Master Thesis

TVVR 17/5002

A dynamic and spatially distributed rainfall runoff model

- Developing a model for overland flow in GIS, based on a multiple flow direction algorithm

Hampus Nilsson

Division of Water Resources Engineering
Department of Building and Environmental Technology
Lund University

A dynamic and spatially distributed rainfall runoff model

- Developing a model for overland flow in GIS, based on a multiple flow direction algorithm

By:
Hampus Nilsson

Master Thesis

Division of Water Resources Engineering
Department of Building & Environmental Technology
Lund University
Box 118
221 00 Lund, Sweden

Water Resources Engineering
TVVR-17/5002
ISSN 1101-9824

Lund 2017
www.tvrl.lth.se

Master Thesis
Division of Water Resources Engineering
Department of Building & Environmental Technology
Lund University

Swedish title:	En dynamisk och distribuerad modell för översvämningsmodellering
English title:	A dynamic and spatially distributed rainfall runoff model - Developing a model for overland flow in GIS, based on a multiple flow direction algorithm
Author:	Hampus Nilsson
Supervisors:	Rolf Larsson Andreas Persson at Department of Physical Geography and Ecosystem Sciences
Assistant supervisor	Gunnar Svensson at Tyréns AB
Examiner:	Ronny Berndtsson
Language:	English
Year:	2017
Keywords:	flood modelling, GIS, TFM, surface runoff, multiple flow, flow routing

Acknowledgements

During the initiation of this work the aim of reaching all objectives seemed very ambitious for a master thesis. However, with the help from many people I was lucky to perform and finish some very interesting work.

Foremost I want to thank my supervisor, Andreas Persson, for encouraging me to setup a dynamic model, sharing your hydrological experience and always taking your time to help out. Also a big thank you to Rolf Larsson and Gunnar Svensson for helping me find my way into hydrological modelling and this thesis subject. I also want to give a special thank you to Petter Pilesjö at GIS Center for good ideas, inspiration and fast feedback. The GIS center's research on multiple flow direction algorithms, and the TFM code that was shared with me, has made my work possible. Additionally, I want to thank the rest of the staff and PhD students at GIS Center for offering a desk, access to the coffee machine and inviting me to be part of their group this semester. Gea Hallen is worth a thanks for sharing data on Gästgivarevägen from her master thesis work. And at last my roommate Alexander Betsholtz is worth a thanks for putting up with dinner discussions about how to route flowing water and how nice it is to design your own modelling program.

Abstract

Climate change is expected to cause 20-25% heavier rainfall in Sweden which, in combination with the ongoing urbanization, increases the probability of flooding in urban areas. These floods may cause large infrastructural damage and economical costs. Therefore, adequate models for overland flow are needed to identify high-risk areas where mitigating actions should be performed.

In this thesis a new model of intermediate complexity is developed. It is the first known dynamic model which uses the triangular form-based multiple flow algorithm by Pilesjö and Hasan (2014). The model only includes some basic hydrological processes and is therefore less complex than the fully physically based models. Meanwhile, the model still resembles the nature of flow to a larger extent than simpler static models. It includes temporally dynamic and spatially distributed precipitation as well as spatially distributed infiltration and surface roughness. The model is implemented as Matlab code which derives results at different modelling times. By using a geographical information system, the results are visualized as maps of water depths at different stages during and after a precipitation event. Modelling results for Swedish urban and rural conditions are presented together with some initial investigations on how resolution, area and runtime are related.

Sammanfattning

Klimatförändringen förväntas orsaka 20-25% kraftigare regnoväder i Sverige vilket, tillsammans med urbaniseringen, ökar sannolikheten för översvämningar i urbana områden. En översvämningen kan innebära stora materiella skador och kostnader för samhället. För att förebygga dessa behövs mer kunskap om vilka områden i staden som är mest utsatta. Information om var och när stora vattenmängder väntas kan uppskattas med hjälp av hydrologiska modeller.

I det här arbetet har en ny dynamisk modell tagits fram. Till skillnad från de vanligt förekommande statiska modellerna beskriver den här modellen även hur vattendjup förändras över tid. Samtidigt är avsikten att modellen ska vara mindre komplex än de modeller som har en fullständigt fysikalisk beskrivning av hydrologin. Modellen är den första dynamiska modellen som använder en flödesalgoritm, kallad TFM (Pilesjö and Hasan, 2014), för att beräkna hur vatten fördelas vid varje tidssteg. Modellen använder höjddata i rasterformat, hanterar nederbörd med temporal och rumslig variation samt inkluderar spatialt distribuerad infiltration och markfriktion. Modellen har implementerats i Matlab och resultaten har visualiserat med geografiska informations system i form av kartor över vattendjup vid olika tidpunkter. Modellens uppbyggnad beskrivs matematiskt och presenteras tillsammans med resultat för svenska urbana och rurala testområden. Förhållanden mellan upplösning, områdesarea och beräkningstid har också undersökts.

Abbreviations

D8	Deterministic eight-node algorithm.
DEM	Digital elevation models.
EFR	Elevation based flow routing.
FD	Flow distribution.
GIS	Geographical information system.
LiDAR	Light Detection and Ranging.
MFD	Multiple flow direction.
PHB	Physically based.
TFM	Triangular form-based multiple flow algorithm.
TFN	Triangular facet network.
VPI	Volume-precipitation-infiltration.

Contents

1 Introduction

1.1 Preparing for heavy rainfalls in a changing climate

During the last decades, humanity's knowledge about the ongoing climate change has increased with information such as the IPCC reports. New regional projections on both temperatures and extreme weather have been made available with every report (IPCC, 2014). In Sweden these projections estimate that heavy rainfalls, with day long durations, will increase in intensity by 20% and shorter rainfalls, under an hour, get a 25% intensity increase by the end of this century (Olsson and Foster, 2013). At the same time the demographic trend in Sweden, as well as for the major part of the world, shows that more and more people are living in cities (Svanström, 2015). The growing urban population requires housing and roads that are replacing unexploited, permeable ground with roofs and asphalt. Roofs and asphalt are normally impervious surfaces which cause overland flow that turn into storm water. If the predictions of increased rainfall intensities are correct then these extreme precipitation events will become recurrent challenges for the city planners, decision makers and the urban population.

Recent examples of heavy rainfall and urban flooding, such as the events in Copenhagen 2011 and Malmö 2014, are having direct effects on people's lives and also getting public attention through news media (Burström et al., 2014, Eriksson, 2015). The Malmö event is estimated to cost the municipality at least 100 million SEK and the insurance claims from citizens are even bigger (Lindher, 2015). People living in urban areas are affected and now much more aware of the problem. The decision makers are looking for more knowledge on where, geographically, to take action for mitigating the effects of extreme rainfalls. Thus, effective and adequate overland flow models have an important role to play when estimating possible future floodings.

1.1.1 Modelling overland flow to prevent future flooding

For city planners to get to know which areas are at risk of flooding, more knowledge is needed about how spatially varying surface conditions correlate to overland flow in urban environments. Geographical information systems (GIS) are essentially a collection of models and software that can be used as tools for storing, analyzing and visualizing geographical data in general. It can also handle spatially varying hydrological properties as input data and analysis results. GIS and geographical information science have set the base for developing geographically distributed models. Also having digital elevation models (DEMs), describing landscape topography, makes a good setup for modelling spatially distributed hydrological phenomena, such as overland flow.

The DEMs are most often described as a grid, i.e. a matrix, called raster in the GIS applications. The DEM raster contains the surface elevation for each cell center. A landscape topography described by the DEM can be used as input to model overland flow. The DEMs elevation information is crucial since the main acting force is gravitation, pulling water ever towards lower elevations. Hydrological modelling in GIS environments usually have a focus on detecting possible water filled depressions and runoff paths.

Throughout Sweden the investigations done for municipalities are using models of varying complexity when estimating surface runoff. Many of the models in GIS's have flow routing algorithms based only on elevation and no other hydrological aspects when routing flow from one cell to the next. They can map possible overland flow paths in high resolutions in a relatively short runtime. These models are referred to as elevation based flow routing (EFR) models.

Physically based hydrological (PHB) models which describe the physical processes in the hydrologic system in great detail, i.e. more factors than just the surface elevation are incorporated. These models are normally dynamic and can produce results on e.g. catchment discharge, water tables and surface water depths with a temporal resolution (Hingray et al., 2014). Mike 21 is an example of a model commonly used in Sweden. It has a distributed and, at least partially, physically based model which incorporate factors such as surface roughness, infiltration, mass and momentum equations, precipitation, evaporation, inlets and outlets (DHI, 2015). Each of these factors are represented by equations containing many parameters which need to be correctly set before modelling.

Many of the overland flow investigations are done on a municipal level. When an investigation is initiated there is a choice to be made by either using the PHB models, with many parameters and a demanding setup, or the EFR models with fewer parameters. In reality that is modelling with a temporal dimension and many more hydrological processes (PHB models) or modelling only with respect to elevation, without the dynamics of a time dimension, and without estimates on water depths (EFR models).

The PHB model investigations demands an experienced specialist and are therefore most often done by consultants, take more time and are costlier than an EFR model investigation. Since the EFR models also require less data, fewer parameters and less runtime, than the PHB models, they require less hydrological knowledge and less experience to run and achieve a result from. These differences may be crucial when a model choice is made in a municipality. As an example a publicly accessible manual directed towards municipalities advocates for using an EFR model setup and also guides the reader step by step through the setup (Sylvén and Ekelund, 2015).

The modelling differences, between EFR and PHB models, in demand of hydrological knowledge, experience, time and setup complexity yields differences in what

kind of results are possible. When making a model choice the investigation scenarios and what kind of estimates that are of interest should be a major consideration.

A major factor which can affect the result when modelling urban areas is the DEM resolution. Different resolutions are used in different applications but for urban flood forecasting 1-5 meter grid size is often recommended (Henonin et al., 2013). In Sweden 2x2 m DEMs are easily accessible since it is the new standard DEM delivered from Lantmäteriet (Lantmäteriet, n.d.). Even so 1x1 m DEMs can in many cases be generated from Lantmäteriet or from municipal Light Detection and Ranging (LiDAR) raw data and provide more detailed maps.

Even though 1x1 m and 2x2 m DEMs are available the model running in Mike 21 isn't efficient for larger areas (whole cities) and high resolutions. The calculations needed makes the program too time consuming for a lot of investigations. A doubled resolution is estimated to generate a factor ten run time. An example, calculating the whole city of Landskrona (approximately 25-30 km^2) in 2x2 m, is estimated to take a week or more depending on the computer (Mårtensson and Gustafsson, 2014). It is also common that a higher resolution is more likely to be unstable, require a more extensive parameter calibration and need much more work for the model to run (Svensson, 2016, September 6, personal communication).

Due to the mentioned run time issues in Mike 21 it is instead often run in 4x4 m resolution when urban investigations of overland flow are done commercially (Svensson, 2016, September 8, personal communication). A coarser resolution like 4x4 m might be less likely to accurately describe important details in the urban environment. Examples of such are street curvature, curbstones and ditches that will have effect on the runoff paths.

1.1.2 Motivation for developing a new model

The need for developing a new model is justified by the thought that some extra data and a few extra settings, additional to those in the elevation based (EFR) models, can be collected and handled by many investigators. This in contrast to a physically based (PHB) model, that demands much more experience, more data and requires a setup with many more parameters. One of the reasons that the static EFR models have been used to a great extent is that elevation is the easiest available and used data. Elevation is treated in DEMs that are easy to handle computationally since they are continuous, easy to store and easy to combine with other data. The availability is due to Lantmäteriet and the Swedish Municipalities collecting LiDAR data to a much greater extent than e.g. collecting samples of soil characteristics or classifying surfaces depending on their friction. This thesis is an attempt at developing a new model and getting further than just incorporating elevation, without setting up a model as complex as todays PHB models.

1.2 Problem formulation

The work performed in this thesis intend to meet the demand for more knowledge on overland flow regarding the following questions:

- How can a new dynamic rainfall runoff model, which incorporates an adequate elevation based flow routing model, be developed?
- How can such a model incorporate hydrological processes?

The EFR models operate under the condition that the ground is impermeable and the gravitational pull is the only major force to consider. Surface runoff over an impermeable ground is the only hydrological process included, no temporal dimension is modelled, which implies that all surface runoff is modelled to flow all the way downstream. This thesis starts from the basics of the EFR models and investigates the possibility to add other hydrological processes and dynamics.

1.3 Aims and objectives

The aim of this thesis is to setup a new model that is somewhere in the complexity spectrum between the EFR and PHB models. It should handle more input data than just elevation, although it should not require a setup as complex as the PHB models. The new model should reflect on dynamic changes in the runoff process, i.e. estimate surface water depths and their spatial and temporal distribution. It should be able to operate in high resolutions. The intention is to start developing the new model with inspiration from the EFR models.

The main objectives are to introduce (1) a time dimension to model the dynamics and (2) add new hydrological processes/aspects. The hydrological processes/aspects to incorporate are:

- Temporally heterogeneous precipitation.
- Spatially heterogeneous ground infiltration that is constant over time.
- Spatially distributed surface roughness.

Additional objectives (time permitting) are (3) to introduce storm sewer inlets and outlets in the model and (4) to investigate possibilities for the model to operate reasonably fast in high resolution in both urban and rural environments.

1.4 Limitations and delimitations

This thesis is delimited to the development of a new model for estimating overland flow. Any time a new working model is developed it is advisable to validate it using earlier

recorded data or against collected data, before using it in research. In this case, data on precipitation and water depths with a temporal resolution would be the adequate reference. However, this is beyond the scope of this thesis due to limitations in time and economic resources.

Another delimitation with this work is that focus is on developing a new model that only incorporates some of the physical processes and aspects in the overland flow phenomena. No full review of all known hydrological processes and their physical properties is done. Rather an assumption is made that the hydrological processes and aspects chosen as objectives are relevant enough to describe the dynamics of overland flow due to heavy rainfall.

The developed model is limited to using only raster based input data regarding elevation (DEMs), infiltration, precipitation and surface friction. All raster data needs to be stored in the same resolution and cover exactly the same area.

2 Background and theory

The first section in this chapter gives a short review on different types of rainfall runoff models and how a digital representation of topography is constructed. The following sections gives a more comprehensive background to flow algorithms and can be read either in detail or from a more general perspective depending on the reader's interest and knowledge.

2.1.1 Main characteristics of hydrological models

In hydrological modelling mainly two different approaches for spatial representation exists, the spatially distributed and the lumped models. The so far mentioned models (EFR and PHB) are spatially distributed, using a raster structure, built up of equally sized and regularly gridded cells, which are the spatial components. A lumped model, on the other hand, consist of some kind of spatial averaging where whole catchments or sub-catchments are treated as the spatial components (Fletcher et al., 2013). A distributed model can describe urban details at raster resolution level and generate distributed results on water depths that are crucial for the possibility to evaluate effects of a possible future flooding.

In regard to the description of hydrological processes either a physically based approach, a conceptually based approach or a combination of the two is chosen when a hydrological model is constructed. All of these approaches typically consists of different sub-models that are dedicated to describe certain hydrological processes, e.g. infiltration. The sub-models are coupled together to describe a hydrological system. What distinguishes the physically based models is that their parameters are chosen based on physical laws, a parameter can be e.g. a measurable soil property such as hydraulic conductivity. For conceptual models on the other hand they are designed to represent the hydrological system without a parametrization of the physical laws. Instead they are based on the hydrologist's perception of the sub-models, how they function and how they interact with each other. Its common that conceptual models have between 5 and 15 parameters whereas physically based models often have several dozen (Hingray et al., 2014).

2.1.2 Water balance in surface runoff

When modelling surface runoff there are a lot of hydrological and hydraulic processes to take into consideration. For a physically based representation to be used in an urban setting, it could be expected to include surface roughness, infiltration, mass and momentum equations, precipitation, evaporation, storm sewer inlets and outlets. In contrast, attempts at estimating drainage with the static EFR models are done by only taking topography into account. This approach is mainly based on the principle that water

always flows downstream in a landscape. One purpose for this procedure can be to map the possible runoff paths that excessive water theoretically can take during a heavy rainfall. That could serve to give an initial and somewhat rough idea of which areas might be at risk of flooding. However, this kind of modelling (EFR models) won't provide any information on which water depths are likely, nor do they estimate where, when or what maximum water depths could emerge.

