributed models since:

- GIS relational database management was used,
- GIS typical spatial objects like shapes and grids are involved at their most efficient level,

Through implementation problems, I have also understood from a new point of view how human activities and urbanisation affect natural hydrological processes. Although impacts such as the increase of soil imperviousness and flow direction disturbance require a huge amount of time to correct and modify data in consequence, the flexibility offered by a GIS environment like ArcView to easily design new land uses is definitely an advantage of GIS-based hydrological modeling.

The first sensitivity tests and simulations showed that the theory and the theoretical values for parameters used in the Unit Hydrograph Model are not completely unrealistic, although further calibration is necessary before the model can be used as a prediction tool. However, I have tried to make calibration processes simpler by giving easy and clear access to them by the means of lookup tables and graphic menus.

Appendices

Appendix 1 Agricultural Land Reserve

The province of British Columbia is very mountainous and therefore only 5% of its land are suitable for farming, according to criteria based on topography, soils and climate characteristics, and drainage. As most of the best agricultural lands are in narrow valley bottoms, where they must compete with other uses (such as housing, industry, transportation...), the BC Government had to protect those lands in order to ensure long term viability for its agriculture and food self-sufficiency.

Since 1973, lands have been classified all over BC according to their agricultural potential and limitations using the following rating class system:

Agricultural Capability Classes	
Classes	Description
1	Class 1 land is capable of producing the very widest range of crops. Soil and climate conditions are optimum, resulting in easy management.
2	Class 2 land is capable of producing a wide range of crops. Minor restrictions of soil or climate may reduce capability but pose no major difficulties in management. Class 3 land is capable of producing a fairly wide range of crops under good management practices. Soil and/or climate limitations are somewhat restrictive.
3	Class 3 land is capable of producing a fairly wide range of crops under good management practices. Soil and/or climate limitations are somewhat restrictive.
4	Class 4 land is capable of a restricted range of crops. Soil and climate conditions require special management considerations.
5	Class 5 land is capable of production of cultivated perennial forage crops and specially adapted crops. Soil and/or climate conditions severely limit capability.
6	Class 6 land is important in its natural state as grazing land. These lands cannot be cultivated due to soil and/or climate limitations.
7	Class 7 land has no capability for soil bound agriculture.

Source: BC Land Reserve Commission

Together with the class system, there are subclasses, which identify limitations or special management practices needed to improve the soil (such as topography, stoniness, soil moisture deficiency, low fertility...) for farming:

Agriculture Capability Subclasses			
Class	Description	Class	Description
A & M	Soil moisture deficiency	S	Salinity
C	Adverse climate (excluding precipitation)	P	Stoniness
D	Undesirable soil texture	R	Shallow soil over bedrock and/or bedrock outcroppings
E	Erosion	T	Topography
F	Low Fertility	W	Excess water (groundwater)
I	Inundation (flooding by streams, etc.)	S & X	Cumulative and minor adverse conditions

Source: BC Land Reserve Commission

Appendix 2 Interception storage capacity

Interception storage capacity for different vegetation types (after Horton, 1919; Leyton and Thompson, 1967; Zinke, 1967; Rutter and Morton, 1977; Herwitz, 1985). Table compiled by Morgan and Rickson, 1995.

Vegetation type	Interception storage capacity, IC_{max} (mm)
Fescue grass	1.2
Molinia	0.2
Rye grass	2.5
Meadow grass, clover	2.0
Blue stern grass	2.3
Heather	1.5
Bracken	1.3
Tropical rain forest	0.8-2.5
Temperate deciduous forest (summer)	1.0
Temperate deciduous forest (winter)	0.5
Needle leaf forest (pines)	1.0
Needle leaf forest (spruce, firs)	1.5
Evergreen hardwood forest	0.8
Soya beans	0.7
Potatoes	0.9
Cabbage	0.5
Brussels sprouts	1.0
Sugar beet	0.6
Millet	0.3
Spring wheat	1.8
Winter wheat	3.0
Barley, rye, oats	1.2
Maize	0.8
Tobacco	1.8
Alfalfa	2.8
Apple	0.5

