

UPTEC W 06 004  
ISSN 1401-5765

Examensarbete  
M.Sc. Thesis Work

# Dynamisk modellering av en dagvattendamm

Dynamic Modelling of a Storm Water Dam

KLARA LARSSON  
Februari 2006



## Referat

### Dynamisk modellering av en dagvattendamm

*Klara Larsson*

Dagvatten är det regn- eller smältvatten som ytavrinner från mark eller byggnader. Dagvattnets miljöpåverkan har på senare år uppmärksammats mer då större krav har ställts på en mer effektiv och ekonomisk rening.

I detta arbete har en dynamisk modell av vattenströmningen i Vattenparken, en dagvattendamm i Borlänge, konstruerats. Modellen har byggts upp i Matlab 7.0 utifrån Shallow Water Ekvationerna (SWE), vilka är en förenkling av Navier Stokes Ekvationer. SWE beskriver en tvådimensionell strömning i en inkompressibel fluid. Vid användning av SWE antas det att vattnets strömningshastighet är lika stor på alla djup i vattnet för varje lägeskoordinat. I modellen har SWE differentierats med en semiimplicit metod, Crank-Nicholson.

Vattenparkens bottenpografi har interpolerats fram i ArcMap, ett GIS-verktyg. Strömningshastigheten har sedan iterativt beräknats för olika inflödeshastigheter. Sedimentprov togs från olika delar i dammen och utifrån strömningshastigheterna kunde sedimentationsegenskaperna i olika delar bestämmas. Modellen blev dock inte helt stabil. Resultatet av simuleringarna kunde därför inte användas på det sätt som var tänkt. För att få visa på något resultat utfördes ändå sedimentationsanalysen på simuleringar som avslutats innan divergens uppstod. Dessa resultat kunde ej tolkas som vetenskapligt korrekta, men visade att det inte sedimenterade i dammen för någon inströmningshastighet. Detta berodde till största del på till storleken litet partikulärt material i det inkommande vattnet som lätt transporteras vidare eller resuspenderas från botten.

För att förbättra sedimentationsegenskaperna beräknades även strömningshastigheterna för olika ändringar på dammen. Vallar lades in för att förlänga uppehållstiden i dammen. Detta resulterade inte i någon förbättring av dammens sedimentationsegenskaper.

Nyckelord: Dagvatten, Shallow Water Ekvationerna, Crank-Nicholson, vattenströmning, sedimentation, Matlab, ArcMap

## **Abstract**

### **Dynamic Modelling of a Storm Water Dam**

*Klara Larsson*

Storm water is melted snow or rainwater that runs off from buildings or the ground. The effect of storm water on the environment has been more noticed lately due to higher demands on more efficient and economic purification.

In this thesis, a dynamic model of water streaming in Vattenparken, a storm water dam in Borlänge, Sweden, has been constructed. The model was built in Matlab 7.0 on the basis of the Shallow Water Equations (SWE). SWE describe the streaming water of an incompressible fluid in two dimensions and are a simplification of the Navier Stokes Equations. In SWE it is assumed that the streaming velocity of the water is constant with depth for each position in the dam. In the model, SWE were differentiated using the Crank-Nicholson scheme, which is an implicit numeric method.

The bottom topography of Vattenparken was interpolated in ArcMap. The streaming velocities were then calculated iteratively for different flows at the inlet of the dam. Sediment samples were taken from different parts of the dam. Based on the streaming velocities, the sedimentation properties on different locations in the dam could be decided. The model was not stable, however, and the result of the simulations could not be used as planned. Even though the model was not stable, the sedimentation analysis was still done on simulations finished before divergence occurred. These results could not be considered scientifically correct but showed that sedimentation did not occur for any of the streaming velocities at the inlet. This was mostly due to very small particular material in the incoming water. The particles were easily transported in the water or resuspended from the bottom.

To improve the sedimentation properties, the streaming velocities were also calculated for three changes on the bottom topography of the dam. Banks were constructed to prolong the retention time. This did not result in any improvement of the sedimentation capacity.

Keyword: Storm water, Shallow Water Equations, Crank-Nicholson, water flow, sedimentation, Matlab, ArcMap

Uppsala University  
Department of Information Technology  
Box 337  
SE-75105 Uppsala  
Sweden  
ISSN 1401-5765



## Förord

Detta examensarbete omfattar 20 högskolepoäng och ligger inom civilingenjörsprogrammet Miljö- och vattenteknik vid Uppsala Universitet. Arbetet har utförts på uppdrag av *IVL Svenska Miljöinstitutet AB* i Stockholm. Arbetet syftar till att simulera vattenströmningen i Vattenparken, en dagvattendamm i centrala Borlänge, samt att analysera dammens sedimentationsegenskaper. För att optimera sedimentationen i dammen har vattenströmningen även simulerats för olika varianter av förändringar på dammen. Examensarbetet ingår i projektet *Dagvatten i urban miljö* (DUM) som pågår mellan *IVL, Borlänge Energi* och *Vägverket*.

Handledare på IVL har varit Magnus Rahmberg, civilingenjör. Ämnesgranskare på universitetet har varit Jarmo Rantakokko, universitetslektor vid Institutionen för informationsteknologi, Teknisk databehandling. Examinator har varit Allan Rodhe, professor i hydrologi, Institutionen för geovetenskaper, Luft- och vattenlära.

Jag vill tacka följande för hjälp under examensarbetet: Magnus Rahmberg, min handledare på IVL, för hjälp och diskussioner kring allt som rör arbetet. Jarmo Rantakokko, Uppsala Universitet, för hjälp kring modellbygge och numeriska metoder. Christian Junestedt, IVL, för svar på frågor rörande dammen och trevligt sällskap på resorna till Borlänge. Leif Johansson, Borlänge energi, för all möjlig information på besöken i Borlänge. Annika Ryegård, IVL, för hjälp med interpolering av bottenpografi i ArcMap. Ann-Sofie Allard och Brita Dusan, IVL, för hjälp vid torkning och analys av sedimentprover. Slutligen vill jag tacka min rumskamrat Frida Pollak för trevlig samvaro uppe i exjobbsrummet.

# Innehållsförteckning

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>INLEDNING</b> .....  | <b>2</b>  |
| 1.1      | SYFTE.....  | 2         |
| 1.2      | BAKGRUND – DAGVATTEN.....   | 3         |
| 1.2.1    | <i>Dagvattenanläggningar i Sverige</i> .....  | 3         |
| 1.2.2    | <i>Dagvattendamm i Borlänge</i> .....   | 4         |
| <b>2</b> | <b>TEORI</b> .....  | <b>6</b>  |
| 2.1      | SHALLOW WATER EKVATIONERNA .....  | 6         |
| <b>3</b> | <b>METODER</b> .....  | <b>10</b> |
| 3.1      | INTERPOLERING AV BOTTENTOPOGRAFI.....   | 10        |
| 3.2      | MODELLUPPBYGGNAD .....  | 11        |
| 3.2.1    | <i>Numeriska metoder</i> .....  | 11        |
| 3.2.1.1  | Explicit differentiering .....  | 12        |
| 3.2.1.2  | Semi-Implicit differentiering.....  | 12        |
| 3.2.1.3  | Stabilitetsanalys .....   | 16        |
| 3.3      | IMPLEMENTERING.....   | 17        |
| 3.4      | RANDVILLKOR .....   | 18        |
| 3.4.1    | <i>Inlopp</i> .....   | 19        |
| 3.4.2    | <i>Utlopp</i> .....   | 19        |
| 3.4.3    | <i>Öar och strandkanter</i> .....   | 19        |
| 3.5      | SIMULERING AV VATTENSTRÖMNING .....   | 20        |
| 3.6      | SEDIMENTATIONSANALYS .....  | 22        |
| 3.6.1    | <i>Metod 1 - Hjulströms diagram</i> .....   | 22        |
| 3.6.2    | <i>Metod 2 – Beräkning av sedimentationshastighet</i> .....                           | 24        |
| 3.7      | MANIPULATION AV BOTTENTOPOGRAFI.....  | 25        |
| <b>4</b> | <b>RESULTAT</b> .....   | <b>27</b> |
| 4.1      | RESULTAT METOD 1 – HJULSTRÖMS DIAGRAM .....   | 28        |
| 4.2      | RESULTAT METOD 2 – BERÄKNING AV SEDIMENTATIONSHASTIGHET.....                          | 29        |
| <b>5</b> | <b>DISKUSSION OCH SLUTSATSER</b> .....  | <b>30</b> |
| <b>6</b> | <b>REFERENSER</b> .....   | <b>32</b> |
|          | <b>BILAGA 1 – RESULTAT AV METOD 1 – HJULSTRÖMS DIAGRAM</b> .....                      | <b>35</b> |
|          | <b>BILAGA 2 – RESULTAT AV METOD 2 –BERÄKNING AV<br/>SEDIMENTATIONSHASTIGHET</b> ..... | <b>36</b> |
|          | <b>BILAGA 3 – MATLABFILER</b> .....   | <b>37</b> |
|          | <b>BILAGA 4 – HUVUDKOD</b> .....  | <b>38</b> |
|          | <b>BILAGA 5 - UNDERKOD</b> .....  | <b>41</b> |

# 1 INLEDNING

## 1.1 SYFTE

Dagvatten kallas det regnvatten eller smältvatten som ytavrinner från byggnader eller mark i tätbebyggda områden. Idag är stora ytor i Sverige bebyggda eller täckta med sten eller asfalt vilket genererar betydande mängder dagvatten. Dagvattnets miljöpåverkan har på senare år uppmärksammats och större krav har ställts på en mer effektiv och ekonomisk rening. Den ökade uppmärksamheten kring dagvatten beror till stor del på att de industriella utsläppen renas allt mer vilket gjort att dagvattnets bidrag till miljöföroreningar har ökat.

Vattenparken är en dagvattendamm som ligger i centrala Borlänge. Dammen är en sedimentationsdamm och behöver därför muddras när mängden sedimenterat material överskrider önskad nivå. Problemet med Vattenparken är att den aldrig har behövt muddras (personligen Johansson). Det sedimenterar alltså inte i dammen som det är tänkt. Vid stora flöden resuspenderas material från botten och föroreningar som binder till partiklar i vattnet rinner därför vidare ut genom dammen och in i Lusbäcken, som rinner nedströms dammen. Mätningar som gjorts visar att mängden suspenderat material är mindre i utgående flöde än i ingående flöde under normala förhållanden. Vid extrema förhållanden spolas dock material som sedimenterat ut igen (personligen Junestedt).

För optimal sedimentation ska vattnet flöda långsamt och över en lång sträcka. De största, tyngsta partiklarna sedimenterar närmast inloppet och de minsta, lättaste partiklarna sedimenterar sist. Flödar vattnet för snabbt hinner inte de minsta partiklarna sedimentera och rinner därför ut i Lusbäcken. En lång uppehållstid i dammen är därför nödvändig för optimal sedimentation.

Syftet med examensarbetet är att bygga en modell som simulerar vattnets flöde genom dammen. För att optimera sedimentationen i dammen ska vattenflödet även simuleras för olika förändringar av dammen såsom att göra den djupare på vissa ställen eller att bygga vallar i vissa delar. Dessa ändringar är tänkt att tvinga vattnet att ta en längre väg genom dammen och därmed förlänga uppehållstiden. Matlab 7.0 ansågs som en bra programvara att använda vid konstruerandet av modellen.

Ekvationerna som ska stå till grund i modellen är Shallow Water Ekvationerna. Det är väl beprövade ekvationer vid flödessimulering och har använts tidigare i projekt vid bland annat Universidad Autónoma de Madrid, Spanien och School of Science i Shaanxi, Kina, med goda resultat (Project Euklid, 2005 & Science Direct, 2005).



## 1.2 BAKGRUND – DAGVATTEN

### 1.2.1 Dagvattenanläggningar i Sverige

Dagvatten från vägar och andra hårdgjorda ytor i stadsmiljön innehåller ofta stora mängder metaller och organiska ämnen. Föroreningar som identifierats och som vanligen analyseras i dagvatten är bland annat suspenderat material, BOD, COD, närsalter, metaller och kolväten (Färm, 2003). Dessa föroreningar riskerar att kontaminera marken, grundvattnet, sjöar och vattendrag. Dagvattnet leds antingen direkt ut i recipienten utan rening eller via reningsverk och sedan vidare till recipienten. Problem med föroreningar i dagvatten är omfattande i kommunala reningsverk. Dagvattnet utgör i vissa fall en stor del av mängden avloppsvatten. Föroreningar från dagvattnet binds hårt till slammet i reningsverken. Optimalt vid vattenrening är att slammet återanvänds som till exempel gödsel. Förorenat slam är dock både svårt att använda och att deponera på grund av höga metallhalter.

Den del av dagvattnet som renas men som inte hamnar i kommunala reningsverk behandlas i olika typer av dagvattenanläggningar. Reningen av vattnet i dessa anläggningar sker mer eller mindre naturligt innan vattnet når recipienten. Även våtmarker används på vissa platser som reningsanläggning.

De största bidragen till förorenat dagvatten kommer från vägnätet. Inom det svenska vägnätet finns det cirka 400 dagvattenanläggningar av olika typer, som delas in i fyra huvudgrupper (Lindgren & Svensson, 2003).

- Sedimenteringsdamm/fördröjningsdamm: Partikulära föroreningar avskiljs genom sedimentation
- Översilningsyta: Vegetationsklädd våtmark dit vägdagvattnet leds.
- Infiltrationsanläggning: Anläggning där vägdagvattnet lagras temporärt för infiltration i marken.
- Avsättningsmagasin: En sluten anläggning som vid regn fylls med vägdagvatten. När magasinet är fullt stängs det och sedimentation sker i ostört vatten.

Dominerande bland dagvattenanläggningarna är sedimenteringsdammar/fördröjningsdammar. De utgör cirka 70% av det totala antalet anläggningar i Sverige (Lindgren & Svensson, 2003). Det är även vanligt förekommande att kombinera flera anläggningar för att få en så optimal rening som möjligt. Valet av dagvattenanläggning baseras ofta på vägarnas trafikintensitet.

Brister i drift och underhåll av dagvattenanläggningar har uppmärksammats (Lindgren & Svensson, 2003). Ofta saknas kunskap om dammarnas funktion såsom skötselrutiner och kunskap om driftstörningar vid flödesvariationer. Detta kan orsaka stora problem. Det är viktigt att en helhetssyn skapas för varje dagvattenanläggning och att det finns en genomtänkt plan ända från projektering till investering. I denna plan ska även drift och underhåll vara väl dokumenterat.