2.1.3 Digital representations of topography

When building a model in a digital setting there are different ways of representing the topography. Two major ways of creating a digital representation stands out. The first one having the landscape represented as a DEM raster (matrix), containing regularly spaced cells, where each cell value represent the ground elevation in the cell center. The second representation being vector based e.g. isolines with constant elevation and triangular irregular networks (TIN) with facets, irregularly formed from point data. Yet another vector based example are the LiDAR point clouds which contain points with x, y and z data (Harrie, 2013). This thesis will mainly investigate the use of DEM rasters as input in the modelling.

2.1.4 Filling depressions and sinks

DEMs often come with depressions, here defined as areas without flow paths leading out of the them. The smallest depression is an individual cell that is surrounded by elevated cells, it is from here on referred to as a sink. Both sinks and bigger depressions in a DEM can originate from digital artefacts or real depressions. The real depressions can be natural or manmade features in the landscape, while the artefacts can originate from data sampling, incorrect interpolation or low resolution (Lindsay and Creed, 2005, O'Callaghan and Mark, 1984). In all cases depressions hinder the flow routing from continuing downhill through the DEM. In the event of an extreme rainfall the water could potentially fill the depression, continue out from its lowest boundary point and onwards downhill, flooding lower areas. This is often the reason for modifying the DEM by pre-filling depressions before running the EFR model. When fill is used the method finds all cells within a depression and raises their elevation to the boundary elevation. For all depressions that are filled, a flat area will be created. Often the DEM is prepared by filling all existing depressions and sinks before it is used in the EFR models. Optionally the fill algorithm can run so that it only fills depressions meeting a certain criterion. This could be e.g. only filling depressions that have a specified depth, volume or surface area as illustrated in Figure 1.

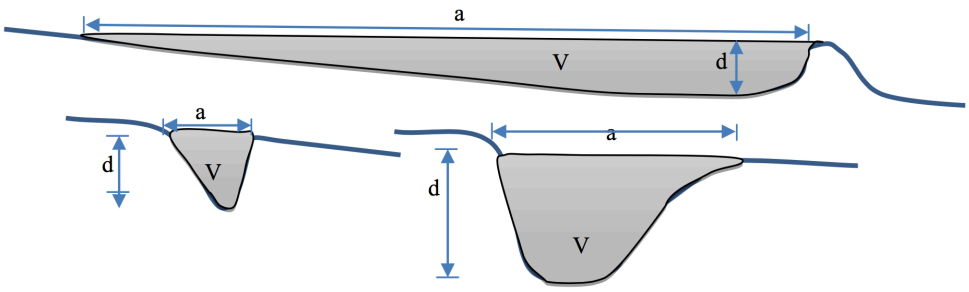

Figure 1. Examples of depressions with varying depth, volume and surface area (Hasan et al., 2012a).

2.2 Single cell flow routing - the steepest neighbor algorithm

2.2.1 D8 - the deterministic eight-node algorithm

One way of doing grid based water routing is to route the water in a cell to one of its lower neighbors. The way of deciding which neighbor to choose can differ slightly between algorithms but a common scheme is the fairly intuitive one of routing all water in the steepest downward sloping direction. This has often been called the D8 algorithm, meaning the deterministic eight-node algorithm referring to routing all water to one of eight neighbors.

2.2.2 D8 - flow directions

The flow direction computation is the heart of what makes the D8 algorithm what it is; fast and fairly easy to understand. For each cell in a DEM raster (matrix) a corresponding flow direction will be calculated. For a DEM matrix consisting of r rows and c columns a corresponding r by c D8 flow distribution matrix (FD_{D8}) is created as matrices containing all flow directions in the area. Each cell in FD_{D8} will be assigned a flow direction towards the neighboring cell with the steepest drop. Meaning the biggest elevation difference divided by the distance (which is a $\sqrt{2}$ factor greater between diagonal neighbors). An example of elevations and the corresponding D8 flow directions is shown Figure 2.

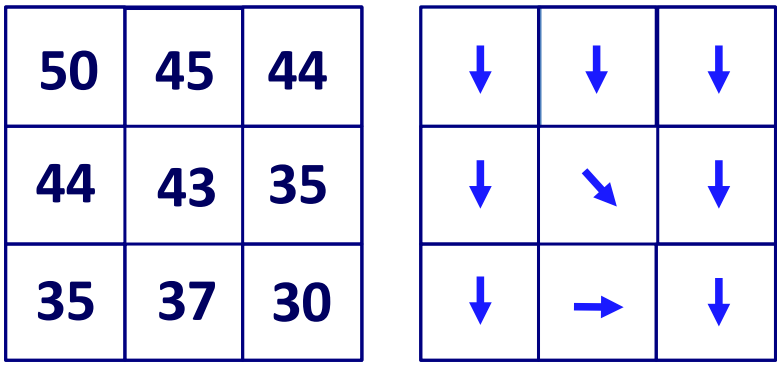

Figure 2. Elevations in a 3-by-3 matrix with corresponding flow directions estimated by the D8 algorithm.

The flow directions are encoded with values equaling 2 to the power of 0-7, i.e. 1, 2, 4, 8, 16, 32, 64 and 128, for the eight possible directions as shown in Figure 3. This coding should be interpreted as for a focus cell, having its flow direction set to 32, the steepest drop direction is to the left and hence the single flow direction is left.

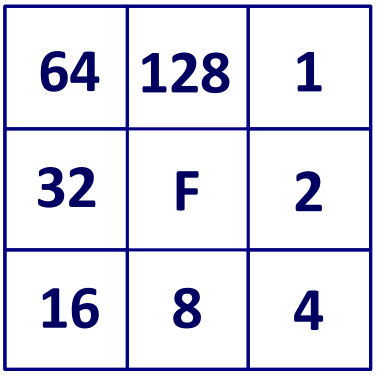

Figure 3. Encoding of neighboring cells used for the flow directions in the D8 algorithm with F indicating the focus cell.

The flow direction algorithm is often to be used on pre-prepared DEMs, that have had their depressions filled. Therefore, it must handle flat areas. The handling of flat areas is defined so that flow directions are assigned for all flat cells. The algorithm goes through the depression starting from the boundary cells that have a flow direction leading out of the flat area. Then, iteratively, each neighbor cell with undefined flow direction is given a flow direction towards the boundary cell. Details on the flow direction estimation are found in **Box 1** (Jenson and Domingue, 1988). From here on it is said to be defined through the function in equation (1).

$$FD_{D8}(r,c) \coloneqq f_{FD_{D8}}\big(z(r+i,c+j)\big) \quad (1)$$

Where: $\boldsymbol{FD_{D8}(r,c)}$ defines the flow distribution in row *r*, column *c* as estimated by the D8 algorithm for all cells inside the border.
$\boldsymbol{i,j}$ indicates that FD_{D8} is a function of the focus cells eight neighboring cells, i.e. $i,j \in [-1,1]$.

Box 1. Details for the D8 flow direction algorithm.

The following steps are executed, in consecutive order, and defines the flow directions function $f_{FD_{D8}}$ in (1):
1. All the cells along the edge are assigned a flow direction out of the raster under the assumption that the area of interest is located inside the raster border.
2. For all other cells, one at a time, the distance weighted elevation drop is calculated to their eight surrounding cells. The distance weighted elevation drop being the difference in elevation divided by the distance. The diagonally adjacent cells having a distance factor of $\sqrt{2}$ and the others 1.
3. Determine the largest drop in elevation by:
3a. If the largest drop is less than zero, assign a negative value to indicate that it's a sink and the flow direction is undefined.
3b. If the largest drop is zero or positive, and bigger than all other drops, assign the flow direction towards that neighbor using the encoding shown in Figure 3.
3c. If the largest drop is zero and occurs in more than one direction, it is a flat area. Then a temporary encoding of the locations of all the lowest neighbors is saved as the sum of the codes for these lowest neighboring cells. The intrinsic reason for the chosen encoding of directions as 1, 2, 4, 8, 16, 32, 64 and 128 is that every sum of codes will be unique and possible to decipher. If the cell is "inside" a flat area e.g. the sum will be 255.
4. For each cell that has many lowest neighbors, every one of its lowest neighbors will be checked if they have a single flow direction encoded. For the first neighbor encountered with a single flow direction encoded and not flowing into the focus cell, this will be the direction of flow assigned to the focus cell.
5. Step 4 is repeated until no more cells can be assigned a flow direction.
6. Now for all cells left with a value not equal to 1, 2, 4, 8, 16, 32, 64 or 128 their codes will be set to a negative value indicating a depression (not happening when fill all has been used).

The basics of this procedure was developed in 1984 (O'Callaghan and Mark, 1984) and then refined to the method described in Box 1 in 1988 (Jenson and Domingue, 1988). The general ideas of the D8 workflow described above are still used in today's GIS software, e.g. in ArcGIS's *Flow direction* function (ESRI, n.d.).

2.2.3 D8 - drainage area estimation

For each cell in the area the drainage area is to be calculated and stored in the drainage area matrix (DA_{D8}). To begin with the DA_{D8} is filled with zeros meaning no cell is assigned its own area. This is done step by step and starts at the most upstream cells, those which have no other cells draining to them. Going from these peak cells along their flow directions in FD_{D8} the next downstream cell will get the area from the upstream cell added to its current value in DA_{D8}. Every time accumulated area is passed on downstream it

consists of the drainage area value of the upstream cells plus the draining cells own cell area. In many cases cells will get accumulated drainage added from many neighboring cells. This goes on in an order built up so that the cell in focus gets the accumulated drainage area added from all neighboring cells that have a flow direction pointing towards it. Then, not earlier, the cell in focus is ready to pass its accumulated drainage area to the neighbor in its steepest sloping direction. When all cells in the raster have passed on their drainage area then the DA_{D8} matrix is complete (O'Callaghan and Mark, 1984, Jenson and Domingue, 1988). As an example the DEM and its flow direction in Figure 2 would generate a DA matrix as illustrated in Figure 4.

0	0	0
1	1	1
2	0	5

Figure 4. An illustration of the accumulated drainage area estimated by the D8 algorithm for the DEM and flow directions shown in Figure 2. Each cell in the DEM is considered to have an area equal to 1. The drainage area of each cell is the sum of all the incoming drainage area.

2.3 Multiple flow – routing water to many neighbors

Another grid based way of routing water is using a multiple flow algorithm. The multiple flow meaning that the water routed from one cell can be divided between many of its neighboring cells depending on the slopes and the elevation setting.

As an example one can think about an east to west stretching ridge represented in a DEM. Let the crest, the highest elevation along the ridge, be represented by a row of cells with their centers exactly on the crest. The D8 single flow algorithm would here route all water falling on a ridge cell to either north or south (for details see step 4 in Box 1). The possibility to instead divide the flow from each cell to be 50 percent north and 50 percent south, for a more accurate representation of reality, becomes appealing. Especially for convex landscapes, such as ridges, the need for better (diverging) routing was shown already in 1991 (Freeman, 1991).

The multiple flow direction (MFD) algorithms can generate flow divergence, meaning that water flow can be routed to more than one of the neighboring downhill cells. Every

MFD algorithm implements its own way of deciding how this division should be done, hence the complexity and the computation time differs among the algorithms.

2.4 The triangular form-based multiple flow algorithm (TFM)

The multiple flow direction algorithm TFM has been developed for calculating divergent flow directions, realistic drainage area and still being reasonably fast in comparison to other MFD algorithms (Pilesjö and Hasan, 2014). It can be seen as an improvement on the triangular facet network (TFN) algorithm. The TFN algorithm is partially vector based and has been shown to estimate flow over theoretical surfaces with accurate results as well as flow distribution over real surfaces without noticeable artefacts. The TFN was also proposed as an option for dynamical modelling (Zhou et al., 2011). The TFM on the other hand is fully raster based and outperforms the TFN and also many MFDs, in respect to estimating flow over theoretical surfaces of convex form, concave form, flat areas, as well as combinations of those (Pilesjö and Hasan, 2014). Raster based flow algorithms are expected to be less computationally demanding than the vector based algorithms. This together with the TFMs accurate flow estimation motivates choosing it for estimating flow in a new model. The following TFM background is referring to the work done by Pilesjö and Hasan (2014).

2.4.1 TFM - water movement within the cell

The model is built on the raster grid with the theory that every cell can be assigned flow packages of water which will spread downhill in the landscape. TFM does raster calculations based on the elevation in each cell but it also refines the detail by dividing every cell into eight triangles, called facets. Each facets area is 1/8 of the cell area. This is done by the division of the cell horizontally, vertically and diagonally as seen in Figure 5.

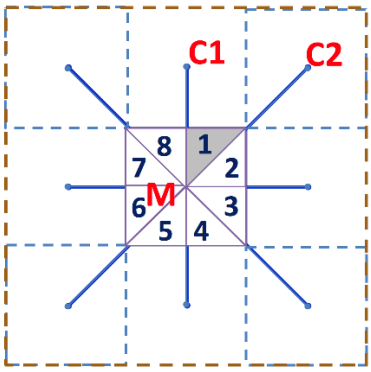

Figure 5. Each cell is divided into eight facets where each facets is a plane formed by three cell centers and their elevation. Facet 1 is formed by the focus cells center point (M), the cell center point above (C1) and the diagonal cell center point (C2). (Pilesjö and Hasan, 2014)

Each facet represents a first order trend surface, a triangular plane, with constant slope and aspect (slope direction). For each facet, these can be calculated from the elevation in the focus cell and the elevation in two neighboring cells. In Figure 5 these are denoted as $C_1(x_1,y_1,z_1)$, $C_2(x_2,y_2,z_2)$ and $M(x_3,x_3,z_3)$. Together they form the plane

$$z = f(x,y) = ax + by + c \tag{2}$$

where a, b and c can be derived as:

$$\begin{cases} a = \dfrac{(y_1 - y_3)(z_1 - z_2) - (y_1 - y_2)(z_1 - z_3)}{(x_1 - x_2)(y_1 - y_3) - (x_1 - x_3)(y_1 - y_2)} \\ b = \dfrac{(x_1 - x_2)(z_1 - z_3) - (x_1 - x_3)(z_1 - z_2)}{(x_1 - x_2)(y_1 - y_3) - (x_1 - x_3)(y_1 - y_2)} \\ c = z_1 - ax_1 - by_1 \end{cases} \tag{3}$$

If p and q are set to be the gradients in the W-E and N-S directions respectively, then, given equation (2), the gradients, in x and y direction, for a facet will be:

$$p = f_x = \frac{\partial f}{\partial x} = a \quad \text{, and} \quad p = f_y = \frac{\partial f}{\partial y} = b \tag{4}$$

The slope (β) and aspect (α) of the **facet** can then be derived as:

$$\beta = \arctan\sqrt{p^2 + q^2} = \arctan\sqrt{a^2 + b^2} \tag{5}$$

$$\alpha = 180° - \arctan\frac{q}{p} + 90°\frac{p}{|p|} = 180° - \arctan\frac{b}{a} + 90°\frac{a}{|a|} \tag{6}$$

Before the **routing** of water, from one cell to its neighbors, can be estimated, a cell internal water **movement** process is done. At the start of this process each of the eight facets gets the same amount of water. The water put in a facet can be either (1) coded to stay in the facet (for external routing later on), (2) moved within the focus cell to another facet or (3-4) split between facets and/or neighboring cells. For each facet, its water routing is hence coded to (1) ***stay*** temporarily in this facet, (2) ***move*** to an adjacent facet, (3) ***split*** between an adjacent facet and neighboring cells or (4) **internal split** between two facets in the focus cell.

The four different cases are illustrated in Figure 6 and can be characterized as follows (with cases 1-4 and a-f referring to Figure 6 and facets numbers referring to Figure 5):

1. **Stay**. This is when the facet aspect is 0-45° (a) and called stay because this water won't be internally moved between facets (although externally routed later on).

2. **Move**. If the facet aspect is 90-180° (d) the water will be internally moved to facet number 2. If the aspect is 225-270° (e) it will instead move to facet number 8. When a facet is coded *move*, all of its water will be moved internally to another facet.