Appendix 3 Lookup tables used in the Unit Hydrograph Model

Vegetation interception lookup table

DEFINITION	VEG_KEY	INT_P100	V45_M_S
Hardwood - growing season	hwgs	9.90	1.00
Hardwood - dormant season	hwds	8.60	1.00
Kauri forest	krft	40.00	1.00
Manuka shrub	mksh	56.00	1.00
Mountain beech forest	mbft	31.00	1.00
Mean forest	mnft	9.00	0.07
Mean grass	mngr	3.00	0.14
Mean crop	mncp	6.00	1.40
Bareground	grnd	0.00	3.00
Streams, rivers, ponds... water	H2O	0.00	6.00
Paved areas, concrete...	ccrt	0.00	3.00

Urban lookup table

DEFINITION	URBAN_KEY	IMP_P100
High density residential	hdr	65.00
Low density residential	ldr	25.00
Industrial	ind	72.00
Parking, paved spaces (paved roads...)	imp	100.00
Residential 8000m2 lot size (2 acres)	rd80	12.00
Residential 4000m2 lot size (1 acre)	rd40	20.00
Residential 2000m2 lot size (1/2 acre)	rd20	25.00
Residential 1333m2 lot size (1/3 acre)	rd13	30.00
Residential 1000m2 lot size (1/4 acre)	rd10	38.00
Residential 500m2 lot size (1/8 acre)	rd5	65.00
Unpaved road (dirt, gravel...)	uprd	60.00

Soil texture lookup table

TEX_CSSC	INF_CM_H	WST_CM_M
S	20.00	32
LS	5.00	37
FSL	3.00	37
SL	2.60	31
L	1.30	30
SIL	0.70	31
SCL	0.43	19
CL	0.23	26
SICL	0.15	23
SC	0.13	19
SIC	0.10	21
C	0.05	19
P	3.00	80
HC	0.10	20
LFH	3.00	50

Soil moisture lookup table

MOIST_KEY	DEFINITION	MOIST_P100
A	Xeric	0.00
B	Arid	0.00
C	Subarid	0.00
D	Semiarid	0.00
E	Subhumid	0.00
F	Humid	0.00
G	Perhumid	0.00
H	Subaquic	0.00
I	Aquic	0.00
J	Per aquic	0.00

Appendix 4 USDA Soil Conservation Service curve number method

$$Q = \frac{(P - I_a)^2}{(P - I_a) + S} \quad (1)$$

Q = runoff (in)
P = rainfall (in)
S = potential maximum retention
after runoff begins
 I_a = initial abstractions

The SCS curve number method is a simple, widely used and efficient method developed by the United States Department of Agriculture for determining the approximate amount of runoff generated from a rainfall event in a particular area. Although the method is designed for a single storm event, it can be scaled to find average annual runoff values. The data requirements for this method are very low, rainfall amount and curve number. The curve number is based on the area's hydrologic soil group, land use, treatment and hydrologic condition, the 2 former being of greatest importance.

$$I_a = 0.2 S \quad (2)$$

The general equation for the SCS curve number method is as follows:

$$Q = \frac{(P - 0.2 S)^2}{(P + 0.8 S)} \quad (3)$$

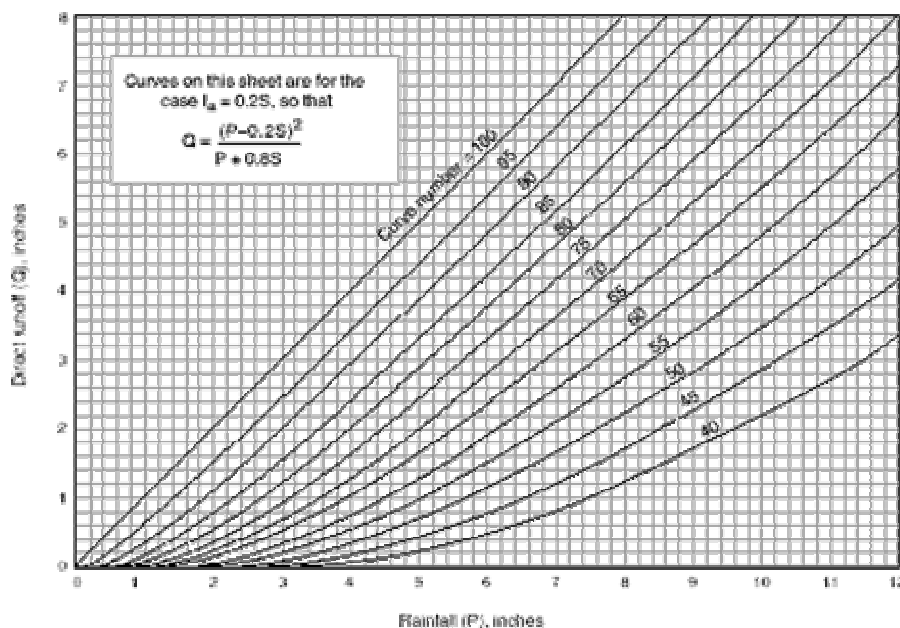
The initial equation (1) is based on trends observed in data from collected sites in the US, therefore it is an empirical equation instead of a physically based equation. After further empirical evaluation of the trends in the database, the initial abstractions, I_a , could be defined as a percentage of S (2).

$$S = \frac{1000}{CN} - 10 \quad (4)$$

With this assumption, the equation (3) could be written in a more simplified form with only 3 variables. The parameter

CN is a transformation of S, and it is used to make interpolating, averaging, and weighting operations more linear (4).

With the following chart, the amount of runoff can be found, if the rainfall amount (in inches) and curve number is known.



Land Use Description on Input Screen	Description and Curve Numbers					
	Cover Description		Curve Number for Hydrologic Soil Group			
	Cover Type and Hydrologic Condition	% Impervious Areas	A	B	C	D
Agricultural	Row Crops - Straight Rows + Crop Residue Cover-Good Condition (1)		64	75	82	85
Commercial	Urban Districts: Commercial and Business	85	89	92	94	95
Forest	Woods(2) - Good Condition		30	55	70	77
Grass/Pasture	Pasture, Grassland, or Range(3) - Good Condition		39	61	74	80
High Density Residential	Residential districts by average lot size: 1/8 acre or less	65	77	85	90	92
Industrial	Urban district: Industrial	72	81	88	91	93
Low Density Residential	Residential districts by average lot size: 1/2 acre lot	25	54	70	80	85
Open Spaces	Open Space (lawns, parks, golf courses, cemeteries, etc.)(4) Fair Condition (grass cover 50% to 70%)		49	69	79	84
Parking and Paved Spaces	Impervious areas: Paved parking lots, roofs, driveways, etc. (excluding right-of-way)	100	98	98	98	98
Residential 1/8 acre	Residential districts by average lot size: 1/8 acre or less	65	77	85	90	92
Residential 1/4 acre	Residential districts by average lot size: 1/4 acre	38	61	75	83	87
Residential 1/3 acre	Residential districts by average lot size: 1/3 acre	30	57	72	81	86
Residential 1/2 acre	Residential districts by average lot size: 1/2 acre	25	54	70	80	85
Residential 1 acre	Residential districts by average lot size: 1 acre	20	51	68	79	84
Residential 2 acres	Residential districts by average lot size: 2 acre	12	46	65	77	82
Water/ Wetlands		0	0	0	0	0

Notes:

(1) Hydraulic condition is based on combination factors that affect infiltration and runoff, including (a) density and canopy of vegetative areas, (b) amount of year-round cover, (c) amount of grass or close-seeded legumes, (d) percent of residue on the land surface (good \geq 20%), and (e) degree of surface roughness.

(2) Good: Woods are protected from grazing, and litter and brush adequately cover the soil.

(3) Good: $>75\%$ ground cover and lightly or only occasionally grazed.

(4) CN's shown are equivalent to those of pasture. Composite CN's may be computed for other combinations of open space cover type.

Appendix 5 The "aGrid.Con (yesGrid, noGrid)" request.

The following explanations are copy-pasted from the ArcView online help.