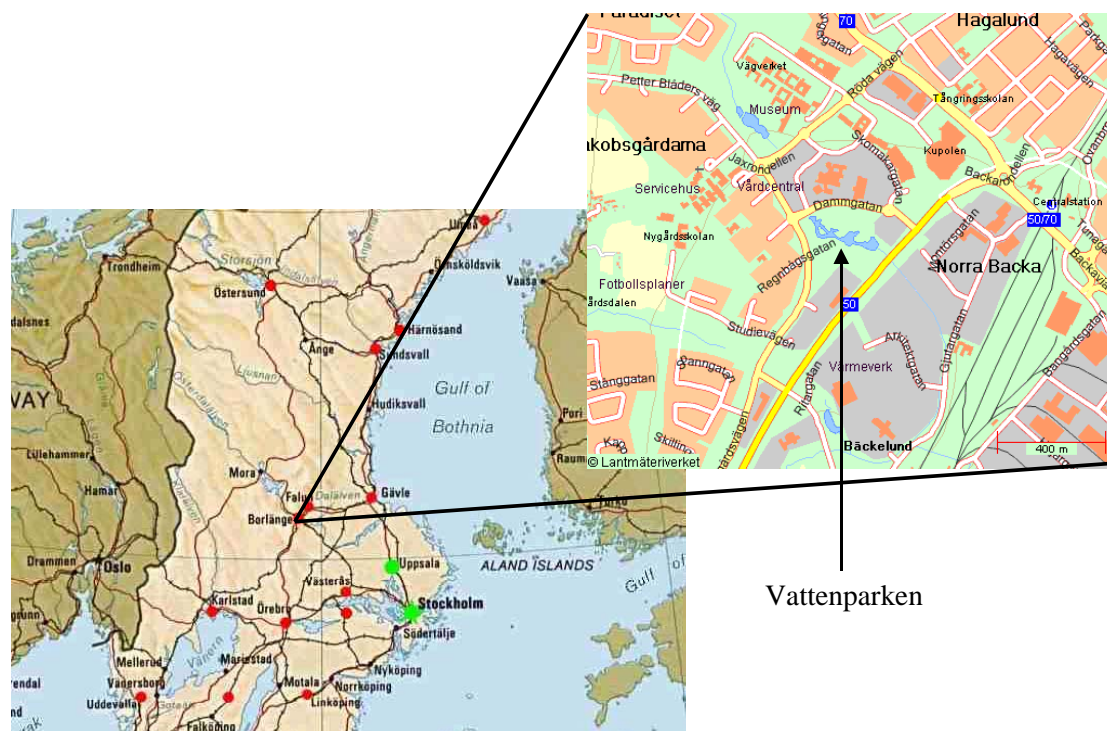
## 1.2.2 Dagvattendamm i Borlänge

I slutet på 1990-talet beslutade Borlänge kommun att området kring Högskolan, Teknikdalen och Vägverket i centrala Borlänge skulle byggas om. Syftet var att locka nya studenter och företag till området genom att göra det till en attraktiv och kreativ miljö (Borlänge energi, 2003). På grund av detta anordnades därför 1998 en arkitekt-tävling där förslaget *Korspunkter* konstruerat av Arken Arkitekter vann. Arbetet med att bygga om parker, gator och torg i området påbörjades 2001 och avslutades 2003.

Vattenparken, som dammen heter, ligger i Framtidsdalen och ingick i arkitektförslaget *Korspunkter*. Huvudsyftet med dammen är estetiskt. Den ska fungera som rekreationsområde för allmänheten. Dammen har ytterligare ett ändamål vilket är att rena dagvatten genom sedimentationsprincipen. Lusbäcken, som rinner nedströms dammen är förorenad av stora dagvattenutsläpp (Borlänge energi, 2003). Dammen hjälper därför till med eventuell rening av dagvattnet innan det når bäcken. Tillflödet till dammen utgörs av flera olika källor av dagvatten från Borlänge stad vilka några listas nedan.

- Bostadsområden, bland annat Jakobsgårdarna
- Parkeringsytor, bland annat ICA Maxi
- Riksväg 70

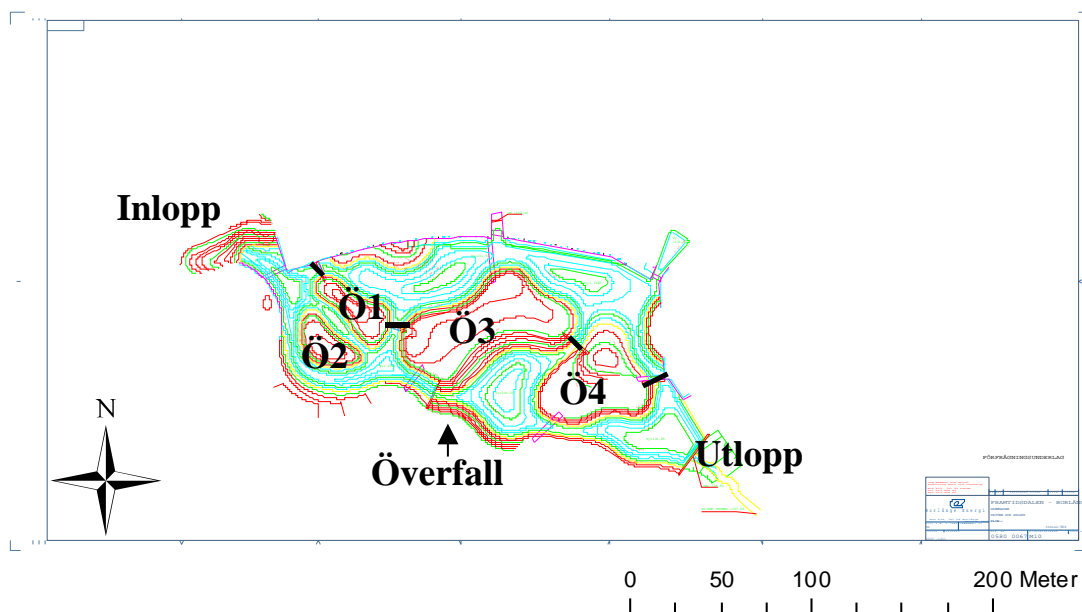
Dammens flödesvariationer är mycket stora under året. Vid mätningar har flödet varierat mellan cirka 5 l/s och 5 m<sup>3</sup>/s (personligen Junestedt).



Figur 1. Lokalisering av Vattenparken i Borlänge (Uppsala Universitet, 2006 och Eniro Sverige AB, 2005).

Vattenparken är cirka 300 meter lång och cirka 100 meter bred. Djupet varierar men är maximalt cirka 2 meter. I dammen finns fyra mindre öar, vilka är markerade som Ö1-Ö4 i Figur 2 nedan. Växtligheten kring dammen är riklig. Vissa delar av strandzonen är anlagd med vattenälskande växter på färdigvuxna strandmattor. Andra delar är anlagda med kokosnät som är till för att stabilisera strandkanten. De flesta växter som anlagts finns vilt i den svenska naturen, men som experiment har även växter som normalt inte ska klara sig så långt norrut anlagts (Borlänge energi, 2003). Under 2003 släpptes även ett antal spegelkarpar ut i dammen.

Efter ungefär halva dammen, mellan Ö3 och den nedre dammväggen finns ett överfall, det vill säga, vatten kan rinna nedströms över överfallet men inte uppströms. Överfall finns även mellan Ö3 och Ö4, mellan Ö4 och den högra väggen samt vid utloppet. Detta medför att dammen är uppdelad i två delar. En övre del som bara existerar i rekreationssyfte där vattnet är stillastående och en nedre del med strömmande vatten mellan inlopp och utlopp som både har ett rekreations- och ett renande syfte. Inget dagvatten passerar därför den övre delen. Figur 2 visar Vattenparken med höjdkonturer mot ett lokalt referenssystem. Inloppet ligger uppe till vänster i dammen och utloppet längst ner till höger. Området mellan Ö1 och dammens kant är avstängt vilket medför att inget vatten kan passera där emellan. Detsamma gäller för området mellan Ö1 och Ö3. Svarta markeringar visar avdelningen mellan övre och nedre del.



Figur 2. Vattenparken i Borlänge.

## 2 TEORI

### 2.1 SHALLOW WATER EKVATIONERNA

För att beräkna strömningar i ett grunt vatten kan de så kallade Shallow Water Ekvationerna (SWE) användas. SWE beskriver en tvådimensionell strömning i en inkompressibel fluid. SWE består av ett icke-linjärt ekvationssystem av partiella differentialekvationer som approximerar vattenytans höjdläge och de horisontella rörelserna i vattnet. Ekvationerna är en förenkling av Navier Stokes Ekvationer (NSE) som beskriver flöden i tre dimensioner. NSE bygger på kontinuitetsekvationen (masskonserveringskvationen) och rörelsemängdsekvationen. Dessa bygger i sin tur på Newtons andra lag. Vid användning av SWE antas det att vattnets strömningshastighet är lika stor på alla djup i vattnet för varje lägeskoordinat.

I SWE görs förenklingar och antaganden som gör beräkningarna enklare jämfört med NSE. Beräkningar på komplexa geometrier är mycket mer arbetskrävande med NSE än med SWE.

Kontinuitetsekvationen skrivs enligt (Wolfram Research, 2005).

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{u}) \quad (1)$$

där  $\mathbf{u}=(u,v,w)$  är en hastighetsvektor i x-, y- och z-led.

Rörelsemängdsekvationen skrivs enligt (Wolfram Research, 2005).

$$\rho \left( \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) = -\frac{\partial P}{\partial x} + \nu \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) + F_x \quad (2)$$

$$\rho \left( \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} \right) = -\frac{\partial P}{\partial y} + \nu \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} \right) + F_y \quad (3)$$

$$\rho \left( \frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} \right) = -\frac{\partial P}{\partial z} + \nu \left( \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \right) + F_z \quad (4)$$

där  $\rho$  är vattnets densitet,  $P$  totaltryck,  $\nu$  viskositet och  $F$  Coriolisparametern. Fluidens viskositet kan i SWE bortses ifrån då den är försumbar jämfört med de övriga termerna (George, 2004).

Totaltrycket kan delas upp i en djupberoende jämviktsterm och en fluktuerande term enligt

$$P = p_0(z) + p'(x, y, z, t) \quad (5)$$

och detsamma kan göras för densiteten enligt

$$\rho = \rho_0(z) + \rho'(x, y, z, t) \quad (6)$$

Sedan definieras  $\rho_0$  och  $p_0$  så att de uppfyller den hydrostatiska ekvationen (7) (Randall, 2004). Den hydrostatiska ekvationen gäller när förhållandet mellan den vertikala skalan och den horisontella är litet (Geophysical Fluid Dynamics Laboratory, 2005).

$$\frac{\partial p_0}{\partial z} = -g\rho_0 \quad (7)$$

Då kan rörelsemängsekvationen, enligt Randall (2004), skrivas

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = \frac{1}{\rho} \frac{\partial p'}{\partial x} + F_x \quad (8)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = \frac{1}{\rho} \frac{\partial p'}{\partial y} + F_y \quad (9)$$

$$\frac{\partial w}{\partial t} + u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} = \frac{1}{\rho} \frac{\partial p'}{\partial z} + F_z \quad (10)$$

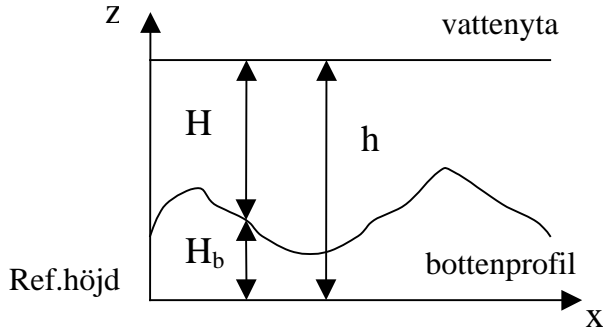
Den första approximationen i SWE är att fluiden är inkompressibel, det vill säga densiteten ( $\rho$ ) är konstant vilket ger  $\rho = \rho_0$  och  $\rho' = 0$ . Kontinuitetsekvationen reduceras då till

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \quad (11)$$

Den andra approximationen är att försumma Corioliseffekten, som uppstår på grund av jordens rotationsacceleration. Den påverkar inte strömningen nämnvärt på en så begränsad volym som Vattenparken är.

För att begränsa beräkningarna till två dimensioner antas effekten av den vertikala skjuvningen från den horisontella strömningen vara försumbar. Hastigheten på alla djup för en viss lägeskoordinat antas vara av samma storlek. Denna approximation kan anses godtagbar för grunda vatten (Randall, 2004).

Djupet i dammen sätts till  $H$  och höjden på botten, utgående från ett lokalt höjdsystem, sätts till  $H_b$ . Den totala höjden vid vattenytan blir då  $h = H + H_b$  (se Figur 3).



Figur 3. Definierade höjder i dammen.

Som randvillkor vid vattenytan ( $z = h$ ) sätts att trycket är konstant och inget flöde kan ske genom ytan. Flödet i vertikalled blir då samma som höjdförändringen av vattenytan.

$$w(x, y, h, t) = \frac{\partial h}{\partial t} \quad (12)$$

Vid botten sätts att inget normalflöde kan ske. Det kan alltså inte flöda vinkelrätt mot botten så flödet måste följa bottenprofilen. Detta ger

$$w(x, y, H_b, t) = \frac{DH_b}{Dt} \quad (13)$$

eller

$$w(x, y, H_b, t) = \frac{\partial H_b}{\partial t} + \mathbf{v} \cdot \nabla H_b \quad (14)$$

där  $\mathbf{v} = (u, v)$ . Då bottenprofilen är konstant i tiden kan den vertikala hastighetskomponenten vid botten beskrivas enligt

$$w(x, y, H_b, t) = u \frac{\partial H_b}{\partial x} + v \frac{\partial H_b}{\partial y} \quad (15)$$

De horisontella krafterna är oberoende av höjden,  $z$ , vilket ger att den vertikala kontinuitetsekvationen kan integreras med avseende på  $z$ .

$$w(x, y, z, t) = -z \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + w(x, y, H_b, t) \quad (16)$$

För  $z = h$  kan randvillkoren för vattenytan och botten sättas in.

$$\frac{\partial h}{\partial t} = -h \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) + u \frac{\partial H_b}{\partial x} + v \frac{\partial H_b}{\partial y} \quad (17)$$

Då  $h=H+H_b$  blir  $H=h- H_b$  och den slutliga förenklade kontinuitetsekvationen beskrivs då enligt

$$\frac{\partial h}{\partial t} + \frac{\partial}{\partial x}(u \cdot H) + \frac{\partial}{\partial y}(v \cdot H) = 0 \quad (18)$$

Om den hydrostatiska ekvationen integreras mellan ett godtyckligt djup och vattenytan fås

$$p(x, y, z, t) - p(x, y, h, t) = -g\rho(h - z) \quad (19)$$

Randvillkor för vattenytan och det godtyckliga djupet ges enligt

$$p(x, y, h, t) = p_h \quad (20)$$

$$p(x, y, z, t) = g\rho(h - z) + p_h \quad (21)$$

Eftersom  $p = -g\rho z + p'(x, y, z, t)$  fås

$$p'(x, y, z, t) = g\rho h + p(h) \quad (22)$$

Derivering av  $p'$  och insättning i de horisontella rörelsemängsekvationerna ger

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -g \frac{\partial h}{\partial x} \quad (23)$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = -g \frac{\partial h}{\partial y} \quad (24)$$

De två förenklade ekvationerna för horisontell rörelsemängd (23 och 24) och den förenklade vertikala kontinuitetsekvationen (18) bildar tillsammans Shallow Water Ekvationerna, SWE, med de tre okända variablerna  $u$ ,  $v$  och  $h$ .