3. **Split**. If the facet aspect is 45-90° (b) one part of the water will be moved to facet number 2 and the other part externally routed out of the cell. The proportions routed externally and moved internally are decided by a split of the aspect vector. It is split between the 45° vector, pointing externally towards the neighboring cell C_2, and the 90° vector, pointing internally towards facet number 2 inside the focus cell. If the facet aspect is 270-360° a corresponding split will be done with some water moved to facet number 8 and the rest being externally routed.

4. **Internal split**. If instead the facet aspect is between 180° and 225° (f) the water will be split between two of the focus cells facets, number 2 and number 8. After the *split* no water will remain in facet number 1.

Note that in the special case of two facets sloping towards each other these facets will be coded with *stay*.

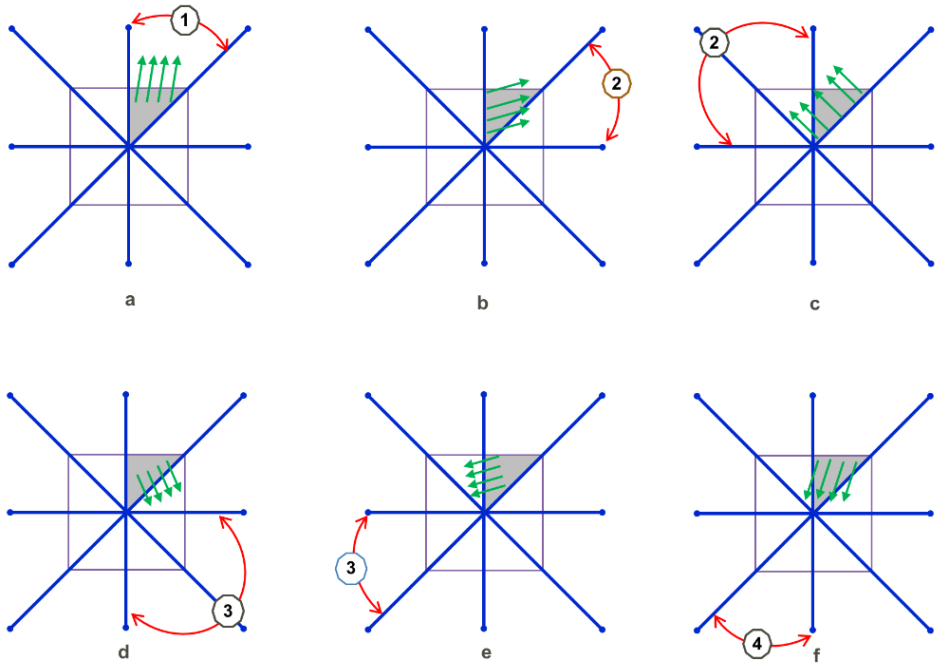

Figure 6. Shows the different possible facet aspect values and how the water is coded in 4 different ways (1 stay, 2 move, 3 split and 4 internal split). The figure is reprinted with minor modifications and permission from Pilesjö and Hasan (2014).

The water is moved within a cell from facet to facet. Sometimes water can be moved through more than one facet, e.g. from facet 1 to facet 2 and continuing to facet 3 and even farther. This goes on until all water ends up in facets denoted as *stay* or *split* (fractions that are to be externally routed). Now the internal routing process is done and the water is coded for the calculations of external routing to neighboring cells.

Depending of the shape of the DEM surface the water distribution between facets can look very different. Three examples of facet distributions, which are shown in Figure 7a-c, will now be described. For a cell (a) located at a saddle point in the landscape the cell internal process can conclude that no water is to be moved internally and all facets are coded *stay*. This means that the water from every facet is ready to be routed externally out of the cell. As a contrast to that case, another cell (b) could be located in a landscape sloping so that all of its water is moved internally, to one single facet from which it is to be externally routed or remain (if the cell is a sink). In a third example (c) the water could also be moved internally so that only a couple of facets are holding the water for external routing.

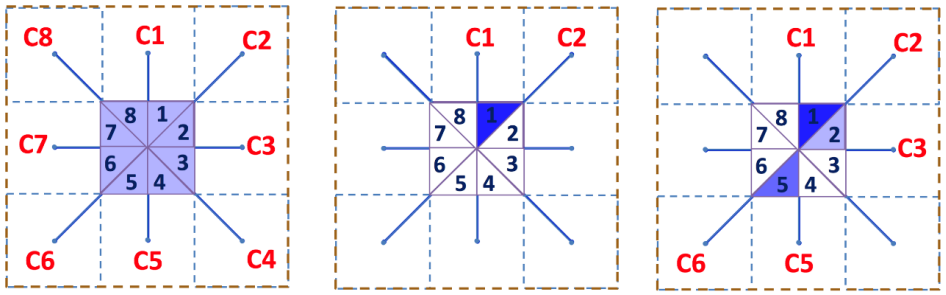

Figure 7. Illustration of facet water distributions where a darker blue indicates more water. Examples showing (a) having 1/8 of the water in each facet, (b) having all water in facet number 1 and (c) having most of the water in facet 1, some in facet 5 and some less in facet 2. The neighboring cells are having C1-C8 written in red out as an indication of which neighboring cells that the external flow can be routed to. The figure is reprinted with modifications to and permission from Pilesjö and Hasan (2014).

2.4.2 TFM - flow routing between cells

After the cell internal "water-moving" process is done the external routing to neighboring cells takes place. For each facet holding water this can be routed to 0, 1 or 2 neighboring cells (in Figure 7b cells `C1` and `C2` are the possible neighbors getting water routed from facet 1). All water in one facet will be routed to one cell if it is the only neighboring cell lower than the focus cell. If none of the neighboring cells are lower in elevation no water will be routed to them. If both neighboring cells are lower than the focus cell then the flow proportions are divided between them as:

$$f_i = \frac{(\tan \beta_i)^x}{\sum_{j=1}^{2}(\tan \beta_j)^x} \quad , for\ all\ \beta > 0 \tag{7}$$

Where: ***i, j***=flow directions to lower cells for $i, j \in \{1,2\}$
$\boldsymbol{f_i}$=flow proportion in direction *i* for $f_i \in [0,1]$
tan $\boldsymbol{\beta_i}$= slope gradient between cell *i* and the focus cell.
x=a variable exponent.

In (7) x can be set as preferred. There are two totally different approaches that can be observed, when either x is set to 1, or on the other hand when $x \longrightarrow \infty$. When $x = 1$ the flow is distributed between the two downhill, neighboring cells proportionally to the slope and when x is big (approaches infinity) all flow is distributed to the neighbor cell with the steepest slope. In this study x is set to 1 since it has been shown to give the best results for both concave and convex surfaces (Pilesjö and Zhou, 1997).

Based on the eight facets flow proportions (f_i) the flow directions for each cell inside the border is estimated (the border cells are not assigned any flow directions). When done for a full area a set of *flow distribution* matrices (FD_{TFM}) are generated. The FD_{TFM}-matrices define the proportion of water routed from the focus cells to their neighboring cells. Since there are eight neighbors to each cell there will be eight matrices $FD_{TFM}(r,c)_k$ with the k indicating the flow direction as illustrated in Figure 8. Each of $FD_{TFM}(r,c)_k$ contains the fractions of water distributed from the focus cell to its neighboring cell in direction k. $FD_{TFM}(r,c)_1$ will denote the proportion routed from the focus cell in row *r*, column *c* to the neighboring cell C1 and $FD(r,c)_2$ the proportion to neighbor C2 (as the neighbors where denoted in Figure 7). Hence this flow distribution algorithm describes a function $f_{FD_{TFM}}$ that generates the $FD_{TFM,k}$ matrices based on the ground elevation in each focus cell and its eight neighbors:

$$FD_{TFM}(r,c)_k := f_{FD_{TFM}}\big(z(r+i, c+j)\big) \tag{8}$$

Where: $\boldsymbol{FD_{TFM}(r,c)_k}$ defines the flow distribution in row *r*, column *c* as estimated by the TFM algorithm for all cells inside the border in all directions $k \in \{1..8\}$.
$\boldsymbol{i,j}$ indicates that FD_{TFM} is a function of the focus cells eight neighboring cells, i.e. $i, j \in \{-1,0,1\}$.

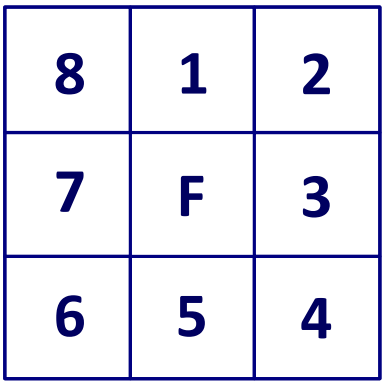

Figure 8. Illustration of indexes for the eight flow directions out of a focus cell (F). This neighbor direction index is used throughout the thesis and is denoted *k*.

2.4.3 TFM - drainage area estimation

The drainage area, sometimes called flow accumulation, is the total area that can drain to a certain cell given no interfering processes such as infiltration, evaporation or similar. The drainage area will be estimated for every cell in the area of interest. To begin with all cells will first be assigned water corresponding to one cell area. These are the initial values in the drainage area matrix (DA). Then the analysis continues by letting water flow out from the cells that don't have any flow going into them (saddle points). These cells will pass on their water to downstream neighbors according to the flow distribution, the $FD_{TFM,k}$ proportions. The cells getting incoming water add this to their current cell value. This continued routing process is done with a design so that each cell only gets to pass on water downstream when it has gotten the inflow from every one of its neighbors which has a higher elevation value. The algorithm continues until all cells have had water routed in and out as the FD_{TFM}-matrices indicates. The resulting DA_{TFM}-matrix contains the accumulated drainage area (m^2) for every cell inside the border cells.

2.4.4 TFM flow distribution and drainage area visualized

One of the main things that sets the TFM apart from the D8 is that it can estimate the flow to diverge into many neighboring cells as would happen at saddle points in the landscape. Such a case is exemplified in Figure 9 with a 3x3 elevation window showing a saddle point in the center cell, the corresponding contour lines and the logical flow lines of the diverging flow. In Figure 10 the resulting flow proportions are plotted as percentage of the center cells water that is routed in each direction.

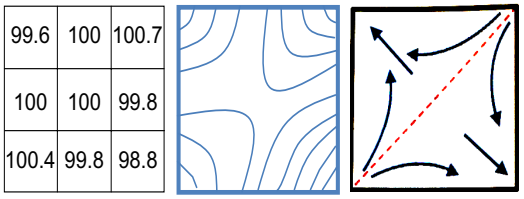

Figure 9. An example of elevation setting for a saddle point, its contour lines and the corresponding logical flow in case of rainfall runoff (Pilesjö and Hasan, 2014).

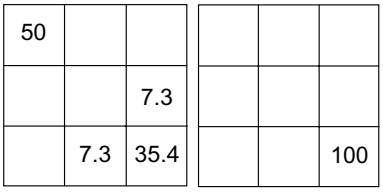

Figure 10. The flow distribution from the focus cell estimated by the TFM (left) and the D8 (right) (Pilesjö and Hasan, 2014).

An example of the TFM algorithms estimated drainage area can be seen in Figure 11 together with corresponding results from the D8 algorithm. It shows how the width of the DA_{D8} drainage paths are always 1 cell wide while the TFM estimates drainage that is both getting wide and narrow during the runoff. This is a result of only allowing converging flow in the D8 while the TFM also can estimate diverging flow (when the surface is convex).

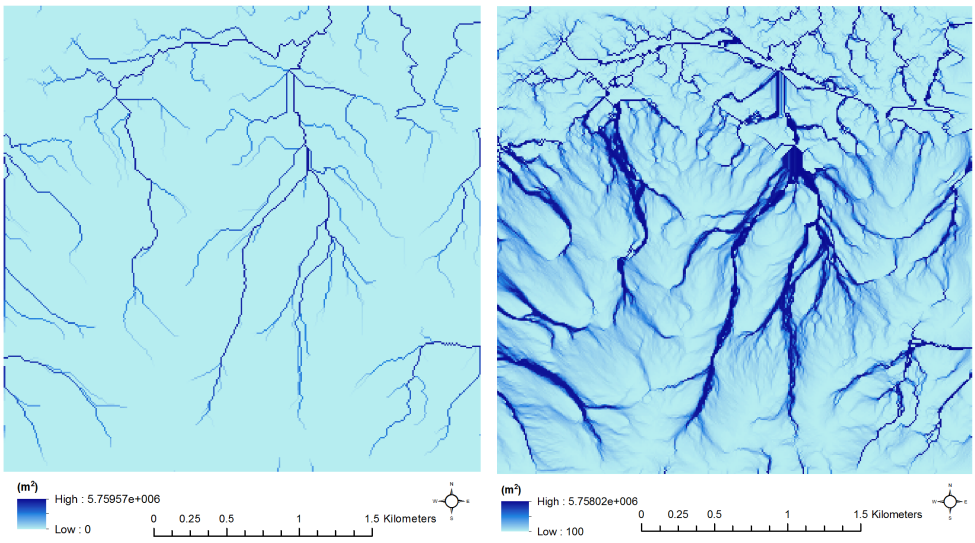

Figure 11. An example of the estimated accumulated drainage areas by the D8 (left) and the TFM (right). The multiple flow directions used in TFM enables the flow to shift between narrow and wide whereas the single flow direction in D8 only generates flow paths of single cell width. Located up and right from the maps center there is an area with straight flow paths. They are straight due to it being a flat surface for which the algorithms generate flow paths that are leading straight through it and continuing downhill.

3 Method

This chapter starts off with the ideas, mathematics and methods for the new, developed, model. That is followed by a review of the data used for the test areas. The chapter is finished with a description of how the test study setups were chosen and used for trying out the model.

3.1 Setting up a new dynamic model

The focus of this study is to setup a new model developed from the EFR models. For this purpose, TFM is chosen, among other flow direction algorithms, as core method for flow routing. This is motivated by its accurate performance at mapping drainage area for theoretical flow over mathematical surfaces (Pilesjö and Hasan, 2014). In contrast to existing EFR models, that estimate accumulated drainage area (m^2), the new model will estimate the distribution of water volumes (m^3) and water depths (m). Building from the EFR models, additional input data and dynamics are introduced step by step throughout this chapter.

3.1.1 Precipitation, infiltration and surface roughness

To get a more accurate description of reality the concepts of heterogeneous rainfall and heterogeneous ground characteristics are introduced. That is to describe the spatial variation in precipitation, infiltration, and surface roughness. Also using intensities, e.g. mm/h or m/s, enables the model to estimate water depths instead of the possible drainage areas that are estimated by the EFR models. To take the spatial variation into account matrices for precipitation (P) and infiltration (I) are input to the model framework. Each cell in these input-matrices describe the intensities in mm/h (the units are transformed to m/s for use in the upcoming calculations). A third matrix containing surface roughness (n), known as Manning's friction value, is also a required model input. Both surface roughness and infiltration will vary in relation to land-use.

3.1.2 Introducing a time step

A time dimension is introduced in an attempt at capturing the runoff process' dynamic changes during and after a rainfall. The time dimension makes it possible to map where and when there will be high water levels. This in contrast to the static model that only mapped the total area that possibly could drain to each cell.

For the discretization of time, a time step Δt is set, and iterations are done from starting time $t = 0$ until the desired end time $t = T_{total}$. The duration of the rainfall T_p is set by the user and can be less than the total run time, i.e. the modelling can go on after

it has stopped raining. The precipitation intensity can vary with time and is therefore indexed as P_t.

To model one time step the procedure starts by letting it rain and infiltrate in each cell. This is here called a *volume-precipitation-infiltration* (*VPI*) calculation which is carried out by adding precipitation to, and subtracting infiltration from, the current water volume in each cell:

$$\begin{cases} VPI(r,c)_t = \big(P(r,c)_t - I(r,c)\big) * \Delta t * cellsize^2 & , for\ t = 0 \\ VPI(r,c)_t = V(r,c)_{t-1} + \big(P(r,c)_t - I(\mathrm{r},\mathrm{c})\big) * \Delta t * cellsize^2 & , for\ t > 0 \end{cases} \quad (9)$$

Where: $\boldsymbol{VPI(r,c)_t}$ is the water volume (m^3) in row r and column c at time step t to be used in the upcoming calculations.
$\boldsymbol{V(r,c)_{t-1}}$ is the volume (m^3) from the last time step.
$\boldsymbol{P_t}$is the precipitation (m/s) at time step t.
$\boldsymbol{I}$ the infiltration (m/s) which is set constant over time.
$\boldsymbol{\Delta t}$ is the time step size (s).
$\boldsymbol{cellsize}$ is the length (m) of the side of each cell (square cells).