The aGrid.Con (yesGrid, noGrid) request performs a conditional if/else evaluation on a cell-by-cell basis. aGrid can be any valid Boolean or relational expression involving multiple Grids and Numbers (after being converted to Grids with aNumb.AsGrid). Cells for which the expression is TRUE are given the value found in yesGrid. Cells for which the expression is FALSE are given the value found in noGrid. If aGrid is a single Grid, then cells which are non-zero are given the value found in yesGrid, while values of 0 are given the value found in noGrid. The yesGrid or noGrid can be a single Grid, or any valid expression involving operators and requests that result in a Grid object.

A value of No Data in aGrid does not receive the value of the noGrid, it remains as No Data.

To understand the aGrid.Con request, you must first break it into smaller components. For example, in the expression:

`outGrid = (inGrid > 5.AsGrid).Con(100.AsGrid, 50.AsGrid)`

the "(inGrid > 5.AsGrid)" is the condition, "100.AsGrid" is what is returned if the expression is TRUE, and "50.AsGrid" is what is returned if the expression is FALSE. The logic underlying the expression is that if the value of inGrid is greater than 5, then assign the value of 100 to outGrid, and if the value of inGrid is 5 or less, then assign the value of 50 to outGrid.

A more complex example is an embedded aGrid.Con request within another aGrid.Con request:

`outGrid = (inGrid >= 20.AsGrid).Con((inGrid < 50.AsGrid).Con(40.AsGrid,
100.AsGrid),200.AsGrid)`

Here, the condition for the first aGrid.Con request is "(inGrid >= 20)". The other aGrid.Con request, "(inGrid < 50.AsGrid).Con(40.AsGrid, 100.AsGrid)", is returned if the condition is TRUE. If the condition is FALSE, then "200.AsGrid" is returned. The condition for the embedded aGrid.Con request is "(inGrid < 50.AsGrid)". If the condition for the embedded aGrid.Con request is TRUE, then "40.AsGrid" is returned. If it is FALSE, then "100.AsGrid" is returned. The logic of the expression follows: if the value of inGrid is greater than or equal to 20 but less than 50, assign the value of 40 to outGrid. If the value of inGrid is greater than 50, assign the value of 100 to outGrid. If the value of inGrid is less than 20, assign the value of 200 to outGrid.

Appendix 6 The Chilliwack TIN

The DTM supplied is a Triangulated Irregular Network made by John Wiley (the Chilliwack City Hall GIS supervisor) using Classic ArcInfo workstation and ArcGIS Spatial Analyst. It is based on the several data sources:

- the uplands data came from McElhanney geosurveys in 1989 and 1990 stereo plot with an accuracy of 2 metres from traditional aerial photography. the tops of Mt. Shannon (known as Little Mountain) and Chilliwack Mountain were done “in house” by Chilliwack City from survey plans done to 1/1 000 scale with 2 metre contours. Therefore, **vertical accuracy is 2 metres on the mountainous area.**

- the lowland was digitised from 1/5 000 floodplain maps done by the Province of BC in 1976, and historically this mapping was done every 10 years with **centimetre accuracy for spot elevations.** The cadastral was adjusted using GPS in 1995 and the algorithm performed well, matching McElhanney trilaterations perfectly.

Horizontal accuracy is less than 5 metres for the whole TIN.

Appendix 7 UHModel user's guide

Readers of this appendix must be familiar with a few ArcView concepts and vocabulary:

*Themes (Fthemes, Gthemes),
Shapefiles and grids,
Tables,
Views,
Scripts*

If not, please read section **III.1** of the report.

The model is an application (an ArcView project), which requires ArcView3.2 to be installed on the user's computer and ArcView Spatial Analyst extension to be loaded (otherwise ArcView is not able to handle grids). Beside those requirements, the model does not require any extra programs or application to be installed. UHModel.zip is a set of files and subfolders, which must be uncompressed where you want the main file to be launched and the simulations to be saved. This location in your file system is referred to as the **root** of the model. Uncompressing this file will create the following folders at the root:

Folder and subfolder organisation

└ DataDefault

This folder contains the land use files (there are Fthemes shapefiles and grids subfolders) and the tables I processed and produced for the model. Those also are the files loaded by default when the model is launched. If you process the files in this folder, you can store them here (therefore, they will be loaded by default) or in the DataUser subfolder. Unless you want to modify the default input files of the model, avoid storing files here. In order to be loaded automatically, the default themes and tables must have the following file names:

- the bare ground polygon shapefile prefix (a shapefile is a set of files, which start with the same prefix, but with different suffixes (shp, shx, sbx, sbn, dbf...)) must be "baregrounds",
- the crop polygon shapefile suffix must be "crops"
- the outlet point shapefile suffix must be "outlet"
- the roads polyline shapefile suffix must be "roads",
- the soils polygon shapefile suffix must be "soils",
- the urban polygons shapefile suffix must be "urbanpolygons",
- the vegetation polygon shapefile suffix must be "vegetation",
- the DEM grid dataset folder name must be "demhill",
- the flow accumulation grid dataset folder name must be "flaccwhole". This grid is not necessary for the simulation, but as it shows the natural drainage network derived from the flow direction grid, it may be interesting for the user for any modification of the stream network,
- the flow direction grid dataset folder name must be "fldr",
- the hillshade grid dataset folder name must be "hlshdwhole". This grid is only necessary to provide a 3-D look to the displayed Gthemes,
- the mountainous area extent mask grid dataset folder name must be "maskhill". This grid is not necessary for the simulation, it just shows the delineation of the DEM and sets a green background for this area,

- the stream network grid dataset folder name must be “maskstreams”,
- the valley area extent grid dataset folder name must be “maskvalley”,
- the Interception Ditch watershed extent grid data set folder name must be “maskwhole”. Combined with the hillshade grid, this grid just gives a 3-D background to the InputView.
- the horizon information Vtab must be stored as “infohorizons.dbf”,
- the soil information Vtab must be stored as “infoiltypes.dbf”,
- the soil moisture LUTab must be stored as “lutsoilmoisture.dbf”,
- the soil texture LUTab must be stored as “lutsoiltexture.dbf”,
- the urban/imperviousness LUTab must be stored as “luturban.dbf”,
- the vegetation/interception/V45 LUTab must be stored as “lutvegetation.dbf”,

└ DataUser

This folder is originally empty, but the user may store relevant files here. The model application never looks for files in this folder, but you can easily load files manually into the application.

└ Workspace

With DataDefault folder, WorkSpace is the most sensitive folder used by the model. When the model application is launched, all the temporary files the user might create (temporary shapefiles, temporary grids, temporary tables...) during simulations or when processing land use files are stored in this folder. Any new grid created requires 2,16 MB, therefore, make sure that your hard drive has enough free space. If you want to save a temporary grid as a new input for the model, you may want to save it in the DataUser folder. After a few simulations, this folder will contain a lot of temporary files, which occupy a lot of megabytes and should be deleted on a regular basis. However, do not clean the folder manually but use the “File/Manage Data Sources” menu, available when the active window in ArcView is a view.

└ SimulationOutputs

Temporary files are created and processed during a simulation. This folder contains the ones in which the user is likely to be the most interested in:

- the runoff table: named “Runoff[2 digits].dbf”, this Vtab contains the area of each time zone (columns) and the amount of runoff generated over them at each time step (rows). It is the Vtab used for cross-summing.
- the discharge table: named “Discha[2 digits].dbf”, this Vtab contains the discharge values (result of the cross-summing) at each time step of the simulation.

However, an OutputView is also created and displays temporary Gthemes in which the user might be interested in:

- the accumulated height of rainwater intercepted by vegetation (in mm),
- the accumulated height of rainwater infiltrated in the soil (in mm),
- the location and delineation of the time zone areas,
- the vegetation interception ratio. It is the percentage of rainwater absorbed, returned to the atmosphere by the vegetation cover,
- the speed at which overland water flows across a cell,
- the soil imperviousness,
- the soil water storage capacity, the height (expressed in mm) of rainwater, which can be absorbed by the soil. The name of the theme contains the water table and the moisture mode names,
- the soil infiltration rate value (in cm/h). The name of the theme contains the water table and the infiltration rate mode names,

Those temporary grid folders are stored in the Workspace folder where they remain until the user deletes them (see Workspace folder description).