### 3 METODER

Modellens uppbyggnad bestod av flera steg. Först interpolerades dammens botten-topografi fram i ArcMap, ett GIS-program. Botten-topografin importerades sedan till Matlab. Shallow Water Ekvationerna som användes vid flödessimuleringen differentierades först innan de också kunde implementeras i Matlab. Efter implementering och vissa justeringar av modellen kunde vattenströmningen i dammen simuleras för olika inflöden. Hur varje steg utfördes redovisas mer i detalj nedan.

#### 3.1 INTERPOLERING AV BOTTENTOPOGRAFI

För att kunna simulera flödet genom dammen måste dammens botten-topografi finnas tillgänglig. Efter litteraturstudier av olika interpoleringsmetoder i GIS-programmet *ArcMap* (*ArcView*) beslutades det att interpoleringen skulle genomföras med en operator tillhörande *Spatial Analyst*, en *Extension* i ArcMap. Operatoren, *Topo To Raster*, använder en iterativ finit differens interpolationsteknik. Den är optimerad till att ha de beräkningsegenskaper som lokala interpolationsmetoder har, såsom *avståndsviktad medelvärdesinterpolation*. Den ska samtidigt inte förlora ytans kontinuitet. Detta fås bäst genom globala interpolationsmetoder, såsom *Spline* och *Kriging* (ArcGIS Desktop Help, 2005). Det är den enda interpolationsoperatoren i ArcGIS som är specialiserad på att arbeta med konturer som indata.

Innan interpoleringen lades höjdkonturer över dammen in i ArcMap baserat på en bakomliggande karta. Det visade sig att interpolering med *Topo To Raster* inte lyckades varken för interpolering mellan punkter eller mellan konturer. Även flera andra interpolationsmetoder i ArcMap testades men lyckades ej. Den enda metod som producerade ett resultat i ArcMap var *avståndsviktad medelvärdesinterpolation*. Ett problem med denna metod är att man ofta kan se i den interpolerade ytmodellen var de ursprungliga datapunkterna ligger, vilket även skedde i detta fall. Ursprungspunkterna syntes tydligt och visade sig som lokala höjder eller sänkor. Detta berodde på att de ursprungliga datapunkterna fick mycket stor effekt på den beräkningscell som de låg inom (Eklundh, 2001). Det beslutades därför att inte använda denna metod.

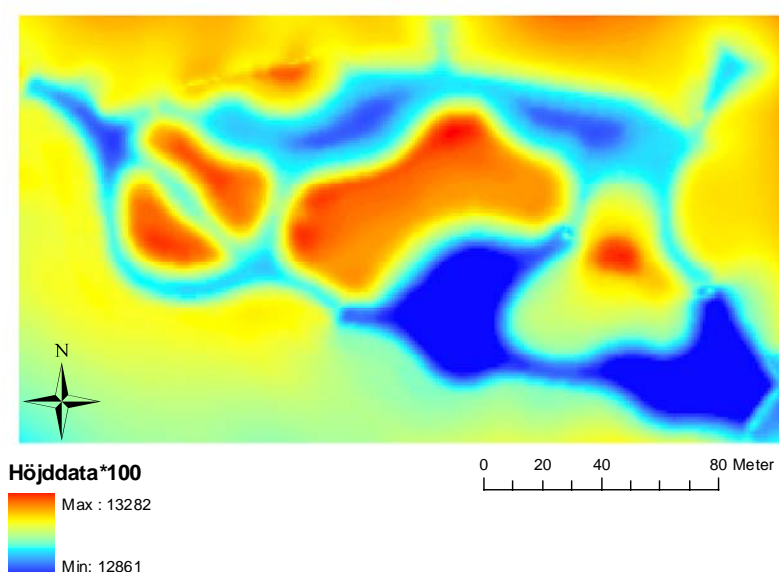
Efter det misslyckade försöket att använda ArcMap vid interpoleringen användes istället operatoren *TOPOGRID* som kan nås från Arc-fönstret. De flesta verktyg som finns i ArcMap finns även i Arc. Topogrid fungerar på exakt samma sätt som *Topo To Raster* och i detta fall lyckades interpoleringen. Den enda skillnaden var att *shapefilen* innehållande höjdkonturer fick konverteras om till en *coverage* innan interpoleringen. Indata i interpoleringsprocessen var de inlagda höjdkonturerna. Antalet iterationer sattes till 50. *Enforce*, ett kommando som fyller igen sänkor, valdes inaktivt.

För att kunna importera matrisen med höjddata över dammen till Matlab konverterades data till en *.tif-fil* vilken sedan kunde läsas direkt Matlab. Filer i *.tif*-format kan bara hantera heltal och därför multiplicerades alla höjddata med hundra för att inte förlora några decimaler då höjdskillnaderna i dammen är små. Multiplikationen utfördes med verktyget *Raster Calculator* som finns i *Spatial Analyst*. I Matlab konverterades filen sedan om till en *double array* så att den även kan hantera



decimaltal och alla höjddata dividerades med hundra igen för att få de verkliga värdena. Höjdmatriken innehöll således höjder med en noggrannhet på två decimaler, det vill säga 1 centimeter.

Den framinterpolerade höjdmatriken innehöll 147 rader och 261 kolumner. Varje höjdvärde låg med en meters mellanrum i både x- och y-led vilket medförde att den framinterpolerade bottenpografien var 146 meter bred och 260 meter lång och innehöll drygt 38000 höjdvärden. Höjderna gavs i meter mot ett lokalt referenssystem. Det högsta höjdvärdet var 132,82 meter och det lägsta 128,61 meter. Dammen var inte rektangulär, och dessutom mindre än den framinterpolerade ytan, vilket medförde att en stor del av den interpolerade ytan bestod av landområden runt omkring dammen. Den framinterpolerade bottenpografien innan höjddata dividerats med hundra visas i Figur 4.



Figur 4. Den framinterpolerade bottenpografien med höjddata multiplicerat med 100.

## 3.2 MODELLUPPBYGGNAD

### 3.2.1 Numeriska metoder

Vid implementering av SWE i Matlab måste ekvationerna först differentieras. Det finns ett stort antal numeriska metoder att använda vid differentiering. Det enklaste sättet att differentiera ekvationerna är att använda en explicit metod. En explicit metod är inte ovillkorligt stabil, men den är lika noggrann som en implicit metod. Om simuleringen av flödet i dammen kan fås stabilt för en explicit metod krävs mindre arbete och därför användes först en explicit metod vid differentiering av SWE. På tidsderivatan användes första ordningens Euler framåt och på alla rumsderivator användes centrerade differenser.

Explicita metoder är enkla att implementera, men är ofta känsliga för numeriska instabiliteter. Denna känslighet kan man få bukt med genom att istället använda en implicit metod vid differentieringen. I implicita metoder utvecklas rumsderivatan helt

eller delvis i slutet av tidssteget ( $t+I$ ). Dessvärre är implicita metoder vanligtvis mycket mer komplicerade att implementera än explicita. (Flow Science, 2005)

En välkänd implicit metod är Crank-Nicholson (C-N). C-N tar medelvärdet av rumsderivatan mellan början och slutet på tidssteget. Detta ger att C-N är en andra ordningens metod i tiden och rummet. C-N är en mycket noggrann och ovillkorligt stabil metod (se nästa avsnitt), för linjära problem utan randvillkor. För varje tidssteg måste en femdiagonal (tridiagonal vid en dimension) matris inverteras vilket är arbetskrävande. Detta kan resultera i en långsam modell. På grund av stabiliteten kan större tidssteg tas än för explicita metoder (Heath, 2002).

### 3.2.1.1 Explicit differentiering

För att förenkla differentieringen ansattes  $z$  och  $q$  enligt ekvation (25) och (26).

$$z = u * H \quad (25)$$

$$q = v * H \quad (26)$$

SWE efter explicit differentiering ges av ekvation (27)-(29).  $i$  är indexering i y-led,  $j$  indexering i x-led och  $t$  indexering i tiden i alla kommande ekvationer.

$$u_{i,j}^{t+1} = u_{i,j}^t - \left[ \frac{u_{i,j}^t (u_{i,j+1}^t - u_{i,j-1}^t)}{2dx} + \frac{v_{i,j}^t (u_{i+1,j}^t - u_{i-1,j}^t)}{2dy} + \frac{g(h_{i,j+1}^t - h_{i,j-1}^t)}{2dx} \right] dt \quad (27)$$

$$v_{i,j}^{t+1} = v_{i,j}^t - \left[ \frac{u_{i,j}^t (v_{i,j+1}^t - v_{i,j-1}^t)}{2dx} + \frac{v_{i,j}^t (v_{i+1,j}^t - v_{i-1,j}^t)}{2dy} + \frac{g(h_{i+1,j}^t - h_{i-1,j}^t)}{2dy} \right] dt \quad (28)$$

$$h_{i,j}^{t+1} = h_{i,j}^t - \left[ \frac{z_{i,j+1}^t - z_{i,j-1}^t}{2dx} + \frac{q_{i+1,j}^t - q_{i-1,j}^t}{2dy} \right] dt \quad (29)$$

### 3.2.1.2 Semi-Implicit differentiering

Implicit differentiering med C-N gav

$$u_{i,j}^{t+1} = u_{i,j}^t - \left[ u_{i,j}^t \left( \frac{u_{i,j+1}^{t+1} - u_{i,j-1}^{t+1}}{4dx} + \frac{u_{i,j+1}^t - u_{i,j-1}^t}{4dx} \right) + v_{i,j}^t \left( \frac{u_{i+1,j}^{t+1} - u_{i-1,j}^{t+1}}{4dy} + \frac{u_{i+1,j}^t - u_{i-1,j}^t}{4dy} \right) \right] dt \quad (30)$$

$$- \left[ g \left( \frac{h_{i,j+1}^t - h_{i,j-1}^t}{2dx} \right) \right] dt$$

$$v_{i,j}^{t+1} = v_{i,j}^t - \left[ u_{i,j}^t \left( \frac{v_{i,j+1}^{t+1} - v_{i,j-1}^{t+1}}{4dx} + \frac{v_{i,j+1}^t - v_{i,j-1}^t}{4dx} \right) + v_{i,j}^t \left( \frac{v_{i+1,j}^{t+1} - v_{i-1,j}^{t+1}}{4dy} + \frac{v_{i+1,j}^t - v_{i-1,j}^t}{4dy} \right) \right] dt \quad (31)$$

$$- \left[ g \left( \frac{h_{i+1,j}^t - h_{i-1,j}^t}{2dy} \right) \right] dt$$

$$h_{i,j}^{t+1} = h_{i,j}^t - \left[ u_{i,j}^t \left( \frac{(h-h_B)_{i,j+1}^{t+1} - (h-h_B)_{i,j-1}^{t+1}}{4dx} + \frac{(h-h_B)_{i,j+1}^t - (h-h_B)_{i,j-1}^t}{4dx} \right) \right] dt - \left[ (h-h_B)_{i,j}^t \left( \frac{u_{i,j+1}^t - u_{i,j-1}^t}{2dx} \right) + (h-h_B)_{i,j}^t \left( \frac{v_{i+1,j}^t - v_{i-1,j}^t}{2dy} \right) \right] dt - \quad (32)$$

$$\left[ v_{i,j}^t \left( \frac{(h-h_B)_{i+1,j}^{t+1} - (h-h_B)_{i-1,j}^{t+1}}{4dy} + \frac{(h-h_B)_{i+1,j}^t - (h-h_B)_{i-1,j}^t}{4dy} \right) \right] dt$$

där  $h_B$  är höjden på bottenpografen och  $h$  höjden på bottenpografen plus vattenytan.

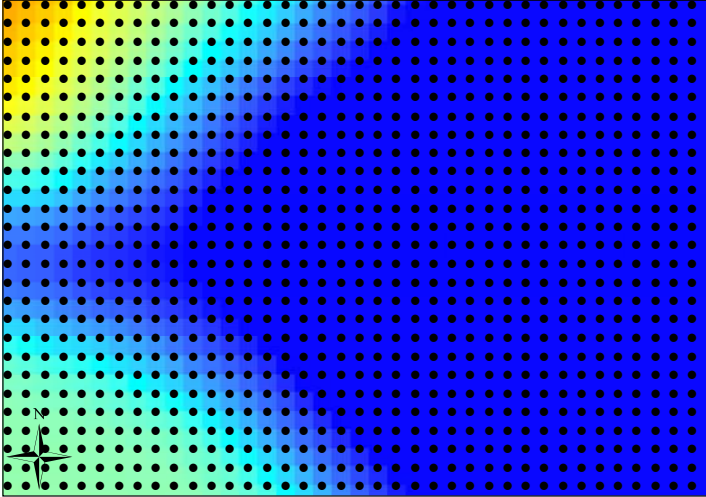
Efter separering av termer med det nya tidssteget i vänster led och det gamla i höger led erhölls

$$u_{i,j}^{t+1} \left( \frac{1}{dt} \right) + u_{i,j+1}^{t+1} \left( \frac{u_{i,j}^t}{4dx} \right) + u_{i,j-1}^{t+1} \left( \frac{-u_{i,j}^t}{4dx} \right) + u_{i+1,j}^{t+1} \left( \frac{v_{i,j}^t}{4dy} \right) + u_{i-1,j}^{t+1} \left( \frac{-v_{i,j}^t}{4dy} \right) = u_{i,j}^t \left( \frac{1}{dt} \right) + u_{i,j+1}^t \left( \frac{-u_{i,j}^t}{4dx} \right) + u_{i,j-1}^t \left( \frac{u_{i,j}^t}{4dx} \right) + u_{i+1,j}^t \left( \frac{-v_{i,j}^t}{4dy} \right) + u_{i-1,j}^t \left( \frac{v_{i,j}^t}{4dy} \right) + h_{i,j+1}^t \left( \frac{-g}{2dx} \right) + h_{i,j-1}^t \left( \frac{g}{2dx} \right) \quad (33)$$

$$v_{i,j}^{t+1} \left( \frac{1}{dt} \right) + v_{i,j+1}^{t+1} \left( \frac{u_{i,j}^t}{4dx} \right) + v_{i,j-1}^{t+1} \left( \frac{-u_{i,j}^t}{4dx} \right) + v_{i+1,j}^{t+1} \left( \frac{v_{i,j}^t}{4dy} \right) + v_{i-1,j}^{t+1} \left( \frac{-v_{i,j}^t}{4dy} \right) = v_{i,j}^t \left( \frac{1}{dt} \right) + v_{i,j+1}^t \left( \frac{-u_{i,j}^t}{4dx} \right) + v_{i,j-1}^t \left( \frac{u_{i,j}^t}{4dx} \right) + v_{i+1,j}^t \left( \frac{-v_{i,j}^t}{4dy} \right) + v_{i-1,j}^t \left( \frac{v_{i,j}^t}{4dy} \right) + h_{i+1,j}^t \left( \frac{-g}{2dy} \right) + h_{i-1,j}^t \left( \frac{g}{2dy} \right) \quad (34)$$







Figur 5. Förstorad bild av beräkningspunkter i dammen precis efter överfallet. Avstånd mellan punkterna var 1 meter i både x- och y-led. Varje beräkningspunkt representerades av ett x- och y-värde samt ett framinterpolerat höjdvärde.