Volumes are the units used for calculations and the volume in each cell is expected to be evenly distributed over the cell surfaces in the beginning of every time step. Only positive water volumes can exist on a surface and hence all cells with negative values, where more water than present could infiltrate, are set to zero in VPI_t before continuing.

By the use of TFM's flow distribution algorithm (chapter 2.4), the distribution of water, routed out of each cell, is determined by the elevation in each cell and its eight neighbors. In the static TFM model this was based on the ground elevation as defined in (8). As of now a model is instead setup where water levels are changing dynamically. Water can be filling up sinks and flowing out of them in new directions, for example. To deal with the change in water surface elevation a new flow distribution is calculated at every time step, now depending on the ground elevation plus the water depth. The model assumes the water to be evenly distributed over the surface in each cell and estimates a flow distribution based on the water surface elevation z'.

$$\begin{cases} d(r,c)_t = \dfrac{VPI(r,c)_t}{cellsize^2} \\ z'(r,c)_t = z(r,c) + d(r,c)_t \end{cases} \quad (10)$$

Where: $\boldsymbol{d(r,c)_t}$ is the water depth (m) in row *r*, column *c* at time step *t*.
$\boldsymbol{VPI(r,c)_t}$ is the water volume (m^3) in row *r* and column *c* at time step *t* before the water is routed between cells.
$\boldsymbol{z'(r,c)_t}$ is the water surface elevation (m) in row *r*, column *c* at time step *t*.
$\boldsymbol{z(r,c)}$ is the ground elevation (m) in row *r*, column *c*.
$\boldsymbol{cellsize}$ is the length (m) of the side of each cell (square cells).

The flow distribution is estimated for all cells, except the border cells, as in the theory in equation (8), although now estimated from the water surface elevation $\boldsymbol{z'}$ instead of the ground elevation $\boldsymbol{z}$:

$$FD_{TFM}(r,c)_{\,t,k} = f_{FD_{TFM}}(z'(r+i,c+j)_t) \tag{11}$$

Where: $\boldsymbol{FD_{TFM}(r,c)_{\,t,k}}$ is the new flow distribution proportions in row *r*, cell *c,* at time step t with index $\boldsymbol{k}$ indicating routing direction.
$\boldsymbol{f_{FD_{TFM}}}$ is the TFM flow direction function.
$\boldsymbol{z'(r+i,c+j)_t}$ is the water surface elevation at time step *t.*
$\boldsymbol{i,j}$ indicates that FD_{TFM} is a function of the focus cells eight neighboring cells, i.e. $i,j \in \{-1,0,1\}$.

Now the water volume $VPI(r,c)_t$ is modelled to flow out of the focus cell, towards its neighbors according to the flow distribution in the $FD(r,c)_{\,t,k}$ matrices. When the water flow between cells is modelled it is assumed to be flowing from the focus cell's center, over a surface with constant slope, to the next cell center. The amount of water routed from a cell, in one time step, will be dependent on the estimated water velocity and how far the average water makes it from one cell center to the next.

The water velocity is estimated by the Manning formula for uniform open channel flow (Ward and Elliot, 1995):

$$v = \frac{1}{n} R_h^{\frac{2}{3}} \beta^{\frac{1}{2}} \tag{12}$$

Where: $\boldsymbol{v}$ is the average velocity (m/s) in the channel cross-section.
$\boldsymbol{n}$ is the Manning friction value.
$\boldsymbol{R_h}$ is the Hydraulic radius (m).
$\boldsymbol{\beta}$ is the slope (m/m).

To define the hydraulic radius and slope in this model two new variables are introduced. The first one is the difference in water surface elevation in the focus cell and the downstream cell, denoted Δh, shown in Figure 12 and defined in (13).

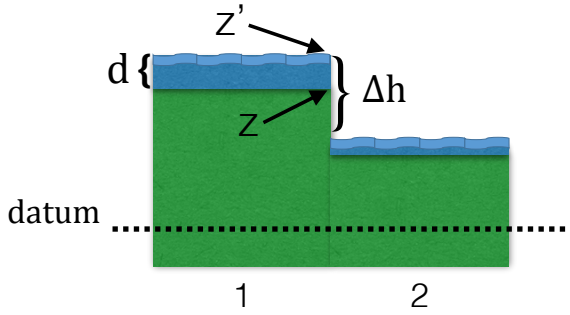

Figure 12. An illustration of a case where water is to be routed from an elevated cell (1) to a downstream neighbor (2). For the elevated cell (1) the ground elevation is z above a certain datum and z' is the the water surface elevation over the same datum.

$$\Delta h(r,c)_{t,k} = z'(r,c)_t - z'(r,c)_{t,k} \quad (13)$$

Where: $\Delta \boldsymbol{h}(\boldsymbol{r},\boldsymbol{c})_{t,k}$ is the water surface elevation difference (m) between the focus cell and the downstream cell in direction *k* at time *t* where $k \in \{1..8\}$.
$\boldsymbol{z}'(\boldsymbol{r},\boldsymbol{c})_t$ is the water surface elevation (m) in row *r*, column *c* at time *t*.
$\boldsymbol{z}'(\boldsymbol{r},\boldsymbol{c})_{t,k}$ is the water surface elevation (m) in the the neighboring cell in direction *k* from row *r* and column *c*.

A second new variable is introduced to be used for the calculating water velocity in the special case where a depression is getting filled with water. In Figure 13, an example is shown where a sink cell (denoted 3) has an elevated water surface in comparison to its neighbor (denoted 2) and therefore water is flowing from (3) to (2). The new variable is called the pressure head height $h_{ph}(r,c)_{t,k}$, is defined in equation (14) and is used in the upcoming velocity calculations. It is the head height that determines the water velocity. This head is chosen as the smallest value between the water depth in the focus cell and the water surface elevation difference to the next cell in direction k to assign a reasonable head for estimating velocity. Only using the water depth d would otherwise assign a too big head, as shown in Figure 13, where Δh rather than d should be used for the velocity estimation.

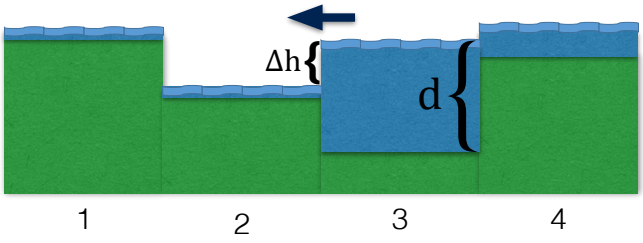

Figure 13. An illustration of the special case when a sink cell (3) is filled and water should flow opposite the ground slope direction. Water depth for cell 3 is marked as d and the water surface elevation difference between cell 3 and 2 is denoted Δh.

$$h_{ph}(r,c)_{t,k} = \min\left(d(r,c)_t, \Delta h(r,c)_{t,k}\right) \quad (14)$$

Where: $\boldsymbol{h_{ph}(r,c)_{t,k}}$ is the pressure head height (m) at time step *t* for water flowing in direction *k* from the cell in row *r*, column *c*.
$\boldsymbol{d(r,c)_t}$ is the water depth (m) in row *r*, column *c* at time step *t*.
$\boldsymbol{\Delta h(r,c)_{t,k}}$ is the water surface elevation difference (m) between the focus cell and the downstream cell in direction *k* at time *t*.

The velocity for overland flow is described as uniform flow in an open channel. This is justified by modelling the water flow, from one cell to the next, as water flowing through a rectangular channel with a base as P, the wetted perimeter, and the height set to h_{ph}, as the pressure head height. Thus the hydraulic radius for flow can be estimated as below and becomes equal to the pressure head height h_{ph}.

$$R_h = \frac{A}{P} = \frac{h_{ph} * P}{P} = h_{ph}$$

Where: $\boldsymbol{R_h}$ is the hydraulic radius (m).
$\boldsymbol{A}$ is the cross-sectional area (m^2).
$\boldsymbol{P}$ is the wetted perimeter (m).
$\boldsymbol{h_{ph}}$ is the pressure head height (m).

The water velocity for water routed out of a cell is estimated as an average water velocity, calculated in each direction $k \in \{1..8\}$ as below.

$$v(r,c)_{t,k} = \frac{1}{n(r,c)} h_{ph}(r,c)_{t,k}^{\frac{2}{3}} \beta^*(r,c)_{t,k}^{\frac{1}{2}} \quad (15)$$

Where: $\boldsymbol{v(r,c)_{t,k}}$is the water velocity (m/s) in direction *k* at time *t* in row *r*, cell *c*.
$\boldsymbol{n(r,c)}$ is the Manning friction value in row r and column c.
$\boldsymbol{h_{ph}(r,c)_{t,k}}$ is the pressure head height difference (m) in focus cell to water surface in neighboring cell, in direction *k*, at time *t*.
$\boldsymbol{\beta^*(r,c)_{t,k}}$ is the slope in regard to water surface, between the focus cell and the neighbor, in direction *k* at time *t*.

The slope $\beta^*_{t,k}$ is the water surface slope towards the k:th neighboring cell at time t and is calculated as a ratio. It is denoted with a * not to be mistaken with other more common definitions of slope in GIS related literature. This slope represents the slope for water flowing from one cell to the next and is calculated from the difference in water surface elevation divided by the distance to next cell center:

$$\begin{cases} \beta^*(r,c)_{t,k} = \dfrac{\Delta h(r,c)_{t,k}}{L_k} \\ L_k = \begin{cases} cellsize & for\ k = 1,3,5,7 \\ \sqrt{2} * cellsize & for\ k = 2,4,6,8 \end{cases} \end{cases} \quad (16)$$

Where: $\boldsymbol{\beta^*(r,c)_{t,k}}$ is the slope in regard to water surfaces, between the focus cell and its neighbor, in direction *k* at time *t*.
$\Delta\boldsymbol{h_{t,k}}$ is the water surface elevation difference (m) between focus cell and downstream cell in direction *k* at time *t*.
$\boldsymbol{L_k}$ is the distance to the next cell center (m) in direction *k*.
$\boldsymbol{cellsize}$ is the length of the cell side (m).

The volume of water available for routing to neighboring cells is determined by how large proportion of the focus cells water is above the water surface in the neighboring cells as is seen in Figure 14.

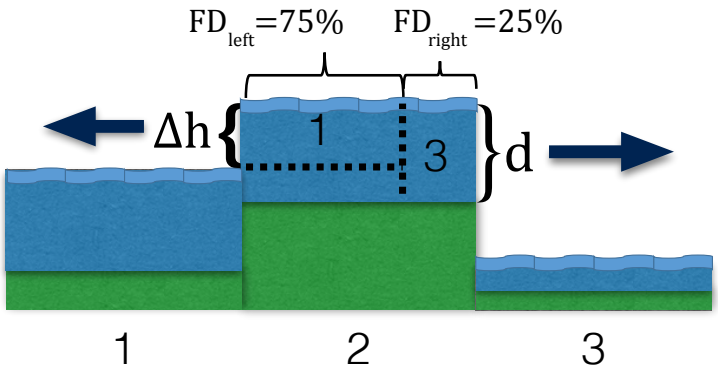

Figure 14. Water in the center cell is to be routed left and right. TFM has estimated 75% of the water to be available for routing left and 25% to be available for routing right. The available water depth for routing left is $\Delta h/d$ while the full depth of water going right is available since the ground elevation in (2) is above the water surface elevation in (3).

The routed water volume in a certain direction is estimated by the available water, which is a proportion of the VPI volume, times the proportion to be distributed in this direction, according to TFM ($FD_{TFM}(r,c)_{t,k}$), times the ratio between: the distance the water flows in one time step ($v * \Delta t$) and the distance to the next cell center (L_k).

$$V_{Out}(r,c)_{t,k} = min\left(\frac{\Delta h(r,c)_{t,k}}{d(r,c)_t}, 1\right) * VPI(r,c)_t * FD_{TFM}(r,c)_{t,k} \frac{v(r,c)_{t,k} * \Delta t}{L_k} \quad (17)$$

Where: $\boldsymbol{V_{Out}(r,c)_{t,k}}$ is the water volume (m^3) routed out from the focus cell at time step *t* in direction *k*.
$\boldsymbol{VPI(r,c)_t}$ is the water volume (m^3) in row *r* and column *c* at time step *t* before the water is routed between cells.
$\boldsymbol{FD_{TFM}(r,c)_{t,k}}$ is the flow distribution proportion routed towards neighbor *k*.
$\boldsymbol{v(r,c)_{t,k}}$is the water velocity (m/s) in direction *k* at time *t* in row *r*, cell *c*.
$\boldsymbol{\Delta t}$ is the time step size (s).
$\boldsymbol{L_k}$ is the distance to the next cell centre (m) in direction *k*.

At each time step t the volume going out, $V_{Out}(r,c)_{t,k}$, is estimated for all non-border cells and their eight possible routing directions (k). Then the new water volume in each cell ($V(r,c)_t$) is estimated as: the water volume from last time step, plus precipitation, minus infiltration (sums up as $VPI(r,c)_t$), minus the sum of water flowing out to any lower neighbors, plus the sum of water flowing in from any elevated neighbors. As described in (18).

$$V(r,c)_t = VPI(r,c)_t - \sum_{1 \le k \le 8} V_{Out}(r,c)_{t,k} + \sum_{\substack{-1 \le i \le 1 \\ -1 \le j \le 1}} V_{out}(r+i, c+j)_{t,k_{focus}} \quad (18)$$

Where: $\boldsymbol{V(r,c)_t}$ is the volume (m^3) at time *t* after the precipitation, infiltration and routing has been done. For each time step this is calculated for all rows and columns.
$\boldsymbol{VPI(r,c)_t}$ is the water volume (m^3) in row *r* and column *c* at time step *t* before water routing between cells.
$\boldsymbol{V_{Out}(r,c)_{t,k}}$ is the water volume (m^3) routed out from the focus cell at time step *t* in direction *k*.
$\boldsymbol{i,j}$ indicates that V_{out} is the volume out of the focus cells neighbors ($i, j \in \{-1,0,1\}$) in direction $\boldsymbol{k_{focus}}$ towards the focus cell in (*r*,*c*).

The model explicitly calculates $V(r,c)_t$ for all cells inside the border, at every time step, whereas the border cells' water volumes are set to zero at every time step not to affect the outflow from their inner neighbors. The results to be analyzed are the water volumes/depths in all cells (except the border cells).

3.1.3 Stability of the model

For the model to make sense it can only route a proportion of the current water volume from one cell to the next, never more water than there is in the cell. Consider water in the focus cell that is being routed towards one of its neighbors. The proportion of available water to flow out in one time step is decided by the velocity multiplied with the time step divided by the distance to next cell center. This is the ratio applied in equation (17) and is present on its own in (19). As the theory on stability in explicit numerical models proclaims this ratio is called the Courant number and has to stay below 1 (Hingray et al., 2014, Courant et al., 1967). This is because if the velocity multiplied

with the time step gets bigger than the distance, then the water flow farther than to the next cell, which cannot not be allowed.

$$\frac{\Delta t * v}{L} \leq 1 \tag{19}$$

Where: Δt is the time step size (s).
v is the velocity (m/s).
L is the distance (m) to the next cell center.

As the model is implemented it will stop and give a warning if the Courant number gets bigger than 1 in any cell at any time. To avoid that this happens the program has been given two specific implementations. First the user is recommended to set a maximum allowed slope β^*_{max} to be used in the velocity calculation (15). This meaning that every cell sloping steeper towards its neighbors is instead associated with the artificial max slope β^*_{max} that is set by the user. This max slope doesn't change the DEM but is used to prevent "unnaturally" steep features from generating extreme velocities that could violate the stability condition (19). One example of this is the drop from a building rooftop cell to a neighboring ground cell. Thus, to deal with the special case of steep drops, the slope calculation in equation (16) is modified as follows:

$$\beta^*(r,c)_{t,k} = \begin{cases} \dfrac{\Delta h(r,c)_{t,k}}{L_k} & if\ \dfrac{\Delta h(r,c)_{t,k}}{L_k} < \beta^*_{max} \\ \beta^*_{max} & if\ \dfrac{\Delta h(r,c)_{t,k}}{L_k} \geq \beta^*_{max} \end{cases} \tag{20}$$

Where: $\beta^*(r,c)_{t,k}$ is the slope (as a fraction) in regard to water surface, between the focus cell and the neighbor, in direction *k* at time *t*. It is valid in all cells inside the border, at every time step in all directions $k \in \{1..8\}$.
$\Delta h_{t,k}$ is the water surface elevation difference (m) between focus cell and downstream cell in direction *k* at time *t*.
L_k is the distance to the next cell centre (m) in direction *k*.
β^*_{max} is the maximum slope (as a fraction) set by the user.