Resources

This folder contains different subfolders with a backup of the default land use data files and tables. There is a zipped file containing a backup of the originaldefault data, some scripts (they can be loaded as such in ArcView) I wrote to achieve some intermediary tasks, such as a pseudo-DEM from a stream network grid, a flow direction from an outlet grid and a stream network grid and more.

To run your first simulation

Located at the root, UHModel.apr is the application, which contains all the scripts of the model. If ArcView is installed on the user's computer, double click on this file to launch the model. ArcView starts and loads its GIS environment. If the user has no other extensions loaded by default, UHModel.apr creates and displays an *InputView*.

The *InputView* is the view, which contains all the themes required to start a simulation. In this case, the loaded themes are the files stored in the DataDefault folder. The default information and lookup tables are loaded too. All those input elements are stored in the application as the preference dictionary (stored in a global variable called “_UHMDictionary”). When the *InputView* or another view is displayed as the active window, a set of 6 buttons is displayed in the button bar of the application:



The ***DefaultInputView*** button. When this button is pressed:

- it automatically creates and displays an input view with the default themes stored under the DefaultUser folder if there is no other view available. The new *InputView* is stored in the preference dictionary.
- if there are other views created in the application, you can either make one of the view the *InputView*, or create a new one with the default Themes. In this case, the new *InputView* is named “*InputView*” and any other view named “*InputView*” is renamed “*InputViewBackup*”. The new *InputView* is stored in the preference dictionary.



The ***DefaultTable*** button. When this button is pressed:

- it loads the soil and horizon information tables and the lookup tables in UHModel and stores them in the preference dictionary.



When the user clicks on the third button, it displays a vertical list of two buttons dealing with rain series properties, which are set and stored in the preference dictionary:



- the first button (***RainSeries***) allows you to load a table stored as a dbf file, which contains the amount of precipitation per time step. There has to be a specific field (a column) of the table, which stores the height of precipitation in millimetres. If the name of the field is “Rain_mm”, the field is automatically identified and stored in the preference dictionary. Otherwise a list of the fields pops up and the user has to double-click on the proper name to designate the correct field.

- the second button (*TimeStepDuration*) is only available when a rain series table has been successfully loaded. It allows the user to enter the time step duration in minutes (the duration cannot be less than 5 minutes, otherwise too many time zone areas will be drawn and it might overflow the memory).



The fourth button displays a list of soil property modes, which are available only when the soil and horizon information tables and the soil polygon shapefile have been loaded. Modes are set and stored in the preference dictionary.



- when the *WaterTableDepthMode* button is pressed, a menu pops up and displays a list of five water table depth modes: **Mean** (the mean depth and the depth standard deviation values for each soil type are stored in the soil information table), **High** (mean-standard deviation), **Low** (mean+standard deviation), **Saturation** (the water table has reached the soil level), **Custom...** (the user can enter the depth in centimetres). This setting will only affect soils where a water table is present.

- when the *InfiltrationRateMode* button is pressed, a menu pops up and displays a list of three infiltration rate modes (in order to select one infiltration rate value in the list of horizon series infiltration rate values): Highest (fastest infiltration rate), **Lowest** and **Median**.

- when the *SoilMoistureMode* button is pressed, a menu pops up and displays a list of four soil initial moisture modes: **0%** (soils are totally dry), **100%** (soils are totally saturated, they cannot absorb any rainwater), **User's percentage...**, **Lookup table...** The *lookup table* mode displays the list of tables loaded in ArcView and the user selects the proper one.



The *Show/SetPreferences* button shows a list of the preferences stored in the preference dictionary. Double clicking on one of them allows the user to modify them. The following preferences have to be correctly set in order to launch a simulation:

- the InputView view (which should contain all the necessary Fthemes and Gthemes)
- the outlet point FTheme
- the stream network GTheme
- the road polyline FTheme
- the urban polygon FTheme
- the bare ground polygon FTheme
- the crop polygon FTheme
- the vegetation polygon FTheme
- the flow direction GTheme
- the DEM GTheme
- the soil polygon FTheme
- the soil information table
- the horizon information table
- the soil texture lookup table
- the vegetation lookup table
- the urban lookup table
- the rain series table
- the rain amount field of the rain series table
- the time step duration
- the infiltration mode

- the water table depth mode
- the soil moisture mode (and the soil moisture lookup table if necessary)



The **StartSimulation** button is only available when all the preferences are correctly set. If the button is not available, check the preferences with the *Show/SetPreferences* button. As a warning, a sound is played when the simulation is over.