### 3.2.1.3 Stabilitetsanalys

Vid stabilitetsanalys är det önskvärt att förstärkningsfaktorn,  $\lambda A / < I$ . Om detta villkor gäller är metoden ovillkorligt stabil. Nedan följer en stabilitetsanalys för C-N utifrån en förenkling av rörelsemängdekvationen i en dimension.

$$\frac{\partial u}{\partial t} + U \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2} \quad (42)$$

där  $\nu > 0$  är viskositeten och  $U > 0$  en konstant. Efter differentiering med C-N fås

$$\begin{aligned} u_i^{t+1} + \frac{D}{2}(u_{i+1}^{t+1} - u_{i-1}^{t+1}) - \frac{C}{2}(u_{i+1}^{t+1} - 2u_i^{t+1} + u_{i-1}^{t+1}) = \\ u_i^t - \frac{D}{2}(u_{i+1}^t - u_{i-1}^t) + \frac{C}{2}(u_{i+1}^t - 2u_i^t + u_{i-1}^t) \end{aligned} \quad (43)$$

där  $C$  och  $D$  är en konstant där steglängden är inbakad ( $C > 0, D > 0$ ).

Fouriertransform av ekvation 43 ges av

$$\begin{aligned} V^{t+1} e^{ikdx} + \frac{D}{2} V^{t+1} (e^{I(i+1)kdx} - e^{I(i-1)kdx}) - \frac{C}{2} V^{t+1} (e^{I(i+1)kdx} - 2e^{Iikdx} + e^{I(i-1)kdx}) = \\ V^t e^{ikdx} - \frac{D}{2} V^t (e^{I(i+1)kdx} - e^{I(i-1)kdx}) + \frac{C}{2} V^t (e^{I(i+1)kdx} - 2e^{Iikdx} + e^{I(i-1)kdx}) \end{aligned} \quad (44)$$

där  $k$  är vågtal,  $I = \sqrt{-1}$  som vanligtvis betecknas  $i$  men i detta fall är  $i$  steg i rummet i y-led. Efter användning av matematiska samband och omflyttning av termer fås

$$V^{t+1} \left( 1 + \frac{D}{2} (2I \sin(kdx)) + \frac{C}{2} (2 - \cos(kdx) + I \sin(kdx) - \cos(-kdx) + I \sin(-kdx)) \right) = \quad (45)$$

$$V^t \left( 1 - \frac{D}{2} (2I \sin(kdx)) - \frac{C}{2} (2 - \cos(kdx) + I \sin(kdx) - \cos(-kdx) + I \sin(-kdx)) \right)$$

Ytterligare förenklingar ger

$$\begin{aligned} V^{t+1} (1 + DI \sin(kdx) + 2C \sin^2(kdx/2)) = \\ V^t (1 - DI \sin(kdx) - 2C \sin^2(kdx/2)) \end{aligned} \quad (46)$$

Förstärkningsfaktorn,  $A$ , ges enligt

$$A = \frac{V^{t+1}}{V^t} \quad (47)$$

vilket för diffusionsekvationen då blir

$$A = \frac{1 - DI \sin(kdx) - 2C \sin^2(kdx/2)}{1 + DI \sin(kdx) + 2C \sin^2(kdx/2)} \quad (48)$$

vilket ger att  $|A|^2 < 1$  för alla värden på vågtal,  $k$ . Detta ger att C-N är en ovillkorligt stabil metod. Den implicita metoden som använts i denna uppgift är dock inte ovillkorligt stabil då differentieringen inte är fullt implicit (personligen Rantakokko).

### 3.3 IMPLEMENTERING

För att på ett enkelt och effektivt sätt kunna felsöka kod och få en fungerande modell förenklades först problemet till en dimension, endast en linje. Då är det lättare att se var i området instabiliteter börjar växa, om det är på randen eller i det inre av området. Det är även enklare att kontrollera att alla matriser och ekvationer implementerats på rätt sätt. När den endimensionella modellen sedan fungerade tillfredsställande var det enkelt att bygga ut modellen till en tvådimensionell. Samma ekvationer kunde användas och endast y-derivatorna och termer innehållande hastigheten i y-led,  $v$ , behövde läggas till.

Både den explicita och den implicita metoden implementerades i Matlab 7.0. Den explicita metoden kunde implementeras på samma sätt som i teoridelen ovan, men den implicita var mer komplicerad.

För den implicita metoden fick bottenprofilen först definieras om så att den istället för att vara en matris blev en lång kolonnvektor med alla rader liggandes efter varandra. Även hastighetsvektorerna,  $u$  och  $v$ , och höjden vid vattenytan,  $h$ , definierades som långa kolonnvektorer. Detta var nödvändigt för att kunna utföra beräkningarna. De sökta parametrarna  $u$ ,  $v$  och  $h$  beräknades sedan om i två steg för varje tidssteg enligt

$$\bar{u}temp^{t+1} = A1(\bar{u}^t, \bar{v}^t) \setminus (B1 \cdot \bar{u}^t + \bar{b}_1 + C1 \cdot \bar{h}^t + \bar{b}_h) \quad (49)$$

$$\bar{v}temp^{t+1} = A2(\bar{u}^t, \bar{v}^t) \setminus (B2 \cdot \bar{v}^t + \bar{b}_2 + C2 \cdot \bar{h}^t + \bar{b}_h) \quad (50)$$

$$\bar{h}temp^{t+1} = AH(\bar{u}^t, \bar{v}^t) \setminus (BH \cdot \bar{h}^t + \bar{b}_h + CH \cdot \bar{h}_B) \quad (51)$$

$$\bar{u}^{t+1} = A1(\bar{u}temp^{t+1}, \bar{v}temp^{t+1}) \setminus (B1 \cdot \bar{u}^t + \bar{b}_1 + C1 \cdot ((\bar{h}temp^{t+1} + \bar{h}^t)/2) + \bar{b}_h) \quad (52)$$

$$\bar{v}^{t+1} = A2(\bar{u}temp^{t+1}, \bar{v}temp^{t+1}) \setminus (B2 \cdot \bar{v}^t + \bar{b}_2 + C2 \cdot ((\bar{h}temp^{t+1} + \bar{h}^t)/2) + \bar{b}_h) \quad (53)$$

$$\bar{h}^{t+1} = AH(\bar{u}temp^{t+1}, \bar{v}temp^{t+1}) \setminus (BH \cdot ((\bar{h}temp^{t+1} + \bar{h}^t)/2) + \bar{b}_h + CH \cdot \bar{h}_B) \quad (54)$$

$\bar{b}_1$ ,  $\bar{b}_2$  och  $\bar{b}_h$  var randvillkor för  $u$ ,  $v$  och  $h$  som bakades in i ekvationerna. Hur de sattes redovisas i nästa avsnitt.

För båda metoderna beräknades parametrarna först för ett fixt antal tidssteg, 10 stycken. När dessa steg beräknats fortsatte sedan beräkningarna i en *while*-loop. Så länge som värdet på *epsilon*; som var normen av kvoten mellan skillnaden av det gamla och nya värdet på matrisen  $u$  och det nya värdet på densamma (ekvation 55), var större än 0,0001 fortsatte beräkningarna. När epsilon blev mindre än 0,0001 avslutades beräkningarna och stationärt tillstånd ansågs vara uppnått. Anledningen till att först simulera för ett fixt antal steg är att i de första stegen är ändringen av parametrarna vanligen liten och stabilitetsvillkoret skulle därför kunna uppnås väldigt tidigt utan att egentlig stabilitet uppnåts. Det var därför av säkerhetsskäl som den initiala simuleringen med fixt antal steg skedde.

$$epsilon = norm\left(\frac{\bar{u}^t - \bar{u}^{t+1}}{\bar{u}^t}\right) \quad (55)$$

När beräkningarna var avslutade, för den implicita metoden, konverterades kolonnvektorerne innehållande  $u$ ,  $v$  och  $h$  om till matriser igen för att resultatet skulle kunna tolkas. För den explicita metoden kunde resultatet tolkas direkt utan konverteringar.

### 3.4 RANDVILLKOR

I dammen fanns öar, strandkanter, inlopp och utlopp. Detta ger att stora delar av den framinterpolerade bottenpografin bestod av landområden. På dessa områden flödade inget vatten och därför fick specifika randvillkor ansättas i de mätpunkter där vatten och land angränsade till varandra och vid inlopp och utlopp till dammen. SWE är väldigt känsliga för fel i randvillkoren (McDonald, 2003). Det var därför viktigt att de behandlades med försiktighet. Alla randvillkor bakades in i ekvationerna så att randvillkoren var uppfyllda när SWE beräknats.



### 3.4.1 Inlopp

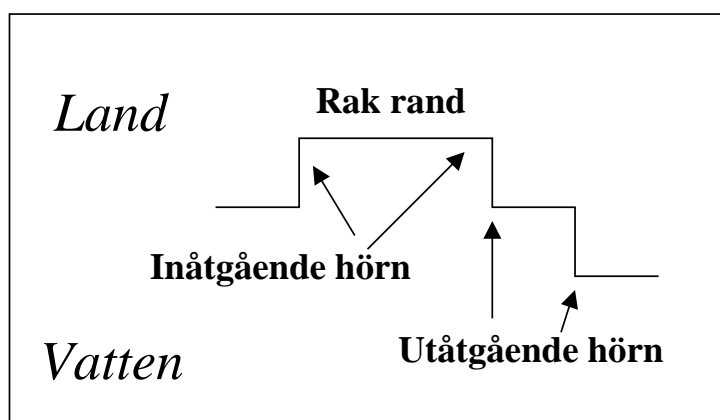
Vid mätningar har flödet in till dammen varierat mellan 5 l/s och 5 m<sup>3</sup>/s. Som inflödesrandvillkor sattes en konstant hastighet som låg inom detta intervall. I SWE beräknas hastigheter och inte flöden och inflödet beräknades därför om till hastigheter med hjälp av inflödesarean. Inströmningshastigheterna blev då mellan  $1,2 \cdot 10^{-3} - 1,2$  m/s. Hastigheten i y-led sattes till inströmningshastigheten då inflödet inte kan vara en resultant av en hastighet i y-led och en i x-led. Inströmningshastigheten i x-led sattes således därför till noll. Randvärdet på  $h$  beräknades enligt SWE. Inloppet sattes som 7 beräkningspunkter längst upp i vänstra hörnet av dammen.

### 3.4.2 Utlopp

I utloppet sattes utströmningshastigheterna så att utflödet blev lika stort som inflödet. Detta för att stationära förhållanden skulle råda.  $h$  beräknades enligt SWE. Utloppet sattes som de 12 beräkningspunkter som låg längst ner i högra hörnet av dammen.

### 3.4.3 Öar och strandkanter

Det fanns fyra öar i dammen och även en lång sträcka runt hela dammen som bestod av strandkant. Eftersom beräkningspunkterna låg med lika stora mellanrum över hela dammen bildades ett kvadratisk rutsystem av beräkningspunkter. Ränderna blev då kantiga och varje randpunkt var antingen ett utåtgående hörn, ett inåtgående hörn eller låg på en rak linje i antingen x- eller y-riktning (Figur 6). Dessa randvillkor var de mest komplicerade att ansätta. Det fanns flera olika alternativ, såsom att inget vatten fick flöda längs ränderna eller att det fick flöda om randen var en rak linje, men inte om den var ett hörn.



Figur 6. Exempel på olika randpunkter i beräkningsområdet.

De randvillkor som valdes i den slutgiltiga modellen var:

- För utåtgående hörn: Inget flöde i randpunkten och  $h$  extrapolerades till samma värde som  $h$  hade i närmaste mät punkt belägen i vatten.

- För inåtgående hörn: Inget flöde i randpunkten och  $h$  beräknades som medelvärde av de två närliggande randpunkterna.
- För raka ränder: Inget flöde i randpunkten och  $h$  beräknades enligt SWE.

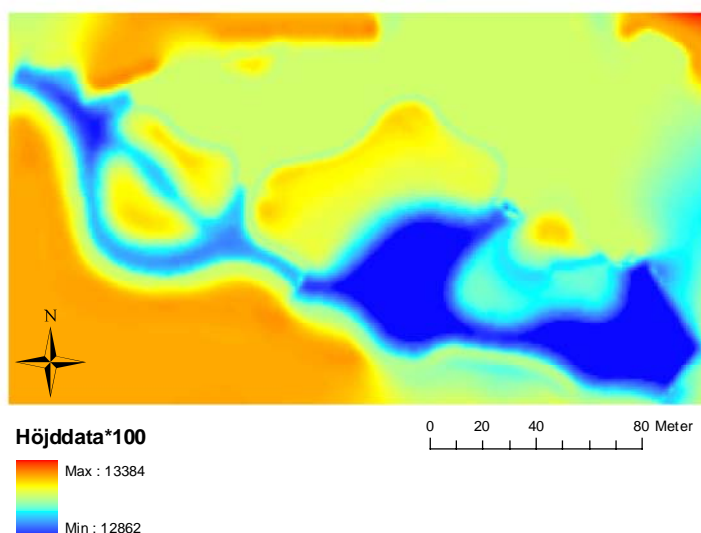
### 3.5 SIMULERING AV VATTENSTRÖMNING

Till att börja med utfördes simuleringar för mindre, enklare bottentopografier än den framinterpolerade. Detta för att få en så enkel simulering som möjligt för att avgöra om modellen fungerade eller ej. Den första topografien som användes var helt jämn utan öar och övriga landområden. Den var även betydligt mindre än dammen (50 meter i både x- och y-led). Sedan användes en varierande bottentopografi av samma storlek men vatten fick flöda fritt över hela området. Inga randvillkor för strandkanter ansattes, vilket medförde att vatten kunde flöda fritt över öar och land.