The second precaution, to avoid violating the stability condition, is to choose a small enough time step already from the beginning. To facilitate this the program asks for a user guess of the maximum water depth h_{guess} (pressure head height) as input and suggests a safe time step $\Delta t'$ (i.e. small enough). The model will then be stable with that time step as long as it doesn't violate the velocity condition in (19). This won't happen as long as the water depth does not exceed the users guessed depth, in the steepest areas, which have the lowest friction values. The time step suggestion is done by first estimating the maximum possible velocity v_{max} for all cells inside the border. v_{max} is estimated for both diagonal directions (will be indexed *d*) and straight directions (will be indexed *s*). Then the maximum velocity is used to estimate a time step as ($\Delta t'$) below:

$$\boldsymbol{v_{max}(m,n)_s} = \max\left((h_{guess})^{\frac{2}{3}} * \beta^*(r,c)_k^{\frac{1}{2}} * \left(\frac{1}{n(r,c)}\right)\right) \qquad ,for\ all\ r,c\ and\ k = 1,3,5,7$$

$$\boldsymbol{v_{max}(m,n)_d} = \max\left((h_{guess})^{\frac{2}{3}} * \beta^*(r,c)_k^{\frac{1}{2}} * \left(\frac{1}{n(r,c)}\right)\right) \qquad ,for\ all\ r,c\ and\ k = 2,4,6,8$$

$$\Delta t' \leq \min\left(\frac{L_s}{\boldsymbol{v_{max}(m,n)_s}}, \frac{L_d}{\boldsymbol{v_{max}(m,n)_d}}\right)$$

Where: $\boldsymbol{v_{max}(m,n)_k}$ is the estimated maximum velocity (m/s) to be modelled in row *m*, column *n* and direction k.
$\boldsymbol{h_{guess}}$ is the users guess of the maximum pressure head height (m) to affect the water velocity.
$\boldsymbol{\beta^*(r,c)_{t,k}}$ is the slope (as a fraction) in regard to water surface, between the focus cell, in row *r*, column *c*, and the neighbor, in direction *k* at time *t*.
$\boldsymbol{n(r,c)}$ is the Manning friction value in row *r* and column *c*.
$\boldsymbol{\Delta t'}$ is the suggested time step (s) small enough for $h_{estimate}$.
$\boldsymbol{L_s}$ is the distance (m) to a horizontally or vertically adjacent cell.
$\boldsymbol{L_d}$ is the distance (m) to a diagonally adjacent cell.

The time step $\Delta t'$ is a suggestion and does not assure the model is going to be stable since the modelled water depths won't be known until runtime.

3.1.4 Storm sewer inlets and outlets

The dynamic model is constructed with the possibility to incorporate the storm sewer system inlets. In urban areas any storm sewer inlets that are known with location can be included if some kind of estimate on inlet capacity is done. The inlets' estimated capacity (volumes per time unit) can then be added to the infiltration raster (as millimeters per hour) as it is representing water that is taken out of the model.

In a corresponding manner any water outlets that are identified can be located, estimated and added to the precipitation values in the precipitation raster.

3.2 Data

3.2.1 Data for rural modelling – the Abisko test area

A 1,5x1,5 km test area located ca. 10 km east of Abisko, northern Sweden was chosen for rural test runs. LiDAR with a mean density of 2.4 points/m² was made available through Lund University (Hasan et al., 2012b). No other data on soil type, vegetation or similar were collected as this was mainly an elevation based test area. Neither were any real measurements on precipitation collected.

3.2.2 Data for urban modelling – the Lund test area

For trying out the model in urban settings an area in northern Lund, Sweden, were chosen based on reports of it having had problems with flooding during heavy rain fall. The area stretches along the Gästgivarevägen street and consists of two neighboring watersheds that together form an area of 13.8816 ha. It has previously been studied for drainage system overflow using a semi distributed model. From that previous study; data on watershed areas, storm sewer inlet coordinates, pervious/impervious areas have been made available by Hallen (2016) and are here presented in Figure 15. LiDAR data with an average density of 12 returns/m², were made available by the municipality in Lund (Lunds Kommun). No recordings on rainfall intensities, water depths or similar, for actual flooding events, was searched for or used as comparison in this theoretical study.

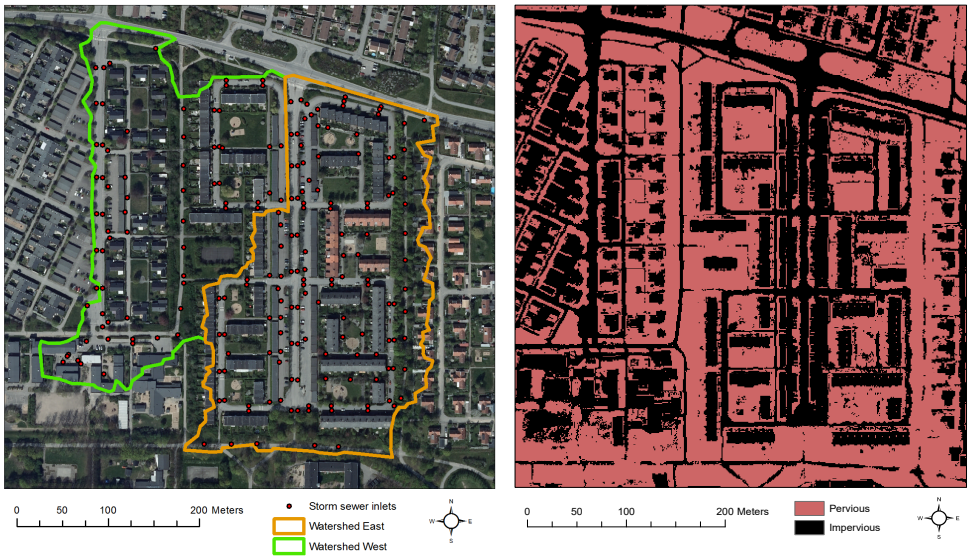

Figure 15. The urban study area is located around Gästgivarevägen in the northern part of Lund, Sweden. The left map illustrates the study areas two delineated watersheds and the storm sewer inlets. The right map shows a 1x1 m resolution raster on which areas are classified as pervious and impervious. The data on watersheds, storm sewer inlets and impervious areas were made available and used with permission from Hallen (2016). GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

3.3 Test areas and study setup

The rural area in Abisko and one urban area in Lund were setup to check what kind of results can be expected from the model. The rural setup was also used for testing how robust the model is to different sizes of time steps and for the gains of some code optimizations. In this chapter the areas, the preparation of elevation data, and the model settings are presented.

3.3.1 Modelling overland flow in Abisko - a rural model setup

For the Abisko test area a 150x150 cell DEM raster with cell size 10x10 m was interpolated from the LiDAR data using invers distance weighted method with minimum of 12 neighbors and the exponent $k = 2$. The DEM, seen in Figure 16, has its south west corner located in x-coordinate 664194.67 and y-coordinate 7584395.46 in the SWEREF99 TM coordinate system.

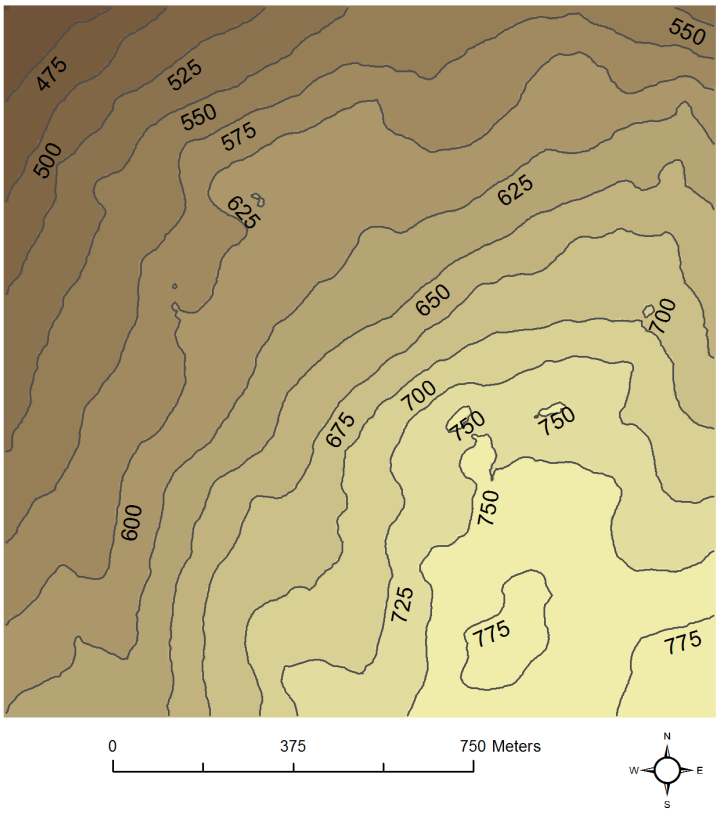

Figure 16. The Abisko test area DEM is a 150x150 cell raster, built up in a 10x10 m resolution. It shows a small mountain with its peak reaching 790 meters above sea level.

To estimate water volumes and depth in Abisko a 30-minute rainfall was used on the newly created DEM. The rain duration was set between T=0 until T=30 and the total modelling time was set to T=40. This means that the model runs for another 10 minutes, after the precipitation has stopped, letting the water flow downhill and infiltrate. The time step (Δt) was set to 3 s. The precipitation was set to 50 mm/h intensity for every cell in the area. Correspondingly the whole area was assigned an infiltration of 40 mm/h. The Manning surface friction value (n) was set to 0.05 for all cells. The max slope (β^*_{max} used in the velocity calculation) was set to 11.3° (20%). Values on rain intensity,

infiltration and Manning surface friction were chosen to represent a possible heavy rain in Swedish conditions, no specific consideration were taken to the local conditions in Abisko.

3.3.2 Modelling overland flow in northern Lund – an urban model setup

Since no actual reports of precipitation during flooding events were searched for or used, a rain with a 30-minute duration and a 100 year return period was chosen as input. Published values on return period and duration for such a rain in Lund city is estimated to an intensity of 89 mm/h (VA Syd, 2012).

The LiDAR data from Lund municipality was used to create two DEM rasters in 1x1 and 2x2 meter resolution. Each DEM was generated by first constructing one building raster and one ground raster, separately. The building rasters were interpolated and sampled based on the LiDAR returns from rooftops only. In the same way the ground rasters were created from LiDAR ground returns only. Both the building and ground rasters were created with inverse distance weighted average interpolation for the 12 nearest neighbors, weighting them with k=2. The rasters were then mosaicked together to form complete DEMs in 1x1 meter and 2x2 meter resolutions. The 1x1 meter DEM is shown in Figure 17. The DEMs covers an area of 400x500 meters and has their south west corners located in x-coordinate 131562.14 and y-coordinate 6177856.81 in the SWEREF99 TM coordinate system.

Coordinates for the storm sewer inlets were used with the purpose of trying out how the model behaves when incorporating the sewer system in a simple manner. The data contained coordinates for 206 inlets. The majority of these inlets were reported to be private and no information on their capacity was available (Hallen, 2016), so an estimate was made. This estimate was based on the assumption that the storm water sewer system should manage a rain with a 10 year return period and a 30 minute duration as suggested in a report published by the Swedish Civil Contingencies Agency (MSB) (Mårtensson and Gustafsson, 2014). Such a rain would have an intensity of 42 mm/h (VA Syd, 2012). Another assumption was that runoff water, mainly from impervious areas, is expected to reach the sewer system and therefore the percentage of impervious areas were calculated and found to be 46.4% of the total area. With the total area of the watersheds being 13.88 ha a rough estimate for a constant inlet capacity was calculated to 3.64 liters per second through (21).

$$InletCapacity = \frac{total\ impervious\ area * rain\ intensity}{number\ of\ inlets} = 3.64\ l/s \qquad (21)$$

This estimate on sewer capacity was added to the infiltration raster and thus this water volume is subtracted from the storm sewer inlet cells at every time step.

Figure 17. Digital elevation model for the case study area around Gästgivarevägen in northern Lund, Sweden. The study area consists of two watersheds, here merged into one area and outlined in black. The model will run for the whole raster area but the results inside the watersheds are in focus. Data on the watersheds were made available and used with permission from Hallen (2016).

Based on the data for pervious and impervious areas different friction and infiltration values were set to commonly used values in Swedish communal storm water investigations. Pervious areas, mainly being grass covered lawns, where assigned a constant infiltration of 35 mm/h and a Manning friction value (n) of 0.5. All the impervious areas, asphalt, concrete and roofs, were assigned zero infiltration and a Manning friction value of 0.02. These infiltration and friction values have been recommended use when modelling Swedish conditions (Svensson, 2016, September 8, personal communication) (Mårtensson and Gustafsson, 2014). The model was setup for running with 30 minutes of rain and then a following 30 minutes without rain. The time step was set to 0.25 s which required 14 400 steps in time for the 60 minutes to be modelled. Results were saved every 15 seconds as well as a snapshot result for the time step with maximum water depth.

3.3.3 Testing model consistency with varying step size

To make sure the model is robust to changing the discretization of time a test was conducted using the same setup as for the 40-minute rain over the Abisko DEM. The model was run again with the same settings as described in 3.3.1 but now changing the step size (Δt) from 3 seconds to 5 seconds. Results on water depths were recorded at 30 and 40 minutes for comparison with the $\Delta t = 3$ s results.

3.3.4 Code optimization and resolution dependency

To have an effective model the runtime has to be reasonable. This model builds on the TFM algorithm for deciding flow directions, it is implemented as Matlab code and was at first not runtime optimized at all. However, an attempt was done at identifying which functions require most of the run time and how well coded they are. No thorough optimizing analysis could be performed within this thesis work due to lack of time, although a small attempt was done with the little time and experience available.

An initial timing was setup to be done with same input data as was used when ensuring consistency for different time steps (the Abisko setup with the 150x150 cell raster). The model setting was run with a step size of 5 seconds for 480 steps (again 30 minutes of rain plus 10 minutes after rain simulation). One day of work where spent on optimization trying both parallel processing of the most demanding program loops as well as trying to identify and redesign a few inefficient structures. The code was then test run against the same Abisko case as before while recording the run time as post optimization results. The computer used in both cases was a MacBook Pro with a 2,4 GHz Intel Core i5 processor and 8 GB 1600 MHz DDR3 RAM memory.

Moreover the slightly optimized code was used for running the urban test case in Lund in 1x1 meter resolution as explained in chapter 3.3.2, while recording the runtime. After this was done the same case was also run in the 2x2 meter resolution with the same settings as before, only changing the time step from 0.25 to 0.5 seconds. For this coarser resolution the modelling time is also of interest for comparison and is thus to be saved and presented among the other results. The choice of taking a double step size ($\Delta t = 0.5$) in 2x2 meter resolution is motivated by the fact that condition (19) will hold given the same maximum water velocity. This is due to the distance to next cell center being doubled (from $L = 1$ to $L = 2$).

4 Results

4.1 Modelling results with dynamic water depths in focus

This chapter starts of by presenting results of the rural and urban case studies and is continued with a presentation of the results on time step consistency and code optimization gains.

4.1.1 Results on overland flow in the rural Abisko test area

From the 30 minute rainfall event, setup for the Abisko test area, water depths at 20 and 30 minutes are presented in Figure 18. At 20 minutes the accumulated water depths above 1 mm can be seen as bright to darker blue. Most cells have less than 0.5 cm of water, but the deepest water filled sink cell has 11 cm of water. At 30 minutes it is still raining and valleys are getting filled with around 3 cm of water. The max cell depths are now 12-30 cm and are found in sinks. The sinks are plotted in significantly darker blue, situated by themselves and surrounded by brighter less flooded cells. These are water filled since the incoming surface runoff has filled them up at a much higher rate than the infiltration rate.