Though runoff and discharge tables can be opened with other programs than ArcView (Excel, Access or any spreadsheet program supporting dbf format), the user may want to save the state of the simulation and be able to reload them in the UHModel application. To save your simulation as a usual ArcView project click on the *Save* icon or File/SaveProject, or use the “Ctrl+S” shortcut key combination.

In order to preserve the structure of the model, the user cannot save his simulations under “UHModel.apr”. Moreover, all simulations must be saved at the root so the Workspace folder is at the same location.

How to calibrate the model?

Most of the parameter values used by the model (imperviousness, infiltration rates, water storage capacities, V45 speeds, vegetation interception...) are accessible and can be modified in the four lookup tables used in the model (cf. **appendix 4**). Refer to the thesis for parameters meaning and effect

References

Literature

- Bell F.C. and Vorst P.C. (1981) Geomorphic Parameters of Representative Basins and their Hydrologic Significance. *Australian Water Resources Council, Australian Government Publishing Service*
- Beven K.J. (2001) Rainfall – Runoff Modelling, the Primer. *John Wiley & Sons, Ltd*
- Beven K.J. and Moore I.D. (edited by) (1993) Terrain Analysis and Distributed Modelling in Hydrology. *John Wiley & Sons, Ltd*
- Brooks K.N., Ffolliott P.F., Gregersen H.M. and Thames J.L. (1991) Hydrology and the Management of Watersheds. *Iowa State University Press/AMES*
- Cudennec C. (2002) Eléments d'hydrologie. *Laboratoire PSNGH de l'ENSAR*
- Kraijenhoff D.A. and Moll J.R. (1986) River Flow Modelling and Forecasting. *D.Reidel Publishing Company*
- Lazaro T. (1979) Urban hydrology, a Multidisciplinary Perspective. *Ann Harbor Science Publishers Inc*
- Morgan R.P.C. and Rickson R.J. (1995) Slope Stabilization and Erosion Control, a Bioengineering Approach. *E&FN SPON*
- Razavi A.H. (1999) ArcView GIS/Avenue Developer's Guide, Third Edition. *OnWord Press*
- Razavi A.H. and Warwick V. (2000) ArcView GIS/Avenue Programmer's Reference, Third Edition. *OnWord Press*
- Stephenson D. and Meadows M.E. (1986) Kinematic Hydrology and Modelling. *Elsevier Canada*
- Soil Survey Committee, Subcommittee on Soil Classification (1978) The Canadian System of Soil Classification. *Canadian Department of Agriculture, publication 1646*

On-line references

- Arocena J.M. and Abley M. Soils of Canada webpage of the University of Northern British Columbia Website
<http://quarles.unbc.edu/nres/soc/index.html>
- Ball J. (2001) Soil and Water Relationships. *The Samuel Roberts Noble Foundation website, Agricultural News, September 2001*
<http://www.noble.org/ag/Soils/SoilWaterRelationships/Index.htm>
- BC Government Agricultural Land Reserve Homepage
http://apps.compasscanada.com/lrc/ALR/ALR_Home_Page.stm
- ESRI user community page of the official ESRI website
<http://arcscripsts.esri.com/>
- Purdue University (US Indiana) webpage about the SCS curve number method
<http://www.ecn.purdue.edu/runoff/documentation/scs.htm>
- Ministry of Sustainable Resource Management, Guide to Spatial Land and Resource Information in LRMP webpage from the Government of British Columbia official website
<http://www.luco.gov.bc.ca/lrmp/spatial/agri.htm>
- Appendix F of the official website of the City of Tallahassee (Florida)
<http://talgov.com/citytlh/growth/manual/appendf.htm>
- Annexe 2 Infiltration rate and infiltration test webpage of the Food and Agriculture Organization of the United Nations website
<http://www.fao.org/docrep/S8684E/s8684e0a.htm>