När modellen fungerade tillfredsställande ansattes en viss höjd som ansågs vara strandkant. Ett stort problem med SWE är att en varierande strandkant, vilket är naturligt, ej kan ansättas. En specifik höjd fick istället bestämmas vara strandkant och inget vatten kunde således flöda ovanför denna höjd. När modellen även fungerade med öar och land inlagt utökades beräkningsområdet till att bestå av det egentliga, nämligen Vattenparken.

Efter ett överfall, som ligger cirka 112 meter från beräkningsområdets vänstra kant ( $x=113$ ), minskade vattnets höjd med cirka 1,25 meter på en bara några centimeters lång sträcka. Detta medförde att den ansatta strandlinjen, även den, måste minskas med lika mycket. Detta löstes genom att alla beräkningspunkter till vänster om överfallet tilldelades en högre strandlinje och alla till höger om överfallet en lägre strandlinje. Den vänstra delen av dammen tilldelades en strandlinje på 131,75 meter och den högra en på 130,5 meter. Höjderna utgick ifrån ett lokalt referenssystem.

För att endast simulera flöden för den delen av dammen som tjänar i reningssyfte, den nedre delen, höjdes den övre delen, rekreationsdelen, upp till landnivå. På så sätt strömmade inget vatten i detta område under simuleringarna och beräkningar behövdes endast utföras i området av intresse. Eftersom de två delarna inte delade samma inlopp och utlopp påverkade detta inte simuleringarnas resultat. Figur 7 visar den framinterpolerade bottentopografien med den övre dammdelen höjd ovan landnivå.



Figur 7. Den framinterpolerade bottentopografin med den övre dammdelen höjd ovan landnivå. Höjddata är multiplicerat med 100.

Simuleringar för hela dammen gick till en början väldigt långsamt på grund av det stora antalet beräkningspunkter. Olika metoder att försöka snabba upp koden tillämpades därför. Med hjälp av funktionen *profile* åskådliggjordes tidsåtgången för olika funktioner och ekvationer (Pärt-Enander & Sjöberg, 1998). Det beslutades därmed vilka delar av koden som behövde mest bearbetning för uppsnabbning. Stort arbete lades ned på tilldelning av diagonaler i matriserna. De kan antingen tilldelas genom att använda en *for*-loop eller genom kommandot *diag* (Pärt-Enander & Sjöberg, 1998). Loopar är allmänt tidskrävande och kräver mer arbete och tar längre tid att utföra än *diag*. Då matriserna som användes i beräkningarna innehöll 38367 element i varje rad och kolumn och således var mycket stora var kommandot *diag* inte tillräckligt effektivt. Matriserna innehöll för många termer och tilldelningen misslyckades. Det blev därför loopar som användes till detta.

För att ytterligare snabba upp koden minskades beräkningsområdet. Endast varannat värde ur höjdmatriken i både x- och y-led användes vid simuleringarna. Detta gjorde att antalet beräkningspunkter minskade med en fjärdedel till 9563 stycken och varje tidssteg beräknades då snabbare än vid beräkningar med hela höjdmatriken. Bottentopografin med varannat värde borttaget studerades och det beslutades att fördelarna med att modellen räknade snabbare övervägde nackdelarna med att bottentopografin blev mindre noggrann.

När modellen uppnått önskad simuleringshastighet kunde den köras för längre tidsserier än tidigare. Detta medförde att instabiliteter som tidigare inte upptäckts visade sig efter ett större antal tidssteg. För att motverka dessa ändrades vissa delar av koden. Tidssteget sattes mindre för att få en mer stabil simulering, vissa ojämnheter i bottentopografin med för skarpa kanter jämnades till och randvillkoren justerades till de som beskrivs ovan.

Längs ränderna var det på vissa ställen väldigt liten höjdskillnad mellan landområden och områden med ovanliggande vatten. Detta resulterade i instabilt flöde på grund av för grunt vatten. I områden där vattendjupet var mindre än 3 dm sänktes därför bottentopografin så att djupet blev minst 3 dm överallt.

Modellen kördes sedan för fyra stycken olika inströmningshastigheter;  $1,2 \cdot 10^{-3}$ ,  $1,2 \cdot 10^{-2}$ ,  $1,2 \cdot 10^{-1}$  och 1,2 m/s. Initialvärden på vattenströmningen sattes lika stort i både x- och y-led, nämligen samma som inflödet för alla beräkningspunkter. Optimalt hade varit att hitta en initialgissning nära de slutliga värdet på hastigheterna i varje beräkningspunkt. Konvergensen hade då blivit mycket snabbare. Att hitta en bra initialgissning visade sig dock vara komplicerat, och därför valdes den enligt ovan. Därefter analyserades vattenströmningen för varje inströmningshastighet. Modellen tog således inte hänsyn till nederbörd över vattenytan eller annat eventuellt vattenintag eller vattenuttag utan endast till in- och utströmning i dammen.

### 3.6 SEDIMENTATIONSANALYS

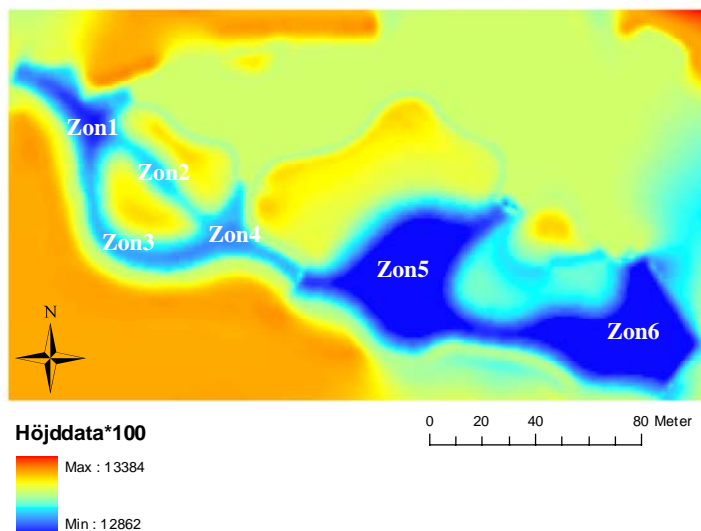
Allmänt gäller att ju långsammare vattnet flödar desto fler partiklar hinner sedimentera. Sedimentationshastigheten beror på densitetsskillnaden mellan partikeln och vätskan partikeln färdas i samt på vätskans turbulens. Tyngre partiklar sedimenterar snabbare än lättare partiklar. Tabell 1 ger en fingervisning på hur länge det tar för en partikel att falla till botten i en 1 meter djup bassäng med stillastående vatten (Greppa Näringen, 2005). Två olika metoder användes vid sedimentationsanalysen vilka förklaras i följande avsnitt.

*Tabell 1. Ungefärlig fallhastighet i en 1 meter djup bassäng med stillastående vatten.*

| <b>Fraktion</b> | <b>Underavdelning</b> | <b>Diameter (mm)</b> | <b>Sedimentationshastighet</b> |
|-----------------|-----------------------|----------------------|--------------------------------|
| Grus            | Grovgrus              | 20-6                 | 1 sekund                       |
|                 | Fingrus               | 6-2                  |                                |
| Sand            | Grovsand              | 2-0,6                | 10 sekunder                    |
|                 | Mellansand            | 0,6-0,2              |                                |
| Mo              | Grovmo                | 0,2-0,06             | 2 minuter                      |
|                 | Finmo                 | 0,06-0,02            |                                |
| Mjåla           | Grovmjåla             | 0,02-0,006           | 2 timmar                       |
|                 | Finmjåla              | 0,006-0,02           |                                |
| Lera            | Grovlera              | 0,002-0,0006         | 8 dygn                         |
|                 | Finlera               | 0,0006-0,0002        |                                |

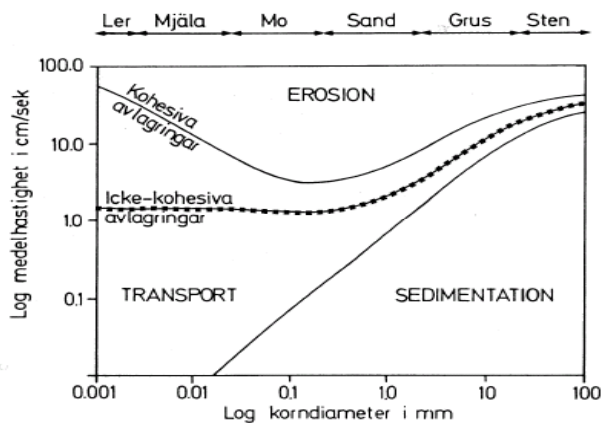
#### 3.6.1 Metod 1 - Hjulströms diagram

För att kunna avgöra om det i dammen sker sedimentation, resuspension eller transport av partiklar vid olika inströmningshastigheter delades först dammen in i sex olika zoner enligt Figur 8. Efter varje flödessimulering studerades sedan strömningshastigheterna inom varje zon.



Figur 8. De sex zonerna i Vattenparken. Höjddata är multiplicerat med 100.

För att kunna avgöra vid vilken strömningshastighet en viss partikelstorlek sedimenterar användes diagrammet i Figur 9. Diagrammet heter Hjulströms diagram och visar vid vilka strömningshastigheter olika partikelstorlekar sedimenterar, fortsätter transporteras med det strömmande vattnet eller resuspenderas (eroderar) från botten. Diagrammet togs fram 1939 och gäller för en plan oföränderlig botten med en meters vattendjup (Physical Geology Laboratory, 2000), men antogs att gälla som approximation i detta fall.



Figur 9. Hjulströms diagram (efter Hjulström, 1939).

För att kunna avläsa diagrammet behövdes information om partikelstorlekarna i det inströmmande vattnet i Vattenparken. Sedimentprov togs därför, med hjälp av en Ekmanhuggare, från tre olika punkter av Vattenparkens botten; från zon 1, zon 4 och från gränsen mellan zon 5 och 6. Proven visade att det fanns mest sediment vid inloppet. Mängden minskade sedan successivt till att det efter överfallet inte fanns något sediment alls. Det tredje provet innehöll således inget sediment alls och partikelanalysen utfördes därför enbart på de två första proverna.

Sedimentproverna torkades i 14 timmar i 105 °C så att allt vatten försvann. Därefter värmdes proven i ytterligare 2 timmar i 540 °C så att allt organiskt material förbrändes och bildade koldioxid och endast kristallint material återstod (glödförlust). Proverna studerades sedan i mikroskop. Provet från zon 1 innehöll partiklar som klassades som finmo. Partiklarna i provet från zon 4 var mindre och de klassades som finmjåla. Att partiklarna var mindre i zon 4 än i zon 1 ansågs naturligt eftersom att det tar längre tid för lättare partiklar att sedimentera än för tyngre. Därefter approximerades zon 1, 2 och 3 som zoner med sediment klassat som finmo och zon 4, 5 och 6 med sediment klassat som finmjåla.

Strömningshastigheten i varje zon beräknades som medelhastigheten av rektangulärt område i zonen. Diagrammet lästes sedan av för de olika zonerna efter varje simulering. Därefter klassades de olika zonerna som antingen sedimentationszon, transportzon eller resuspensionszon för varje simulerad inströmningshastighet.

### 3.6.2 Metod 2 – Beräkning av sedimentationshastighet

För att öka säkerheten kring sedimentationsanalysen användes ytterligare ett tillvägagångssätt för att klassa de olika zonerna. Genom att först beräkna sedimentationshastigheten kunde det utifrån strömningshastigheten och djupet i dammen avgöras var i dammen de olika partikelstorlekarna sedimenterade.

Baban (1995) beskriver en beräkningsmetod för sedimentationshastigheten som är applicerbar oberoende av värdet på Reynolds tal, det vill säga oberoende om flödet var turbulent eller laminärt.

$$V_s = \frac{\left[9v_k^2 + gd^2(G_s - 1)(0.03869 + 24.80d) \cdot 10^{-3}\right]^{1/2} - 3v_k}{(0.11607 + 74.405d) \cdot 10^{-3}} \quad (56)$$

$V_s$  är sedimentationshastigheten i m/s,  $g$  gravitationsaccelerationen i  $m/s^2$ ,  $d$  partikeldiametern i mm,  $G_s$  den specifika gravitationen för partikeln i vätskan och  $v_k$  vätskans kinematiska viskositet i  $m^2/s$ .  $G_s$  är dimensionslös och beräknades enligt ekvation (57) (Biosystems and Agricultural Engineering –University of Kentucky, 2004).

$$G_s = \frac{\rho_s}{\rho_v} \quad (57)$$

Den kinematiska viskositeten beräknades enligt

$$v_k = \frac{v_d}{\rho_v} \quad (58)$$

där  $v_d$  är vätskans dynamiska viskositet i  $Ns/m^2$  (Fox & McDonald, 1994). Den dynamiska viskositeten för vatten sattes till  $1,025 \cdot 10^{-3} Ns/m^2$  (Physics Handbook, 1987). Vid beräkningarna av sedimentationshastigheten sattes sedan  $g=9,81 m/s^2$  och  $\rho_v=1000 kg/m^3$  vilket är densiteten för vatten.  $\rho_s$  sattes till ett medelvärde mellan

densiteten för kvarts ( $2,65 \cdot 10^3 \text{ kg/m}^3$ ) och densiteten för fältspat ( $2,6 \cdot 10^3$ - $2,7 \cdot 10^3 \text{ kg/m}^3$ ), således  $2,65 \cdot 10^3 \text{ kg/m}^3$  (Pellant, 1990). Detta för att det i sedimentproverna kunde urskiljas kvarts- och fältspatmineral.

Sedimentationshastigheten beräknades enligt ekvation (56) för ett medelvärde av de två olika partikelstorlekarna (finmo och finmjäla).

$$V_s(\text{finmo} / \text{finmjäla}) = 0,19 \text{ mm/s} \quad (59)$$

Medeldjupet i dammen sattes till 1,5 meter. För att partiklarna ska sedimentera krävdes då en uppehållstid i dammen på cirka 131 minuter. Utifrån den simulerade strömningshastigheten kunde det avgöras om partiklarna hann sedimentera och var i dammen detta i sådana fall skedde. Upphållstiden beräknades därför för de olika inströmningshastigheterna. Dammens längd approximerades till 300 meter. Med denna metod klassades det enbart om det hann sedimentera i dammen eller ej.

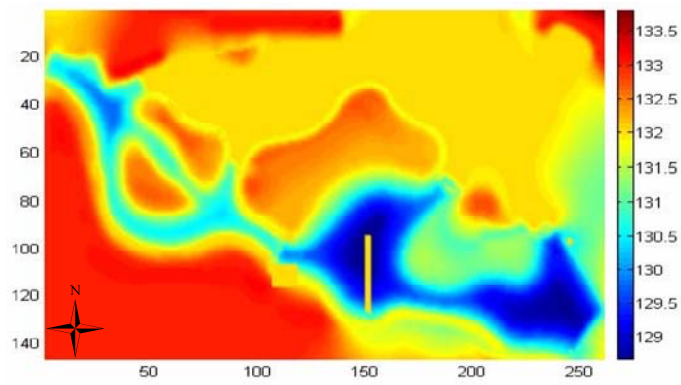
### 3.7 MANIPULATION AV BOTTENTOPOGRAFI

För optimal sedimentation ska vattnet flöda långsamt och över en lång sträcka. För att undersöka om sedimentationsegenskaperna i Vattenparken kunde förbättras skapades två olika manipulationer av bottenprofilen vilka sedan kördes flödessimuleringar över. Manipulationerna syftade till att få vattnet att ta en längre väg genom dammen och på så sätt öka sedimentationen.