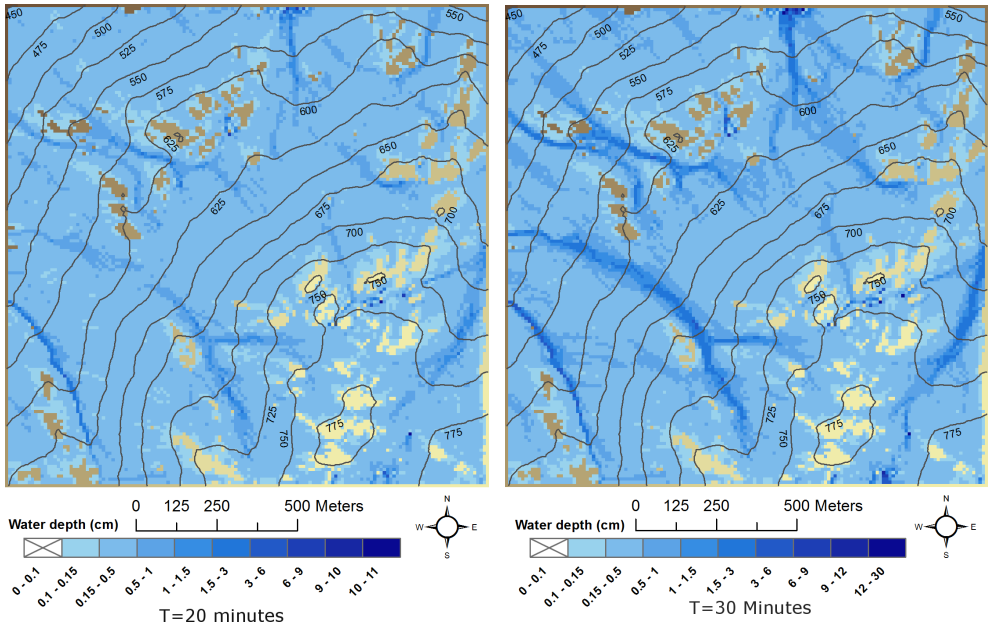

Figure 18. In the left map water depths at 20 minutes are shown for the 50 mm/h rainfall. The maximum cell depth is now 11 cm, but most areas have less than 0.5 cm of water. In the right map, the modelling time is 30 minutes, it is still raining and the valleys are getting flooded with (dark blue) water up to 6 and almost 9 cm. The maximum depths are found in a few sinks and measures up to 30 cm in the deepest sink. In both maps water levels below 1 mm have been exempt.

After 30 minutes the precipitation was stopped while the infiltration and gravitational pull downhill continued as before. Flow results for minutes 35 and 40 minutes are shown in Figure 19, plotted on top of the DEM. These results show more dry cells than before since water has both reached the valleys and also infiltrated. Even though many areas have become dry the water depths in sinks still increases and at 40 minutes the max depth has reached 35.5 cm.

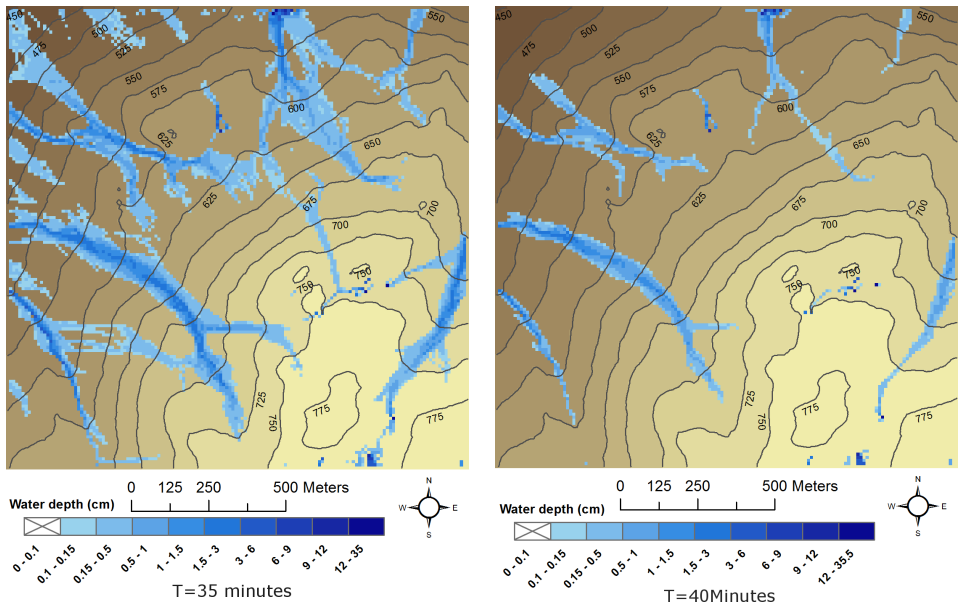

Figure 19. In the left map the water depths at 35 minutes are plotted on top of the Abisko DEM used in the modelling. Precipitation stopped at T=30 minutes and now water is flowing downhill and infiltrating the surface at the set 40 mm/h rate. On the right, modelling results at 40 minutes show that the dark blue colors are concentrated to the valleys while the majority of the cells have zero water left in them. Still, some water filled sinks and small depressions are left with significant water depths (dark blue).

4.1.2 Results on overland flow in the urban test area of northern Lund

Visual results on water depths from the 100 year rainfall with 30 minute duration over Gästgivarevägen in Lund at T=10 and T=20 minutes are presented in Figure 20. What is found to be street areas are filled up with water. Even asphalt areas with multiple storm sewer inlets gets flooded. One example of this is the bigger turnaround area, which has 5 storm sewer inlets (marked in Figure 20). Already after 20 minutes it has more than 20 cm of water in its deepest depressions.

In a few cases water has flooded individual cells located on building roofs (marked in Figure 20). These cells are artefact sinks generated due to few LiDAR returns from the roofs. Since they are so few in comparison to the total number of roof cells the main modelling results shouldn't be affected that much.

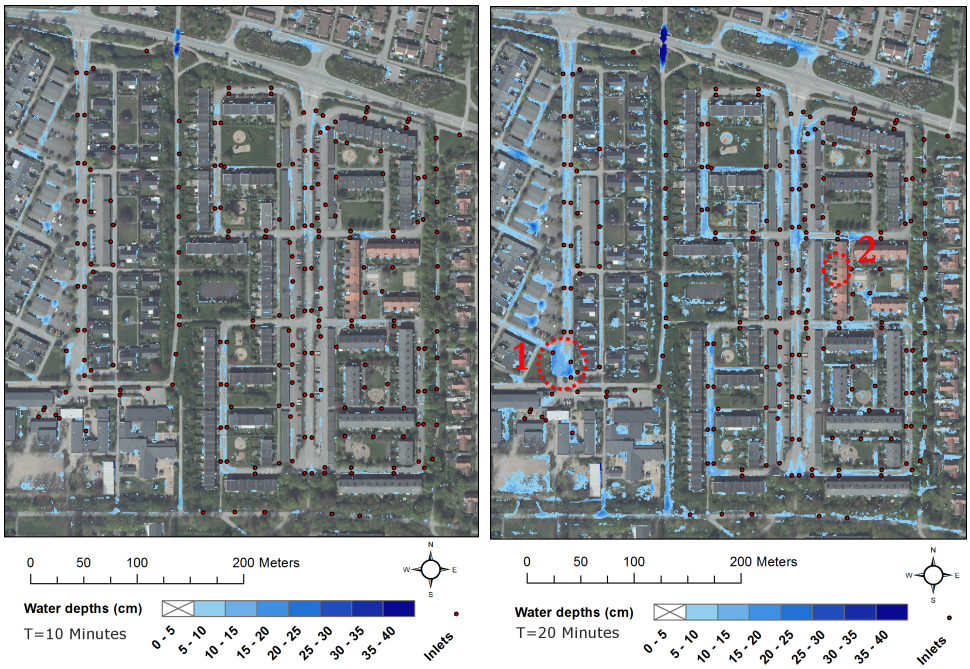

Figure 20. Water levels after 10 and 20 minutes of precipitation over Gästgivarevägen in Lund. Water levels above 5 cm are shown in blue against the background image. Levels below 5 cm have been exempt. The precipitation was set to 89 mm/h, infiltration in pervious areas to 35 mm/h, Manning friction to 0.5 in pervious areas (grass) and 0.02 on impervious areas (roofs and asphalt). The time step (Δt) was set to 0.25 s. The max slope was set to 11.3° and the storm sewer inlets (the red dots) where given a capacity of 3.64 liters per second. Marked in the right figure is (1) a flooded turnaround area and (2) two flooded artefact roof sinks. More detailed figures of the above results are found in the appendix. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

At 30 minutes of rainfall the water depths in grass areas, seen in Figure 21, have risen above 5 cm in many more spots than before. This is an indication of water flowing and filling up small depressions also in the permeable areas. In the 30-minute snapshot the following results are marked: (1) In some street areas the water depth is now rising above 10 and even 15 cm next to the houses which is a potential flood hazard. (2) The deepest flooded areas are estimated to 25-30 cm along some garages. (3) In the big turnaround area the water depths are now in the range from 10 to 30.

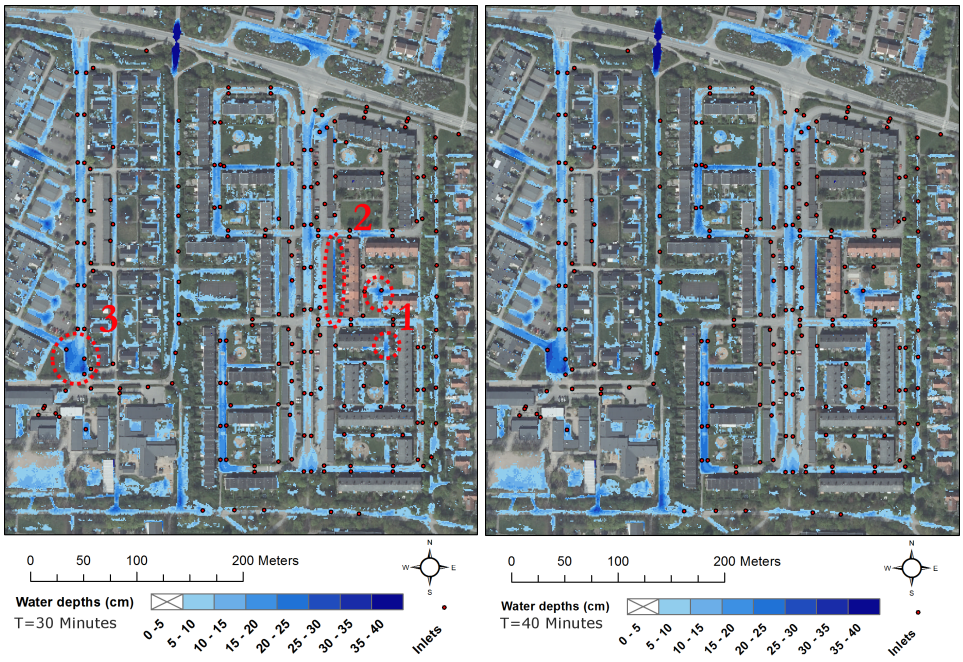

Figure 21. Water levels after 30 and 40 minutes on Gästgivarevägen in Lund. Water levels above 5 cm are shown in blue against the background image. Levels below 5 cm have been exempt. The precipitation was set to 89 mm/h until T=30 minutes and from thereon to 0 mm/h. The rest of the settings are as explained in the previous figure. Marked out are (1) two areas with 10-15 cm depths next to houses, (2) an area with 25-30 cm depth next to garages and (3) the turnaround area with 10-30 cm depths. More detailed figures of the above results are found in the appendix. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

The max water depth in the study area ground cells appeared at 35 minutes in the turnaround area and was 32 cm. After 30 minutes no more precipitation is added and less flood is seen in the grass areas as the runoff process and infiltration continues. This is seen in the 50 and 60 minutes results that are shown in Figure 22. Water is now also diminishing mainly in the north east and eastern part of the area, where the puddles are getting smaller even in the impervious streets. Here the storm sewer inlets (3.64 l/s capacity for every inlet) seems to be enough for the water depths to decrease noticably faster than in the south western part of the area. The area is also sloping towards south west (see DEM in Figure 17) which could cause surface water to flow from one end of the area to the other.

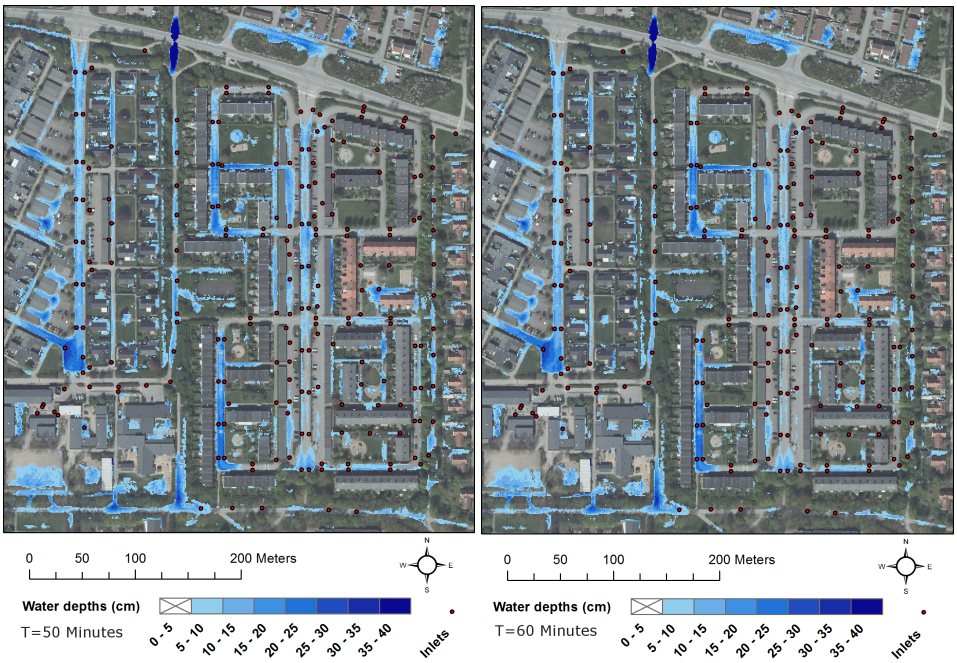

Figure 22. Water levels after 50 and 60 minutes on Gästgivarevägen in Lund. Water levels above 5 cm are shown in blue against the background image. The rainfall stopped at T=30 minutes. The rest of the model settings are explained in the previous figures. More detailed figures of the above results are found in the appendix. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

4.2 Model performance results

4.2.1 Verification of model consistency with varying time steps

The consistency test was conducted on the 150x150 cell raster of Abisko. The results are a comparison of cell water depth at given model times (T) with the step sizes 3 seconds and 5 seconds. Resulting water depth in all cells were compared at T=30 minutes and T=40 minutes. As seen in Table 1 the maximum difference in cell water depth, at T=30 min, was 8 millimeters whereas the mean differences was 8.9 µm.

Table 1. Summary of the mean and max differences in water depths, in all cells for the Abisko test area, when modelling for 480 ($\Delta t = 3\ s$) and 800 steps ($\Delta t = 5\ s$) respectively. The area was a 150x150 m DEM raster applied with a 30 minutes 50 mm/h rain followed by 10 minutes without rain.

Consistency results at model time T (minutes)	T=30	T=40
Mean cell water depth difference (mm)	0.0089	0.0023
Max cell water depth difference (mm)	8.0	2.6

4.2.2 Code optimization results and runtime to resolution dependency

At first the unoptimized code was run for the Abisko setup and timed at 2305 seconds. Then using the optimized code and same setup gave a run time at 2128 seconds, indicating a 7.7% decrease in runtime. The efficiency was gained by redesigning the code in the implemented flow distribution function. A parallel processing redesign was also tried by using Matlabs parallel *for loop* functionality. However, this didn't yield any runtime reduction.

During the timing process it was also found that the flow distribution function, which is calculated at each time step, takes 83% of the total run time.