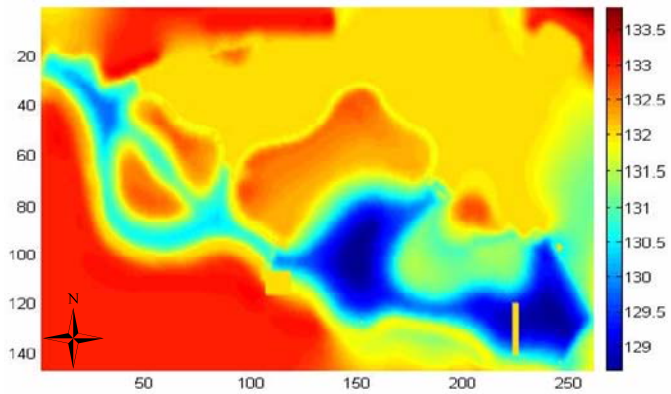
De olika manipuleringar som utfördes var:

- En vall skapades i det djupa området mellan Ö3 och Ö4, mot dammens nedre kant.
- En vall skapades i det djupa området mellan Ö4 och utloppet, mot dammens nedre kant.

För att jämföra om manipulationerna förbättrade sedimentationsegenskaperna i dammen utfördes sedimentationsanalysen även på den manipulerade bottenprofilen. Figur 10 och 11 visar den framinterpolerade bottenprofilen efter de två olika manipulationerna. Axlarna visar längder i meter. Färgskalan till höger visar höjddata i meter.



*Figur 10.* Den framinterpolerade bottenpografin efter manipulation 1. Notera vällen i zon 5.

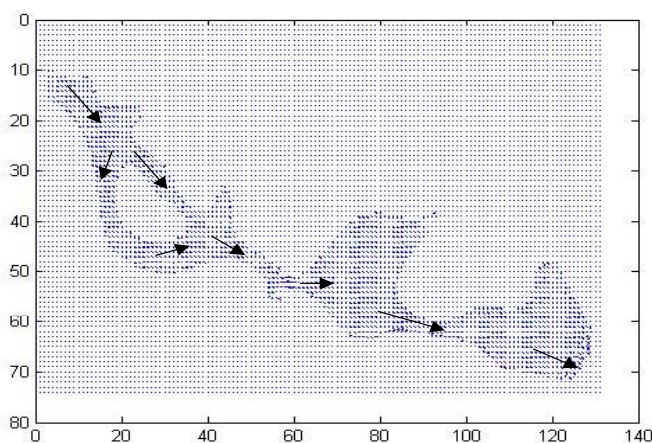


*Figur 11.* Den framinterpolerade bottenpografin efter manipulation 2. Notera vällen i zon 6.

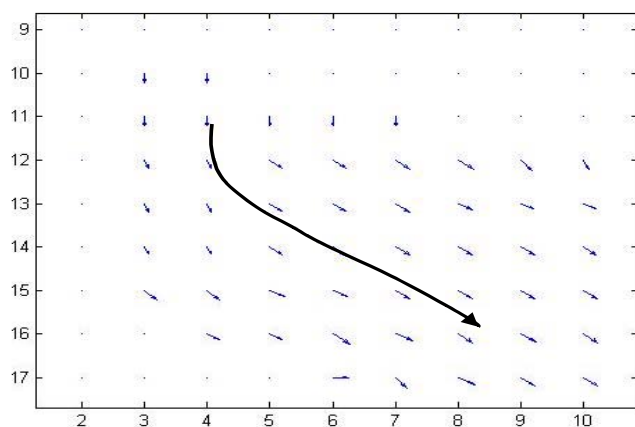


## 4 RESULTAT

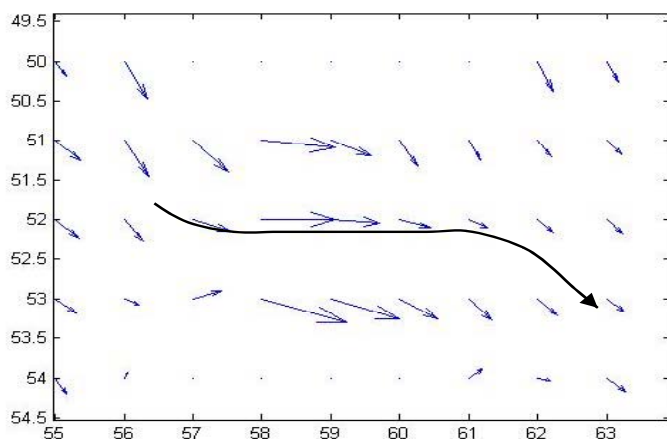
Simuleringar med den explicita metoden blev snabbt instabila och något resultat från dessa kunde därför inte genereras. Med den semiimplicita metoden blev simuleringarna inte heller helt stabila. Instabiliteterna visade sig uppstå vid överfallet i mitten av dammen och sedan sprida sig över hela dammen. Figur 12 visar en illustration på hur strömningshastigheten i dammen såg ut innan modellen blev instabil. Figur 13-15 visar förstoringar av samma bild vid inlopp, överfall och utlopp. De större svarta pilarna är tillsatta i efterhand och förtydligar endast strömningens väg.



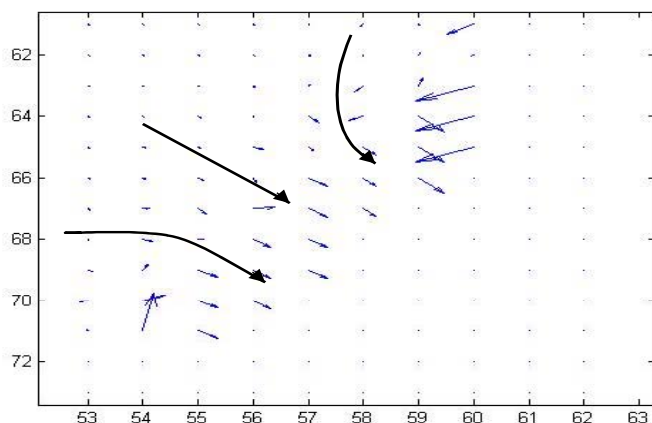
Figur 12. Vattenströmning i dammen efter flödessimulering med inströmningshastighet 0,12 m/s innan divergens uppstod (2800 steg, 1,4 sekunder).



Figur 13. Förstoring av vattenströmning vid inlopp innan divergens uppstod.



Figur 14. Förstoring av vattenströmning vid överfall innan divergens uppstod.



Figur 15. Förstoring av vattenströmning vid utlopp innan divergens uppstod.

Simuleringar med endast vänstra delen av dammen utfördes också, för att undersöka om stabilitet kunde uppnås med överfallet borttaget, men dessa visade sig också ge instabila resultat. Det gick inte visuellt att avgöra exakt vilken beräkningspunkt som orsakade instabiliteten, utan flera punkter blev instabila. Detta för att hela systemet är kopplat och alla punkter påverkar varandra. Resultaten ansågs dock visa på ett troligt scenario och därför utfördes ändå sedimentationsanalysen. Zonerna klassades enligt tidigare beskrivning för modellen innan den började divergera. Dessa resultat kan inte anses som vetenskapligt korrekta, men de redovisas ändå nedan.

#### 4.1 RESULTAT METOD 1 – HJULSTRÖMS DIAGRAM

I bilaga 1 redovisas vad varje zon klassades som för de olika inströmnings-hastigheterna. **Sed** står för sedimentationszon, **Transp** för transportzon och **Resp** för resuspensionszon. Partikelstorleken sattes till finmo i zon 1-3 och finmjäla i zon 4-6. Som påpekats tidigare kan dessa resultat inte anses vara vetenskapligt korrekta då modellen inte uppnådde stabilitet. Resultatet visar att sedimentation inte skedde för någon inströmningshastighet, utan det förekom endast transport- och resuspensionszoner.

## **4.2 RESULTAT METOD 2 – BERÄKNING AV SEDIMENTATIONSHASTIGHET**

Det har tidigare approximativt beräknats att vattnets uppehållstid i dammen behövde vara cirka 131 minuter för att sedimentation ska hinna ske. Uppehållstiden beräknades därför för varje inströmningshastighet och det avgjordes sedan om partiklarna hann sedimentera eller ej. Partikelstorleken beräknades till ett medelvärde av storleken för finmjåla och storleken för finmo. Resultatet som redovisas i Bilaga 2 visar att sedimentation inte hinner ske för någon inströmningshastighet i dammen.

## 5 DISKUSSION OCH SLUTSATSER

Den inledande delen av arbetet med att interpolera fram en bottenpografi fungerade problemfritt. I jämförelse med en höjdtopografikarta över dammen stämde dessa väl överens. Att importera bottenpografien till Matlab visade sig också fungera bra. En möjlig förklaring till att interpolering med Topogrid lyckades, men inte med Topo To Raster, är att i Arc-fönstret utförs bara beräkningar och inga illustrationer visas parallellt. I övrigt är de två metoderna identiska så detta var den enda förklaring som tycktes möjlig.

Shallow Water Ekvationerna som användes i modellen beräknar endast flödet i två dimensioner. De approximerar strömningshastigheten att vara lika stor på alla djup för en viss beräkningspunkt. Detta måste anses orimligt, men förenklar beräkningarna betydligt. Att en fix strandlinje ansattes vilket inget vatten fick flöda över måste även det anses orimligt. Shallow Water Ekvationerna är å andra sidan väl beprövade ekvationer som används i många liknande sammanhang och måste därmed ändå anses kunna ge användbara och rimliga resultat. En ändring av de numeriska approximationerna i ekvationerna skulle eventuellt kunna ge en mer stabil modell. Till exempel skulle en dissipationsterm kunna adderas innehållande andraderivatet för att dämpa instabiliteter.

Den numeriska metod som användes i modellen var inte fullt implicit. Detta kan vara en orsak till instabiliteterna. Om en fullt implicit metod använts kan resultatet ha blivit annorlunda, men en fullt implicit modell skulle även den ha kunnat visa på instabiliteter. Detta på grund av att Shallow Water Ekvationerna är icke-linjära i  $u$ ,  $v$  och  $h$  och de kan därför divergera trots att metoden är implicit. Instabiliteterna kan även bero på att randvillkoren är så pass komplicerade. Ett sätt att visa att differentieringen med C-N är stabil skulle kunna vara att ta bort öarna och ansätta periodiska randvillkor.

För att försöka motverka instabiliteterna minskades tidssteget. Detta visade sig inte hjälpa. För att undersöka om det var överfallet som orsakade instabiliteterna simulerades strömningshastigheten endast för den vänstra sidan av dammen innan överfallet. Detta gav inte heller stabilitet, vilket tolkades som att det var icke-linjäriteterna i Shallow Water Ekvationerna eller de komplicerade randvillkoren som var källan till instabiliteterna. För att förbättra modellen skulle ekvationerna kunna bytas ut mot till exempel Saint Venant Ekvationerna tillsammans med kontinuitetsekvationen. De är också väl beprövade ekvationer i samband med flödessimulering och beräknar även de flödet i två dimensioner och med en fix strandlinje.

Många förenklingar har gjorts i modellen. För att snabba upp modellen har till exempel varannan beräkningspunkt i både  $x$ - och  $y$ -led tagits bort. Detta gav att bottenpografien blev mindre noggrann, men modellen betydligt snabbare. Modellen tar inte heller hänsyn till några yttre påverkningar såsom nederbörd eller utbyte med grundvattnet. Vid en eventuellt fungerande modell bör dessa parametrar tas med i beräkningarna vid fortsatt arbete. Det bör även tas med i beräkningarna hur mycket partikulärt material som kommer in i dammen och hur stor mängd som sedimenterar. Vid en viss mängd sedimenterat material börjar detta resuspenderas igen och

muddring bör ske innan detta sker. Ingen av dessa förenklingar anses dock vara källan till instabiliteterna i modellen utan de gör bara en fungerande modell mindre noggrann.

Vid torkning och analys av sedimentprover antogs en partikelstorlek vara representativ för en hel zon i dammen, nämligen finmo i zon 1-3 och finmjäla i zon 4-6. Sedimentet bestod egentligen av flera olika partikelstorlekar men för att förenkla sedimentationsanalysen gjordes detta antagande vilket inte gav helt rätt bild av sedimentfördelningen i dammen, men ändå en viss fingervisning.

Då modellen inte blev stabil kan inga av dess resultat tolkas som användbara. Sedimentationsanalysen utfördes ändå för att detta moment inte skulle bli helt utelämnat. Analysen visade att sedimentation inte sker för någon inströmningshastighet. Trots att dessa resultat inte kan anses användbara visar de ändå på att det troligtvis sedimenterar väldigt lite material i dammen. Detta beror till stor del av att de partiklar som strömmar in i dammen är så pass små (finmjäla-finmo). För att dessa ska kunna sedimentera krävs, enligt Hjulströms diagram, väldigt låg strömningshastighet. Enligt diagrammet sedimenterar finmjäla inte för någon strömningshastighet och finmo för väldigt låga ( $<0,4$  cm/s). Finmo resuspenderas också för lägre hastigheter ( $>5$  cm/s) än övriga partikelstorlekar och kan då transporteras ut ur dammen. Vid sedimentprovtagning påträffades inga större partiklar än finmo. Om inga större partiklar än så infinner sig i dammen krävs väldigt låga strömningshastigheter för att något bottensediment överhuvudtaget ska bildas. Sand, till exempel, sedimenterar vid en strömningshastighet på cirka 1 cm/s vilket i detta fall hade lett till sedimentation vid de lägre inströmningshastigheterna.

Metod 2 visar knappt någon skillnad av uppehållstid i dammen när vallarna läggs in. Detta är troligtvis på grund av att inte nog många tidssteg hinner genomföras för att skillnad ska hinna uppstå. Upphållstiden borde bli längre med vallar inlagda.

Ingen validering av strömningshastigheten har utförts. Om modellen blivit stabil skulle detta varit en nödvändighet för att kunna tolka resultatet. Hastigheten borde mätas upp i olika punkter i dammen för olika inflöden och sedan jämföras med simulerade hastigheter. Även en massbalans borde ha genomförts vid en fungerande modell. Vid stationära förhållanden ska inflöde, utflöde och flödet över ett tvärsnitt i dammen vara lika stort.

Den konstruerade modellen i detta arbete fungerade förutom instabiliteterna enligt avsikt och borde därför kunna utgöra en grund för fortsatt arbete med flödes-simuleringar. Den går dock inte att använda i nuvarande skick för att simulera vattenströmning, då lösningen inte blir stabil. Några förslag till förbättringar av modellen nämns nedan.

- Använda fullt implicit metod.
- Addera dissipationsterm för att dämpa instabiliteter.
- Använda andra ekvationer än SWE.
- Inkludera in- och utläckage till grundvattnet i beräkningarna.
- Inkludera typ av botten i beräkningarna, exempelvis friktionsfaktor.
- Inkludera mängd inkommande partikulärt material i beräkningarna.
- Använda en mer noggrann metod för sedimentationsanalys.

## 6 REFERENSER

### Litteratur

Baban, R., (1995). *Design of Diversion Weirs -Small Scale Irrigation in Hot Climates*, John Wiley & Sons Ltd, West Sussex, ISBN 0 471 95211 7.

Eklundh, L., (2001). *Geografisk informationsbehandling, metoder och tillämpningar*, Andra upplagan, Byggeforskningsrådet, Stockholm, ISBN 91-540-5841 4.

Fox, R.W., McDonald, A.T., (1994). *Introduction to Fluid Mechanics*, Fourth Edition, John Wiley and Sons, Inc., USA, ISBN 0-471-59274-9.

Färm, C., (2003). *Rening av dagvatten genom filtrering och sedimentation*, VA-Forsk rapport nr 2003-16, VA-Forsk, Svenskt Vatten AB, Stockholm.

George, D. L., (2004). *Numerical Approximation of the Nonlinear Shallow Water Equations with Topography and Dry Beds: A Godunov-Type Scheme*, A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science, University of Washington.

Heath, M. T., (2002). *Scientific Computing; An Introductory Survey*, Second Edition, University of Illinois at Urbana-Champaign, The McGraw-Hill Companies, New York, ISBN 0-07-112229-X.

Lindgren, Å., Svenson, T. (Kontaktpersoner), (2003), *Vägdagvattendammar – En undersökning av funktion och reningseffekt*, Publikation 2003:188, Enheten för Samhälle och trafik, Teknikavdelningen, Vägverket, Borlänge, ISSN 1401-9612

McDonald, A., (2003). *Sensitivity and Robustness of Transparent Boundary Conditions*, Met Éireann, Dublin.

Pellant, C., (1990). *Rocks, Minerals & Fossils of the World*, Pan Books, London, ISBN 0 330 29953 0.

Pärt-Enander, E., Sjöberg, A., (1999). *Användarhandledning för Matlab 5*, Uppsala Universitet, Elanders Gotab, Stockholm, ISBN 91-506-1326-X.

Randall, D. A., (2004). *The Shallow Water Equations*, Department of Atmospheric Science, Colorado State University, Fort Collins, Colorado 80523.

Österman, J., Nordling, C., (1987). *Physics Handbook: Elementary Constants and Units, Tables, Formulae and Diagrams and Mathematical Formulae*, Studentlitteratur, Lund, ISBN 91-44-16574-9.

## Internet

Biosystems and Agricultural Engineering –University of Kentucky (2004). Density and Specific Gravity, <http://www.bae.uky.edu/~snokes/BAE549thermo/physicalproperties/density.htm> (12/12 2005)

Borlänge energi (2003). Vattenparken, <http://www.borlange-energi.se/index.php?PHPSESSID=3de95c74335b883590059cbac1c49373&L VLC=799> (23/8 2005)

Eniro Sverige AB (2005). Kartor, <http://www.eniro.se> (29/11 2005)

Flow Science (2005). Implicit Versus Explicit Methods, <http://www.flow3d.com/Cfd-101/impvexp.htm> (11/11 2005)

Geophysical Fluid Dynamics Laboratory (2005). The Shallow Water Equations, [http://www.gfdl.noaa.gov/fms/pubrel/j/atm\\_dycores/src/atmos\\_spectral\\_shallow/shallow.pdf](http://www.gfdl.noaa.gov/fms/pubrel/j/atm_dycores/src/atmos_spectral_shallow/shallow.pdf) (26/8 2005)

Greppa Näringen (2005). Sedimentationsfällor, <http://www.greppa.nu/kunskapen/uppslagsboken/renavattnet/sedimentation/sedimentationsfallor.4.1d59d3cf8019a445f7fff16384.html> (8/11 2005)

Physical Geology Laboratory (2000). Hjulstrom's diagram, <http://www.gly.fsu.edu/~holm/2010/Hjulstrom.html> (8/11 2005)

Project Euclid (2005), The Cauchy problem for viscous shallow water equations, <http://projecteuclid.org/Dienst/UI/1.0/Summarize/euclid.rmi/1114176224> (22/2 2006)

Science Direct (2005), Computations of shallow water equations with high-order central-upwind schemes on triangular meshes. [http://www.sciencedirect.com/science?\\_ob=ArticleURL&\\_udi=B6TY8-4FC3S48-4&\\_coverDate=11%2F01%2F2005&\\_alid=369167616&\\_rdoc=1&\\_fmt=&\\_orig=search&\\_qd=1&\\_cdi=5612&\\_sort=d&\\_view=c&\\_acct=C000050221&\\_version=1&\\_urlVersion=0&\\_userid=10&md5=97fb5af8aafd57af33536d25ec71c675](http://www.sciencedirect.com/science?_ob=ArticleURL&_udi=B6TY8-4FC3S48-4&_coverDate=11%2F01%2F2005&_alid=369167616&_rdoc=1&_fmt=&_orig=search&_qd=1&_cdi=5612&_sort=d&_view=c&_acct=C000050221&_version=1&_urlVersion=0&_userid=10&md5=97fb5af8aafd57af33536d25ec71c675) (22/2 2006)

Uppsala Universitet (2006). Klickbara kartor, <http://www.uu.se/Univ/Sweden.swe.html> (13/2 2006)

Wolfram Research (2005), Navier-Stokes Equations, <http://scienceworld.wolfram.com/physics/Navier-StokesEquations.html> (24/8 2005)

## Personliga meddelanden

Johansson Leif, 2005. Borlänge energi, Borlänge.

Junestedt Christian, 2005. IVL Svenska Miljöinstitutet AB, Stockholm.

Rantakokko Jarmo, 2005. universitetslektor vid institutionen för informationsteknik,  
Teknisk databehandling, Uppsala Universitet.

### **Programvara**

ArcView, ArcGIS Desktop Help, Dictionary.



## BILAGA 1 – RESULTAT AV METOD 1 – HJULSTRÖMS DIAGRAM

Tabell 2. Klassning av zoner för olika inströmningshastigheter.

| Inflöde (m <sup>3</sup> /s) | Zon 1  | Zon 2 | Zon 3  | Zon 4  | Zon 5  | Zon 6  |
|-----------------------------|--------|-------|--------|--------|--------|--------|
| 5·10 <sup>-3</sup>          | Transp | Resp  | Transp | Transp | Transp | Transp |
| 5·10 <sup>-2</sup>          | Resp   | Resp  | Resp   | Resp   | Transp | Transp |
| 0,5                         | Resp   | Resp  | Resp   | Resp   | Transp | Transp |
| 5                           | Resp   | Resp  | Resp   | Resp   | Resp   | Resp   |

Tabell 3. Klassning av zoner för olika inströmningshastigheter efter manipulation 1.

| Inflöde (m <sup>3</sup> /s) | Zon 1 | Zon 2 | Zon 3 | Zon 4  | Zon 5  | Zon 6  |
|-----------------------------|-------|-------|-------|--------|--------|--------|
| 5·10 <sup>-3</sup>          | Resp  | Resp  | Resp  | Transp | Transp | Transp |
| 5·10 <sup>-2</sup>          | Resp  | Resp  | Resp  | Transp | Transp | Transp |
| 0,5                         | Resp  | Resp  | Resp  | Resp   | Transp | Transp |
| 5                           | Resp  | Resp  | Resp  | Resp   | Resp   | Resp   |

Tabell 4. Klassning av zoner för olika inströmningshastigheter efter manipulation 2.

| Inflöde (m <sup>3</sup> /s) | Zon 1  | Zon 2 | Zon 3 | Zon 4  | Zon 5  | Zon 6  |
|-----------------------------|--------|-------|-------|--------|--------|--------|
| 5·10 <sup>-3</sup>          | Resp   | Resp  | Resp  | Transp | Transp | Transp |
| 5·10 <sup>-2</sup>          | Resp   | Resp  | Resp  | Transp | Transp | Transp |
| 0,5                         | Transp | Resp  | Resp  | Resp   | Transp | Transp |
| 5                           | Resp   | Resp  | Resp  | Resp   | Resp   | Resp   |

Tabell 5. Klassning av zoner för olika inströmningshastigheter efter manipulation 1 och 2 kombinerat.

| Inflöde (m <sup>3</sup> /s) | Zon 1  | Zon 2 | Zon 3 | Zon 4  | Zon 5  | Zon 6  |
|-----------------------------|--------|-------|-------|--------|--------|--------|
| 5·10 <sup>-3</sup>          | Resp   | Resp  | Resp  | Transp | Transp | Transp |
| 5·10 <sup>-2</sup>          | Resp   | Resp  | Resp  | Transp | Transp | Transp |
| 0,5                         | Transp | Resp  | Resp  | Resp   | Transp | Transp |
| 5                           | Resp   | Resp  | Resp  | Resp   | Resp   | Resp   |

5·10<sup>-3</sup> m<sup>3</sup>/s motsvarade en inströmningshastighet i modellen på 1,2·10<sup>-3</sup> m/s, 5·10<sup>-2</sup> m<sup>3</sup>/s motsvarade 1,2·10<sup>-2</sup> m/s, 0,5 m<sup>3</sup>/s motsvarade 0,12 m/s och 5 m<sup>3</sup>/s motsvarade 1,2 m/s.

## BILAGA 2 – RESULTAT AV METOD 2 –BERÄKNING AV SEDIMENTATIONSHASTIGHET

Tabell 6. Dammens ungefärliga uppehållstid samt uppgift om partiklar hinner sedimentera eller ej. Upphållstiden är avrundad till närmaste hundratal.

| Inflöde (m <sup>3</sup> /s) | Ingen manipulation | Manipulation 1   | Manipulation 2   | Manipulation 1 och 2 |
|-----------------------------|--------------------|------------------|------------------|----------------------|
| 5·10 <sup>-3</sup>          | 2800 s / ej sed.   | 2800 s / ej sed. | 2500 s / ej sed. | 2500 s / ej sed.     |
| 5·10 <sup>-2</sup>          | 2900 s / ej sed.   | 2900 s / ej sed. | 2500 s / ej sed. | 2500 s / ej sed.     |
| 0,5                         | 2100 s / ej sed.   | 2100 s / ej sed. | 1900 s / ej sed. | 1900 s / ej sed.     |
| 5                           | 200 s / ej sed.    | 200 s / ej sed.  | 200 s / ej sed.  | 190 s / ej sed.      |

### BILAGA 3 – MATLABFILER

Tabell 7. De Matlabfiler som använts och deras funktion.

| <b>Huvudkod:</b>     | <b>Funktion:</b>                                |
|----------------------|---|
| Implicitdiffwhile2.m | Importerar bottentopografi och beräknar SWE.    |
| <b>Underkod:</b>     |   |
| manip1.m             | Lägger till den första vallen efter överfallet. |
| manip2.m             | Lägger till den andra vallen innan utloppet.    |
| Matriser.m           | Definierar matriserna i SWE.                    |
| yttreander.m         | Definierar yttre randvillkor.                   |
| Rander.m             | Definierar ränder för inlopp och utlopp.        |
| randvillkor.m        | Definierar också ränder för inlopp och utlopp.  |
| inreranderspecial2.m | Definierar ränder för öar och strandkanter.     |
| TempMatriser.m       | Definierar matriser inför andra steget i SWE.   |