The runtime time to raster resolution dependency were investigated by modelling in the northern Lund area in 1x1 m resolution with $\Delta t = 0.25$ and in 2x2 m resolution with $\Delta t = 0.5$. For the 1x1 meter case the model required 58 hours of computation time and for the 2x2 meter case it took 7 hours, indicating about eight times more run time when the cell size and step size (Δt) are halved. This due to the double amount of steps in time and the four times as many cells to calculate, which multiplies to eight times as many computations. Assuming equal time for each computation, would thus estimate the 1x1 m case to $8 * 7\ h = 56$ hours of total runtime in theory. The recorded 58 hours are in the range of the roughly estimated 56 hours.

5 Discussion

This chapter is a discussion on the developed models strengths and weaknesses. It also touches upon how big areas the model can handle, the optimization issue and possible future research. The urban and rural test case results are described and discussed in the result chapter and here discussed only in regard to the general model behaviour. Further test case specific discussion is considered outside the scope of this thesis.

5.1 Evaluating the developed model

The model results presented in this work shows a downhill flow with an organic looking pattern in the rural (Abisko) case. These results look similar to the static drainage area estimations performed by the TFM and gives an indication that the converging and diverging runoff estimations from the static model are preserved in the dynamic model. In the rural test case small depressions and sinks in the landscape can get filled up with water, no prefilling of the DEM is needed. This is a strength in the sense that water will fill depressions at the rate that runoff reaches them. They will only fully flood if the incoming surface runoff and precipitation come at a higher intensity than the depression areas total infiltration.

Differentiating the infiltration rate for pervious and impervious areas, in the urban case, results in water flooding the streets and reaching water depths above 15 cm next to houses. This is what could be expected since the test area had had previous floods. Therefore, these results serve as an indication that this model, used in urban areas, very well could be of interest for validation and thereafter hopefully for future use in overland flow estimations.

The aim of intermediate model complexity could make the developed model a suitable setup for evaluating new storm water treatment and retention plans. Such evaluations could incorporate trying out e.g. new infiltration areas, weirs, curbstones and road bumps. Modelling features like these examples would then only require changes in the DEM, the infiltration and the friction raster.

This model doesn't incorporate what happens underground, in the storm sewer system and therefore it is advisable to use only when drainage system overflow effects aren't expected to be of major importance. However, the possibility of including the storm sewer inlets should be considered when it seems adequate. It is worth noticing that this study doesn't investigate when the use of inlets, assigned a constant capacity, is realistic enough. Therefore, a future investigation could be to compare this way of modelling storm sewer inlets with a plain subtraction of the rainfall intensity (like subtracting a 10-year return rain intensity from the input precipitation) which often is the recommended procedure (Mårtensson and Gustafsson, 2014).

For the flow direction estimation, the TFM is set up with a facet description to get a good enough flow distribution estimate. Then, when estimating what fraction of water will be moved in one direction during a time step, the water was not modelled to flow over these facets. Instead it was modelled as flowing over a plane with constant slope, from the focus cell center to the next cell center. Modelling water velocities and movement in individual facets would be a more detailed representation. Although this would require more memory and computations which would become more time consuming when modelling large areas in high resolution.

The developed model comprises effects of surface roughness, infiltration, precipitation, storm sewer inlets and outlets. It does incorporate mass preservation but not momentum preservation in the way it estimates velocities and routes water. The physical models instead have many more calculations, e.g. the mentioned momentum equations, evaporation, as well as more parameterized descriptions of the hydrological processes. An attempt at modelling with a more thorough physical approach that involves, and numerically solves, the momentum equation could be a future development.

The uniqueness of this model is that it uses the TFM algorithm for calculating flow directions. A topic for future research could be on what effects the use of different flow algorithms have on the results generated from dynamic models. How much such effects differ, when in rural or urban settings, could also be an in-depth investigation.

The main model results are water depths during a rainfall. An additional development could be to present results on water velocities at different times. Velocities could probably be of interest e.g. when investigating how hazardous a flooding is and when planning surface runoff paths in sustainable drainage systems. Velocities are already calculated for water leaving each cell, often it flows towards multiple neighbors. Results on velocities seem suitable to present in the same resolution as the water depths. If so is to be done, the incoming water velocities should probably be estimated as a vector sum of the water flowing into each cell.

To avoid extreme velocities in cells with very steep slope gradients, such as from building rooftops to ground, the β^*_{max} was introduced. This slows down water velocities in any steeper areas and at drops. Effects of this could be that the water depths on the roof edge gets slightly higher than they otherwise would, but greater depths will also increase the velocity estimated in the next time step. So no extreme depths are expected as long as β^*_{max} isn't set very close to zero. Although the effects of different values on β^*_{max} could be further investigated.

Another way to treat building to ground drops could have been to estimate the time it takes for water to fall to the ground but that would require additional data on where

the drops are situated. Since this model is partially justified by its simplicity this solution wasn't further investigated.

5.2 Model consistency

The model was expected to generate similar results, with different step sizes, as long as the velocity condition wasn't violated. The results at the end of the rainfall (T=30 min) showed a maximal water depth difference of 8 millimeters in one out of 22500 cells. The mean difference was very small and therefore the model is considered robust enough.

5.3 Resolution, runtime and future possibilities

The single runtime-to-resolution dependency test performed indicated roughly a factor eight runtime increase when moving from 2x2 m to 1x1 meter resolution. Thus the program behaves as expected in this single test. The time recorded modelling showed that an area of 225 ha, in 1x1 meter resolution, requires around 58 hours for the current implementation, on a standard laptop. Such runtimes are a bit behind the commercial, physically based, models (Svensson, 2016, September 6, personal communication). This also indicates that such areas, not much vaster, are feasible in the current implementation. As of now it seems very time consuming to model normal-sized Swedish cities in 1x1 meter resolution. However, possible gains of a thorough code optimization would probably change that.

The small attempt at code optimization gave some runtime reduction. Another, more time consuming optimization, could be to investigating if a vectorization of the Matlab code is possible. If so, that could avoid nested for-loops, and thereby reduce the runtime by orders of magnitude. If a vectorization turns out to be impossible, then efficient parallel processing methods, in a different programming language, may be the way forward. With more time and a thorough redesign of the code, big gains are probably possible since this model doesn't incorporate momentum and do not have to solve complex differential equations, as the fully physical models do. Therefore, it is plausible that the new model could get runtimes below those of the PHB models if implemented in a more optimal way. Although, if the flow direction algorithm in TFM turns out to be much more demanding (computationally) than those used in PHB models, it may not be possible.

Another idea regarding efficient coding is e.g. calculating the flow directions from cells with very little water. An easy thing to implement could be to enable a setting to not calculate water flow out of cells with less than a user set water depth. An example could be not to make flow calculations for cells with less than 1 mm water depth. This would most probably make a big difference when modelling scenarios with steep and/or dry areas.

Due to lack of time no actual comparison was done on how flooded areas and water depths differ when the model is run in different resolutions. For future research it would be interesting to investigate what differences are found when running the model in 1x1, 2x2 or 4x4 meter. A high resolution might make a bigger difference when modeling urban areas which have manmade obstacles such as houses, street curvature and curbstones. This is also of interest because many Swedish municipalities use 4x4 meter (PHB model) investigation results as a basis for making expensive investments in city planning and storm water retention. If high resolution yields different results, then such results could have led to decisions on different flood prevention actions. On the other hand, if the differences due to resolution choice are small, then, in regard to this new model, the need for modeling in high resolution isn't as big. In that case the need for optimization and large runtime reductions are also less. In that case the model could be setup and run in a coarser resolution with a good enough result.

The runtime can be expected to at least double with a doubled modelling area since the number of calculations are doubled in each time step. The computers working memory limitations could at some point change this increased area-to-runtime relation by requiring more than doubled runtime, or even stop the program from running, due to running out of memory. These questions on implementation level has to be researched depending on implementations and model scenarios.

Another important thing to keep in mind is that running the model in areas with steep slopes or hard surfaces, with low friction coefficients (n), generates higher velocities. This requires smaller steps in time and therefore longer runtime. Hence a rural landscape, often with less slope and higher surface friction, can be modelled with fewer time steps (bigger step size) for the same raster size and scenario time.

A smaller time step is safer in regard to the velocity condition but also means more steps to be calculated and thus more program runtime. The choice of investigation area, resolution and time step size should be carefully considered in order to avoid too long program runtime.

6 Conclusions

The developed model is a new tool for estimating where and when there is a high likelihood of flooding due to heavy rain. It operates by handling input data in a GIS file format and generating a direct output in GIS files. The user can define spatially varying precipitation, infiltration and surface friction. The new model framework handles more input data and parameters than EFR models although being less complex than the PHB models and therefore it meets the aim of having an intermediate complexity design.

Evaluating the new model, with regard to the thesis objectives, it is clear that EFR models can be used when (1) setting up a model that incorporates the time dimension as well as (2) infiltration, surface friction and (3) the storm sewer inlets and outlets. An additional gain with this model is that the DEM pre-filling of depressions, used for drainage accumulation estimations with EFR models, is no longer needed. The last objective (4), stating the intention to investigate the possibility for this model to operate reasonably fast in high resolutions, has only been given minor work due to lack of time. The current program implementation is not optimized and therefore it is yet unknown how fast an optimal implementation could be. To summarize, the first three thesis objectives have been reached and the fourth, additional objective, needs more work.

This model is more realistic than the present static EFR models since it is dynamic and incorporates more hydrological aspects. It incorporates the TFM algorithm that has proven a good flow estimation accuracy. Therefore, the new model deserves to be a considered choice when possible. However, before extensive use in research, or commercial investigations, a model validation against real world measurements would be appropriate.

7 References

7.1 Bibliography

BURSTRÖM, H., BONDPÄ, B. E. & NÄSLUND, L. 2014. *Regnoväder orsakade kaos i Skåne* [Online]. SVT Web: TT. Available: http://www.svt.se/nyheter/lokalt/skane/hundratals-larm-efter-ovader-i-skane [Accessed January 23 2017].

COURANT, R., FRIEDRICHS, K. & LEWY, H. 1967. On the partial difference equations of mathematical physics. *IBM journal,* 11**,** 215-234.

DHI 2015. MIKE 21 Flow Model: Hydrodynamic Module User Guide.

ERIKSSON, J. 2015. Ett år efter översvämningarna i Malmö. *Sydsvenskan*, 27 August 2015.

ESRI. n.d. *How flow direction works* [Online]. Available: http://desktop.arcgis.com/en/arcmap/10.3/tools/spatial-analyst-toolbox/how-flow-direction-works.htm [Accessed 2016-09-13].

FLETCHER, T. D., ANDRIEU, H. & HAMEL, P. 2013. Understanding, management and modelling of urban hydrology and its consequences for receiving waters: A state of the art. *Advances in Water Resources,* 51**,** 261-279.

FREEMAN, T. G. 1991. Calculating catchment area with divergent flow based on a regular grid. *Computers & Geosciences,* 17**,** 413-422.

HALLEN, G. 2016. *Porous asphalt as a method for reducing urban storm water runoff in Lund, Sweden.* Master Thesis nr 407, Lund University, Physical Geography and Ecosystem Sciences.

HARRIE, L. 2013. *Geografisk informationsbehandling: teori, metoder och tillämpningar,* Lund, Studentlitteratur AB.

HASAN, A., PILESJÖ, P. & PERSSON, A. 2012a. Drainage area estimation practice, how to tackle artefacts in real world data. *GIS Ostrava.* Ostrava.

HASAN, A., PILESJÖ, P. & PERSSON, A. 2012b. On generating digital elevation models from liDAR data – resolution versus accuracy and topographic wetness index indices in northern peatlands. *Geodesy and Cartography,* 38**,** 57-69.

HENONIN, J., RUSSO, B., MARK, O. & GOURBESVILLE, P. 2013. Real-time urban flood forecasting and modelling -- a state of the art. *Journal of Hydroinformatics,* 15**,** 717-736.

HINGRAY, B., PICOUET, C. & MUSY, A. 2014. Principles Of Hydrological Modeling. *Hydrology.* CRC Press.

IPCC 2014. *Climate Change 2014: Impacts, Adaptation, and Vulnerability. Part A: Global and Sectoral Aspects. Contribution of Working Group II to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Field, C.B., V.R. Barros, D.J. Dokken, K.J. Mach, M.D. Mastrandrea, T.E. Bilir, M. Chatterjee, K.L. Ebi, Y.O. Estrada, R.C. Genova, B. Girma, E.S. Kissel, A.N. Levy, S. MacCracken, P.R. Mastrandrea, and L.L. White (eds.)],* Cambridge, United Kingdom and New York, NY, USA, Cambridge University Press.

JENSON, S. K. & DOMINGUE, J. O. 1988. Extracting topographic structure from digital elevation data for geographic information system analysis. *Photogrammetric engineering and remote sensing,* 54**,** 1593-1600.

LANTMÄTERIET. n.d. *GSD-Höjddata, grid 2+* [Online]. Available: https://www.lantmateriet.se/sv/Kartor-och-geografisk-information/Hojddata/GSD-Hojddata-grid-2/ [Accessed 13 of May 2016].

LINDHER, F. 2015. Klimatsäkrat Malmö:
Kraftigt skyfall med risk för översvämning. *Granskningsrapport.* Malmö Stad, Stadsrevisionen.

LINDSAY, J. B. & CREED, I. F. 2005. Sensitivity of digital landscapes to artifact depressions in remotely-sensed DEMs. *Photogrammetric Engineering & Remote Sensing,* 9**,** 1029-1036.

MÅRTENSSON, E. & GUSTAFSSON, L.-G. 2014. Kartläggning av skyfalls påverkan på samhällsviktig verksamhet - Framtagande av metodik för utredning på kommunal nivå. Myndigheten för samhällskydd och beredskap (MSB).

O'CALLAGHAN, J. F. & MARK, D. M. 1984. The extraction of drainage networks from digital elevation data. *Computer Vision, Graphics, and Image Processing,* 28**,** 323-344.

OLSSON, J. & FOSTER, K. 2013. Extrem korttidsnederbörd i klimatprojektioner för Sverige. *SMHI Klimatologi,* 6.

PILESJÖ, P. & HASAN, A. 2014. A Triangular Form-based Multiple Flow Algorithm to Estimate Overland Flow Distribution and Accumulation on a Digital Elevation Model. *Transactions in GIS,* 18**,** 108-124.

PILESJÖ, P. & ZHOU, Q. 1997. Theoretical estimation of flow accumulation from a grid-based digital elevation model. *Proceedings of GIS AM/FM ASIA'97 and Geoinformatics'97 Conference***,** 447-456.

SVANSTRÖM, S. 2015. Urbanisering – från land till stad. *Välfärd,* 2015:1.

SVENSSON, G. 2016. *RE: Interview with assistant supervisor.* Type to NILSSON, H.

SYLVÉN, J. & EKELUND, T. 2015. Skyfallskartering i GIS. http://www.lansstyrelsen.se/jonkoping.

VA SYD 2012. Åtgärdsplan för Lunds avlopp.

WARD, A. D. & ELLIOT, W. J. 1995. *Environmental hydrology*, CRC Press LLC.

ZHOU, Q., PILESJÖ, P. & CHEN, Y. 2011. Estimating surface flow paths on a digital elevation model using a triangular facet network. *Water Resources Research,* 47.

7.2 Data Acknowledgement

7.2.1 LiDAR data for the Abisko test area

The funding of the LIDAR survey and DEMs created from it came from a number of agencies through a partnership of researchers. We acknowledge the contributions of the Natural Sciences and Engineering Research Council of Canada, Discovery Grant to Nigel Roulet (McGill University); The Abisko Scientific Research Station supported at the time by the Royal Swedish Academy of Sciences, KVA; Patrick Crill (Stockholm University) by the Swedish Research Council, VR; Torben R. Christensen (Lund University) by the Swedish Research Council, VR; Håkan Olsson (Swedish University of Agricultural Sciences) by the Swedish Environmental Protection Agency; and Andreas Persson's and Petter Pilesjö's research grants at the Lund University GIS Centre.

7.2.2 LiDAR data for Gästgivarevägen in Lund

The funding of the LiDAR survey came from Lund Municipality. We acknowledge that this data has been made available, it is an important contribution in making this thesis work possible.

8 Appendix

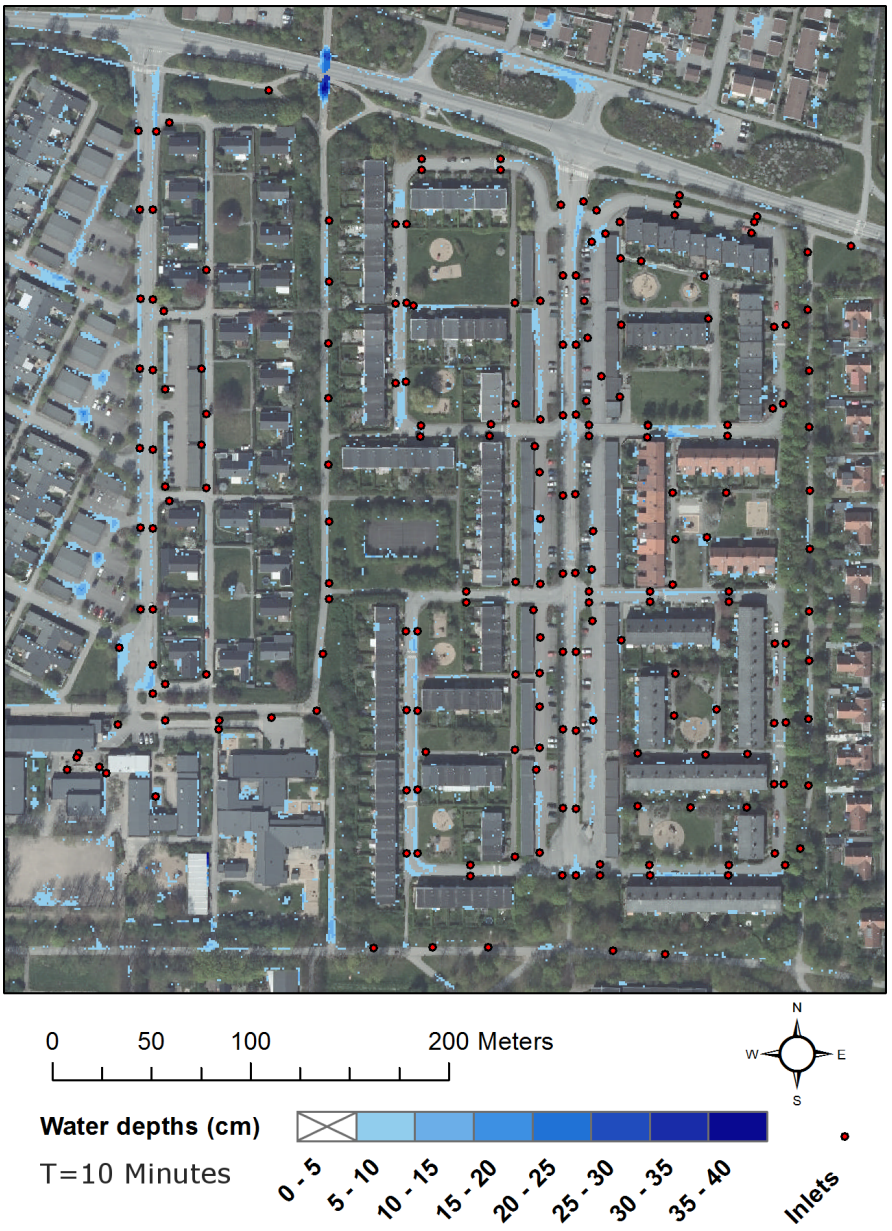

Figure 23. Water levels after 10 minutes of precipitation over Gästgivarevägen in Lund. Water levels above 5 cm are shown in blue against the background image. Levels below 5 cm have been exempt. The precipitation was set to 89 mm/h, infiltration in pervious areas to 35 mm/h, manning friction to 0.5 in pervious areas (grass) and 0.02 on impervious areas (roofs and asphalt). The max slope was set to 11.3° and the storm sewer inlets (the red dots) where given a capacity of 3.64 litres per second. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

Figure 24. Water levels after 20 minutes of precipitation over Gästgivarevägen in Lund. Water levels above 5 cm are shown in blue against the background image. Marked in the figure is (1) a flooded turnaround area and (2) two dark blue cells that are flooded artefact roof sinks. The model settings are explained in the previous figure. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

Figure 25. Water levels after 30 minutes of precipitation over Gästgivarevägen in Lund. Water levels above 5 cm are shown in blue against the background image. Marked out are (1) 10-15 cm depth next to houses, (2) 25-30 cm next to garages and (3) 10-30 cm depths in the turnaround area. The settings are explained in Figure 23. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

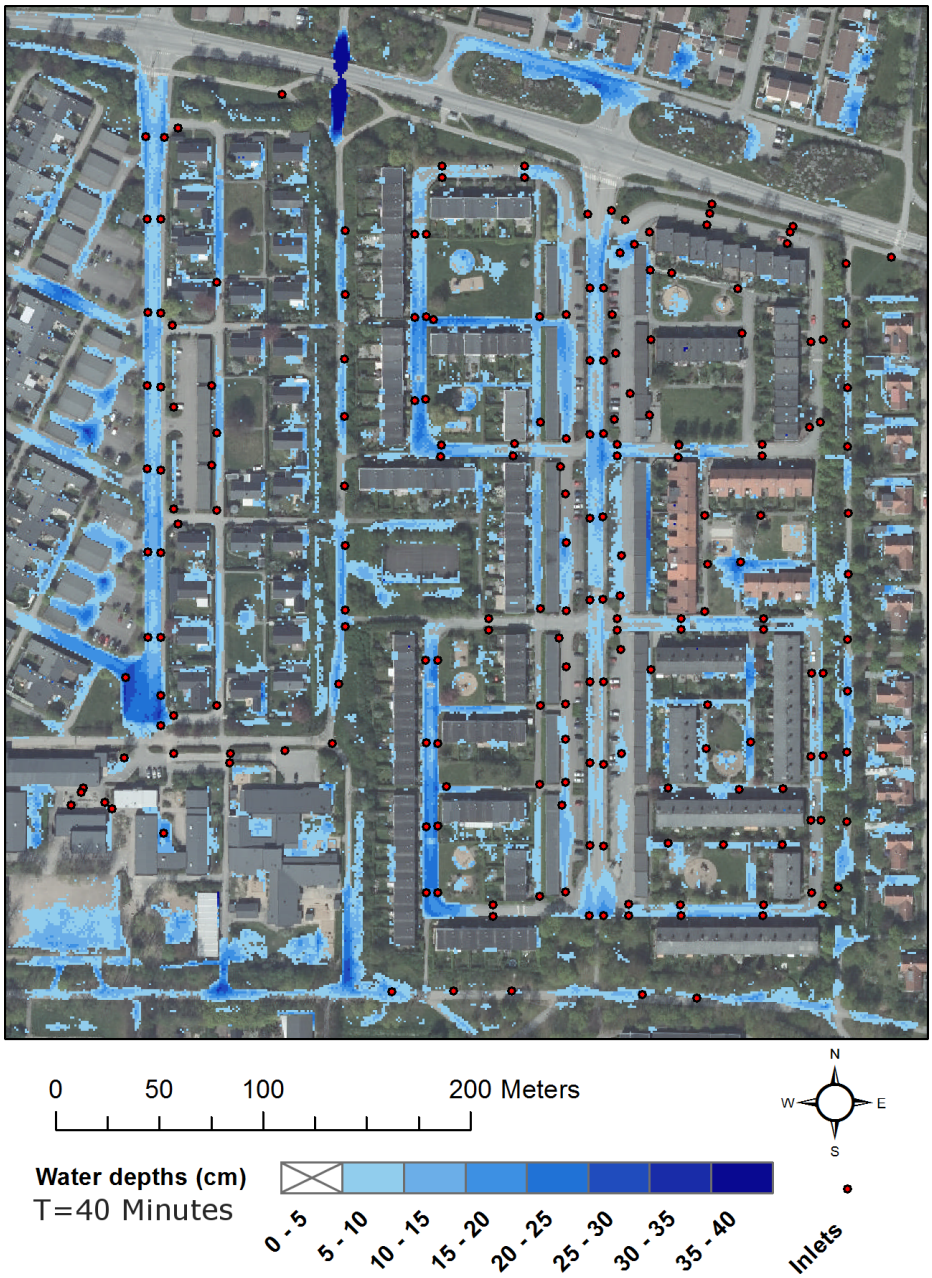

Figure 26. Water depths at T=40 minutes. 10 minutes has passed since the rainfall stopped and now the water is just flowing and infiltrating in permeable areas and into the storm sewer inlets. The settings are explained in Figure 23. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

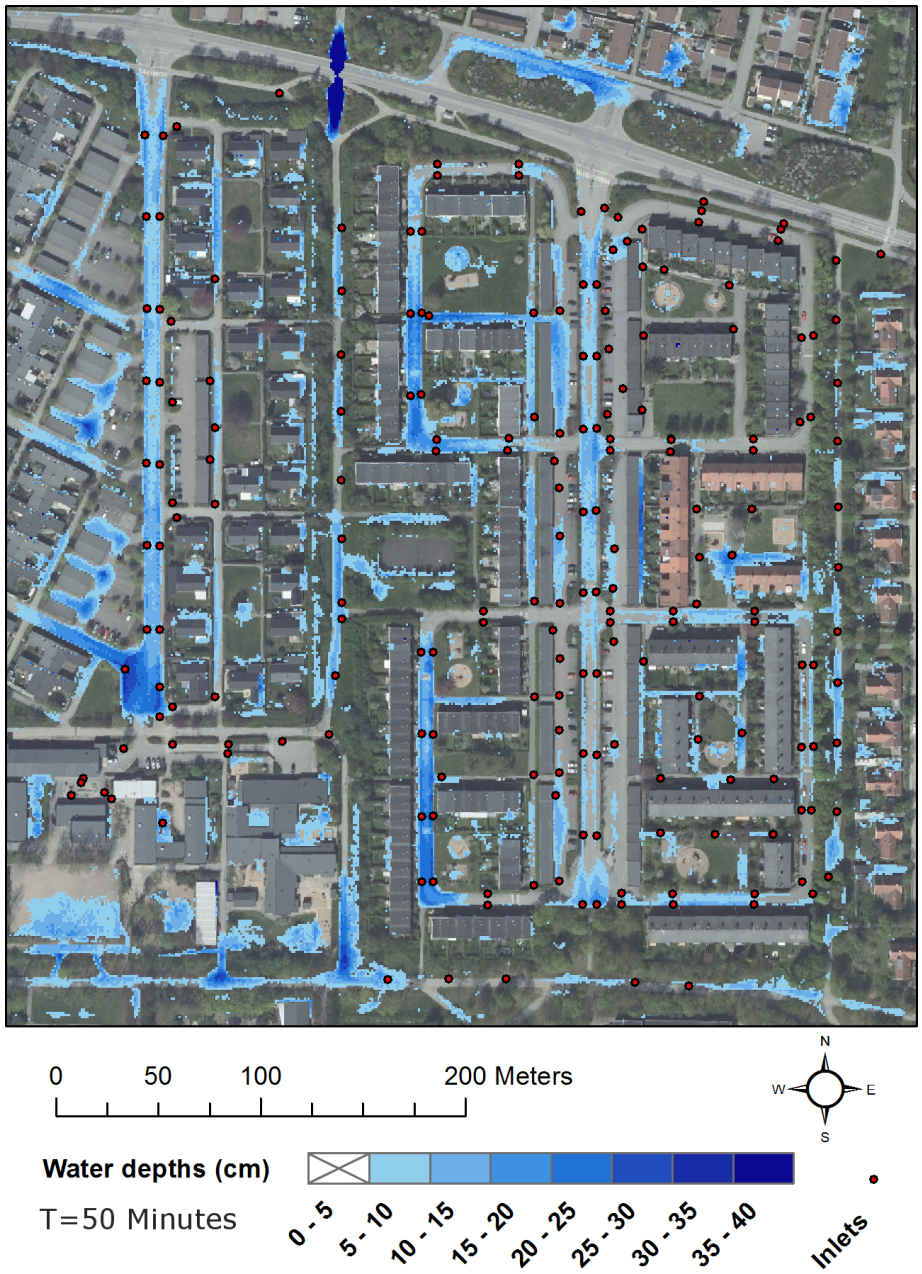

Figure 27. Water depths at T=50 minutes. 20 minutes has passed since the rainfall stopped and now the water is just flowing and infiltrating in permeable areas and into the storm sewer inlets. The settings are explained in Figure 23. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).

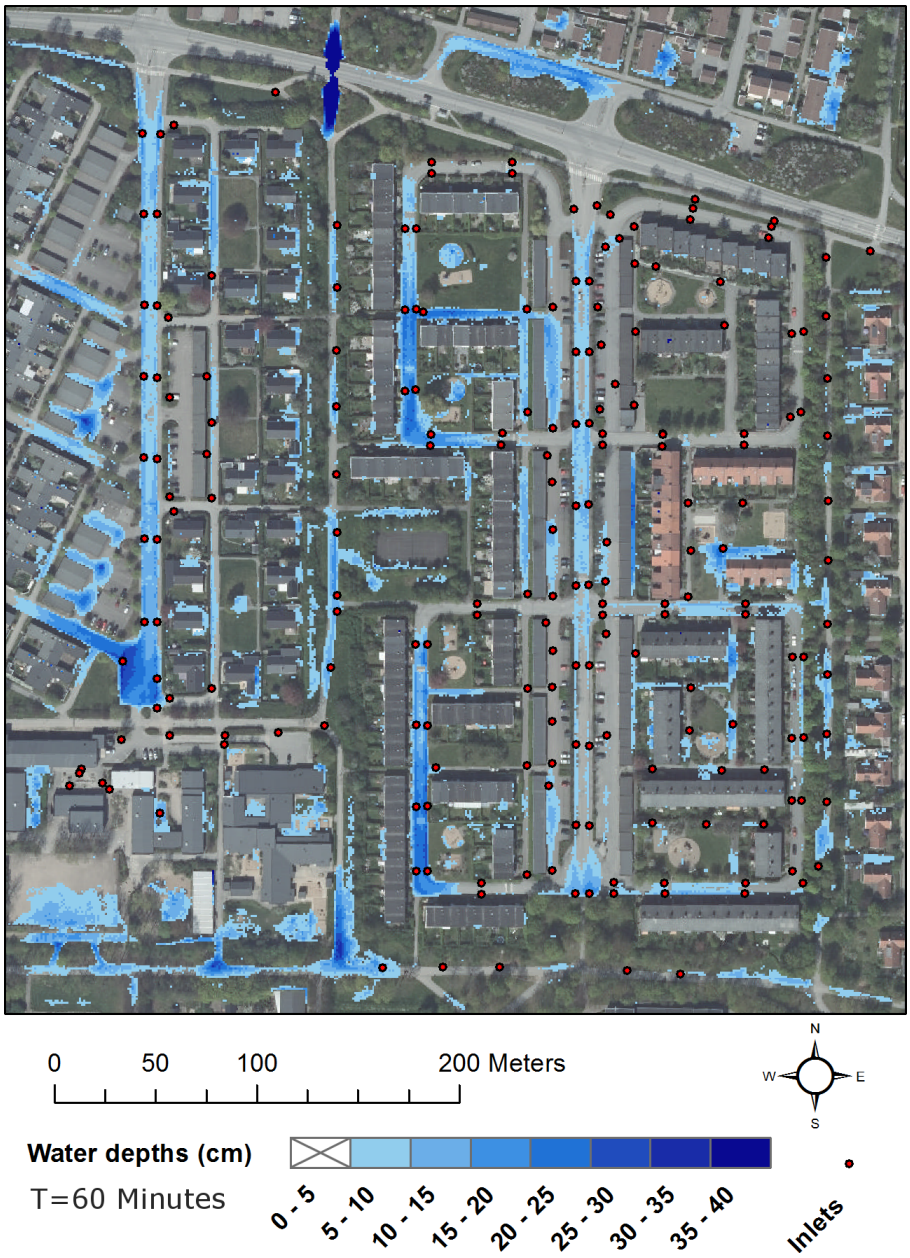

Figure 28. Water depths at T=60 minutes. 30 minutes has passed since the rainfall stopped and now the water is just flowing and infiltrating in permeable areas and into the storm sewer inlets. The settings are explained in Figure 23. GSD-Ortophoto, 0.5 m color © Lantmäteriet (2014).