## BILAGA 4 – HUVUDKOD

### Implicitdiffwhile.m

%Flödessimulering, Implicit differentiering av SWE

```
%%%%%%INLÄSNING AV
BOTTENPROFIL%%%%%%%%%%
imread lm4_int1.tif;      %läser in bottenprofilen
Hb100=ans;                %ger den ett namn
Hb100_2=double(Hb100);   %gör om
bottenprofilen till en double array
Hb1=Hb100_2/100;        %dividerar med 100 för
att få verkliga värden

%Korrigerar höjden i vissa mätpunkter på landområden
%så att det tex inte finns en ensam landpunkt omgiven av
vatten
Hb1(39,59)=132;
Hb1(40,58:60)=132;
Hb1(41,58:61)=132;
Hb1(42,59:62)=132;
Hb1(43,60:63)=132;
Hb1(44,61:64)=132;
Hb1(45,62:65)=132;
Hb1(46,63:64)=132;
Hb1(51,82)=132;
Hb1(52,81:83)=132;
Hb1(53,81:82)=132;
Hb1(54,85)=132;
Hb1(55,84:86)=132;
Hb1(56,84:85)=132;
Hb1(95,111)=132;
Hb1(95:96,112)=132;
Hb1(94:96,113:114)=132;
Hb1(94:97,115)=132;
Hb1(94:98,116)=132;
Hb1(73,187)=132;
Hb1(143,245)=132;
Hb1(127,152)=132;
Hb1(36,49)=132;
Hb1(64,86)=132;
Hb1(73,187)=132;
Hb1(96,244:245)=132;
Hb1(97,244:246)=132;
Hb1(98,244:245)=132;

% manip1      %anropar fil med forsta manipulationen
% manip2      %anropar fil med andra manipulationen

%använder endast varannan punkt i x-led och y-led
Hb1=Hb1(1:2:147,1:2:261);
Hb1(62,84)=130.2;
Hb1(14,1)=132;
Hb1(63:64,130)=132;
Hb1(41,35)=131.7;
Hb1(11:15,2)=132;
Hb1(16,20:23)=132;

[antalrader,antalkolumner]=size(Hb1); %definierar
antalrader som antal rader i matrisen och antalkolumner
%som antal kolumner i matrisen
antaltermen=antalrader*antalkolumner; %Beräknar totalt
antal termen i matrisen

%%%%%%%%%%KONSTANTER%%%%%%%%%%
inflode=5*10^(-3);
inflode=0.012;%*10^(-3); %Inflödeshastigheten definieras
```

```
utflode=inflode/7.89;
utflodevektor=((utflode^2)/2)^(0.5);
dt=0.005; %tidssteg
dx=2; %avstånd mellan mätpunkter i x-led
dy=2; %avstånd mellan mätpunkter i y-led
g=9.81; %gravitationskonstant
alpha=1/dt; %konstant i matris
kappa=g/dx; %konstant i matris
eta=g/dy; %konstant i matris
x=115/2-0.5; %x-koordinat vid överfall där höjden
minskas en meter och 25 cm
eps=zeros(10,1); %sätter eps så att den går in i while-
loopen första gången
```

```
%%%Specifika parametrar för just Vattenparken
strandkant1=131.75;
strandkant2=130.5;
```

```
%Sänker områden som är för grunda så att det blir en
skarpare rand land/vatten
for j=2:antalrader-1
    for i=2:x-1
        if Hb1(j,i)<strandkant1&&Hb1(j,i)>(strandkant1-0.3)
            Hb1(j,i)=strandkant1-0.3;
        end
    end
end
for j=2:antalrader-1
    for i=x:antalkolumner-1
        if Hb1(j,i)<strandkant2&&Hb1(j,i)>(strandkant2-0.3)
            Hb1(j,i)=strandkant2-0.3;
        end
    end
end
```

```
Hb2=Hb1'; %transponerar Hb så att uttrycket nedanför
blir rätt
Hb=Hb2(:); %gör om Hb till en kolonnvektor med alla
rader liggandes efter varandra
%nödvändigt för beräkningar nedan
```

```
%%%INITIALVÄRDEN%%%%%%%%%%
%initialgissning på h är olika före och efter överfall
h=zeros(antaltermen,1);
h2=zeros(antaltermen,1);
for j=1:antalrader
    for i=(j-1)*antalkolumner+1:j*antalkolumner-x+2
        h(i)=131.76;
    end
end
for j=1:antalrader
    for i=(j-1)*antalkolumner+x+1:j*antalkolumner
        h(i)=130.51;
    end
end
```

```
%initialgissningar på de övriga obekanta som ska beräknas
u=zeros(antaltermen,1)+inflode;
v=zeros(antaltermen,1)+inflode;
u2=zeros(antaltermen,1);
v2=zeros(antaltermen,1);
my1=zeros(antaltermen,1);
my2=zeros(antaltermen,1);
my3=zeros(antaltermen,1);
my4=zeros(antaltermen,1);
b1=zeros(antaltermen,1);
b2=zeros(antaltermen,1);
bh=zeros(antaltermen,1);
```

```

c1=zeros(antalterm,1);
c2=zeros(antalterm,1);

%% %% %% STARTAR BERÄKNINGAR I
TIDSLOOP %% %% %% %% %% %% %% %%

%räknar först 10 steg innan stabilitetsvillkoret sätts
for n=1:10 %räknare
n %skriver ut vilket tidssteg det är

%Paramterar i matriserna som ändras i varje tidssteg
beta=u/(4*dx);
gamma=v/(4*dy);

%% %% %% Definierar parametern my i matriserna, alla termer
i my kan inte
%% %% %% beräknas för alla platser i matrisen varpå my delas
upp i fyra delar
for j=1:antalrader
for i=(1+(j-1)*antalkolumner):(j*antalkolumner-1)
my1(i)=u(i+1)/(2*dx);
end
end
for j=1:antalrader
for i=(2+(j-1)*antalkolumner):(j*antalkolumner)
my2(i)=u(i-1)/(2*dx);
end
end
for i=1:(antalterm-antalkolumner)
my3(i)=v(i+antalkolumner)/(2*dy);
end
for i=(1+antalkolumner):antalterm
my4(i)=v(i-antalkolumner)/(2*dy);
end
my=my1-my2+my3-my4; %my beräknas

%% %% %% DIAGONALTILDELNING I
MATRISER %% %% %%
Matriser %anropar separat fil där matriserna
definieras

%% %% %% RANDVILLKOR %% %% %%
yttreander %anropar separat fil med de yttre
randvillkoren
Rander %anropar en separat fil med de inre
randvillkoren för inlopp och utlopp
inreranderspecial2 %anropar en separat fil med
randvillkoren för ör och landområden

%% %% %% %% BERÄKNAR SWE, steg
1 %% %% %% %% Randvillkor inkluderade
utemp=A1*(B1*u+b1+C1*h+c1);
vtemp=A2*(B2*v+b2+C2*h+c2);
htemp=AH*(BH*h+bh+CH*Hb);

%beräknar nya matriser A1, A2 och AH at använda i nästa
steg
TempMatriser

%% %% %% %% BERÄKNAR SWE, steg 2 %% %% %% %%
u2=A1temp*(B1*u+b1+C1*((htemp+h)/2)+c1);
v2=A2temp*(B2*v+b2+C2*((htemp+h)/2)+c2);
h2=AHtemp*(BH*((htemp+h)/2)+bh+CH*Hb);

%beräkna epsilon
eps(10)=norm(u-u2)/norm(u)
%flyttar efter värdet på epsilon så att de tio senast
uträknade finns i eps
for i=1:9
eps(i)=eps(i+1);

```

```

end

%sätter det nyuträknade värdena i SWE till det gamla
u=u2;
v=v2;
h=h2;

end %avslutar tidssteg

%% %% %% HÄR ÖVERGÅR RÄKNINGARNA FRÅN
ETT TIDSBESTÄMT ANTAL STEG %% %% %%
%% %% %% TILL ATT UPPFYLLA
STABILITETSVILLKORET %% %% %% %% %% %% %% %%
%% %% %% %% %%

eps(10)=0.003;

%% %% %% NY WHILE-LOOP %% %% %% beräknar samma sak
som i tidigare fast nu med stabilitetsvillkor
%så länge eps>0.001 fortsätter beräkningarna

while eps(10)>0.0001

n=n+1 %räknar upp tidssteg

%Paramterar i matriserna som ändras i varje tidssteg
beta=u/(4*dx);
gamma=v/(4*dy);

%% %% %% Definierar parametern my i matriserna, alla termer i
my kan inte
%% %% %% beräknas för alla platser i matrisen varpå my delas
upp i fyra delar
for j=1:antalrader
for i=(1+(j-1)*antalkolumner):(j*antalkolumner-1)
my1(i)=u(i+1)/(2*dx);
end
end
for j=1:antalrader
for i=(2+(j-1)*antalkolumner):(j*antalkolumner)
my2(i)=u(i-1)/(2*dx);
end
end
for i=1:(antalterm-antalkolumner)
my3(i)=v(i+antalkolumner)/(2*dy);
end
for i=(1+antalkolumner):antalterm
my4(i)=v(i-antalkolumner)/(2*dy);
end
my=my1-my2+my3-my4; %my beräknas

%% %% %% DIAGONALTILDELNING I
MATRISER %% %% %% %%
Matriser %anropar separat fil där matriserna definieras

%% %% %% RANDVILLKOR %% %% %%
yttreander %anropar separat fil för de yttre randvillkoren
Rander %anropar separat fil för inlopp och utlopp
inreranderspecial2 %anropar separat för randvillkor
land/vatten

%% %% %% %% BERÄKNAR SWE, steg
1 %% %% %% %% Randvillkor inkluderade
utemp=A1*(B1*u+b1+C1*h+c1);
vtemp=A2*(B2*v+b2+C2*h+c2);
htemp=AH*(BH*h+bh+CH*Hb);

%beräknar nya matriser A1, A2 och AH at använda i nästa
steg
TempMatriser

```

```

%% %% %% BERÄKNAR SWE, steg 2 %% %% %%
u2=A1temp/(B1*u+b1+C1*((htemp+h)/2)+c1);
v2=A2temp/(B2*v2+b2+C2*((htemp+h)/2)+c2);
h2=AHtemp/(BH*((htemp+h)/2)+bh+CH*Hb);

for i=1:9
    eps(i)=eps(i+1);
end

eps(10)=norm(u-u2)/norm(u)    %beräknar epsilon som
används i stabilitetsvillkoret

%sätter det nyuträknade värdet till det gamla
u=u2;
v=v2;
h=h2;

end % avslutar tidssteg

%konverterar om kolonnvektorena till matriser igen för
att kunna tolka
%resultatet
u=(reshape(u,antalkolumner,antalrader));
v=(reshape(v,antalkolumner,antalrader));
h=(reshape(h,antalkolumner,antalrader));

```

## BILAGA 5 - UNDERKOD

### manip1.m

```
%tillverkar den första vällen i zon 5
```

```
%tilverkar vällen
```

```
Hb1(95:126,150:154)=132;
```

### manip2.m

```
%tillverkar den andra vällen i zon 6
```

```
Hb1(120:140,223:225)=132;
```

### Matriser.m

```
%tilldelar diagonaler i matriser
```

```
%%%%%%%%DEFINIERAR MATRISERNA%%%%%%%%
```

```
%glesa matriser, endast värden där det är element
```

```
A1=sparse(antalterm,antalterm);
```

```
B1=sparse(antalterm,antalterm);
```

```
BH=sparse(antalterm,antalterm);
```

```
C1=sparse(antalterm,antalterm);
```

```
C2=sparse(antalterm,antalterm);
```

```
CH=sparse(antalterm,antalterm);
```

```
%%tilldelar diagonalelementen
```

```
for i=antalkolumner+1:antalterm-antalkolumner
```

```
    A1(i,i)=alpha;%huvuddiagonalerna
```

```
    B1(i,i)=alpha;
```

```
    BH(i,i)=alpha-my(i);
```

```
    CH(i,i)=my(i);
```

```
    A1(i,i+1)=beta(i);%överdiagonalerna
```

```
    B1(i,i+1)=-beta(i);
```

```
    BH(i,i+1)=-beta(i);
```

```
    C1(i,i+1)=-kappa;
```

```
    CH(i,i+1)=beta(i);
```

```
    A1(i,i-1)=-beta(i);%underdiagonalerna
```

```
    B1(i,i-1)=beta(i);
```

```
    BH(i,i-1)=beta(i);
```

```
    C1(i,i-1)=kappa;
```

```
    CH(i,i-1)=-beta(i);
```

```
A1(i,i+antalkolumner)=gamma(i);%superöverdiagonalerna
```

```
B1(i,i+antalkolumner)=-gamma(i);
```

```
C2(i,i+antalkolumner)=-eta;
```

```
BH(i,i+antalkolumner)=-gamma(i);
```

```
CH(i,i+antalkolumner)=gamma(i);
```

```
A1(i,i-antalkolumner)=-
```

```
gamma(i);%superunderdiagonalerna
```

```
B1(i,i-antalkolumner)=gamma(i);
```

```
C2(i,i-antalkolumner)=eta;
```

```
BH(i,i-antalkolumner)=gamma(i);
```

```
CH(i,i-antalkolumner)=-gamma(i);
```

```
end
```

```
for i=2:antalkolumner
```

```
    A1(i,i)=alpha;%huvuddiagonalerna
```

```
    B1(i,i)=alpha;
```

```
    BH(i,i)=alpha-my(i);
```

```
    CH(i,i)=my(i);
```

```
    A1(i,i+1)=beta(i);%överdiagonalerna
```

```
    B1(i,i+1)=-beta(i);
```

```
    BH(i,i+1)=-beta(i);
```

```
    C1(i,i+1)=-kappa;
```

```
    CH(i,i+1)=beta(i);
```

```
    A1(i,i-1)=-beta(i);%underdiagonalerna
```

```
    B1(i,i-1)=beta(i);
```

```
    BH(i,i-1)=beta(i);
```

```
    C1(i,i-1)=kappa;
```

```
    CH(i,i-1)=-beta(i);
```

```
A1(i,i+antalkolumner)=gamma(i);%superöverdiagonalerna
```

```
B1(i,i+antalkolumner)=-gamma(i);
```

```
C2(i,i+antalkolumner)=-eta;
```

```
BH(i,i+antalkolumner)=-gamma(i);
```

```
CH(i,i+antalkolumner)=gamma(i);
```

```
end
```

```
for i=antalterm-antalkolumner+1:antalterm-1
```

```
    A1(i,i)=alpha;%huvuddiagonalerna
```

```
    B1(i,i)=alpha;
```

```
    BH(i,i)=alpha-my(i);
```

```
    CH(i,i)=my(i);
```

```
    A1(i,i+1)=beta(i);%överdiagonalerna
```

```
    B1(i,i+1)=-beta(i);
```

```
    BH(i,i+1)=-beta(i);
```

```
    C1(i,i+1)=-kappa;
```

```
    CH(i,i+1)=beta(i);
```

```
    A1(i,i-1)=-beta(i);%underdiagonalerna
```

```
    B1(i,i-1)=beta(i);
```

```
    BH(i,i-1)=beta(i);
```

```
    C1(i,i-1)=kappa;
```

```
    CH(i,i-1)=-beta(i);
```

```
A1(i,i-antalkolumner)=-
```

```
gamma(i);%superunderdiagonalerna
```

```
B1(i,i-antalkolumner)=gamma(i);
```

```
C2(i,i-antalkolumner)=eta;
```

```
BH(i,i-antalkolumner)=gamma(i);
```

```
CH(i,i-antalkolumner)=-gamma(i);
```

```
end
```

```
%tilldelar de element som inte blev tilldelade i loopen
```

```
ovan
```

```
i=1;
```

```
A1(i,i)=alpha;
```

```
B1(i,i)=alpha;
```

```
BH(i,i)=alpha-my(i);
```

```
CH(i,i)=my(i);
```

```
A1(i,i+1)=beta(i);
```

```
B1(i,i+1)=-beta(i);
```

```
BH(i,i+1)=-beta(i);
```

```
C1(i,i+1)=-kappa;
```

```
CH(i,i+1)=beta(i);
```

```
A1(i,i+antalkolumner)=gamma(i);
```

```
B1(i,i+antalkolumner)=-gamma(i);
```

```
C2(i,i+antalkolumner)=-eta;
```

```
BH(i,i+antalkolumner)=-gamma(i);
```

```
CH(i,i+antalkolumner)=gamma(i);
```

```
i=antalterm;
```

```
A1(i,i)=alpha;
```

```
B1(i,i)=alpha;
```

```
BH(i,i)=alpha-my(i);
```

```
CH(i,i)=my(i);
```

```
A1(i,i-1)=-beta(i);
```

```
B1(i,i-1)=beta(i);
```

```
BH(i,i-1)=beta(i);
```

```
C1(i,i-1)=kappa;
```

```
CH(i,i-1)=-beta(i);
```

```
A1(i,i-antalkolumner)=-gamma(i);
```

```
B1(i,i-antalkolumner)=gamma(i);
```

```
C2(i,i-antalkolumner)=eta;
```

```
BH(i,i-antalkolumner)=gamma(i);
```

```
CH(i,i-antalkolumner)=-gamma(i);
```

```
%vissa matriser identiska
```

```
A2=A1;
```

```
AH=A1;
```

```
B2=B1;
```

### yttreander.m

```
%%%%%%%%%YTTRE RANDVILLKOR%%%%%%%%%
```

```
%Längst upp i vänstra hörnet, specialfall
```

```

i=1;
b1(i)=-
alpha*u(i)+beta(i)*u(i+1)+gamma(i)*u(i+antalkolumner);
b2(i)=-
alpha*v(i)+beta(i)*v(i+1)+gamma(i)*v(i+antalkolumner);
bh(i)=beta(i)*h(i)+gamma(i)*h(i);
c1(i)=kappa*h(i);
c2(i)=eta*h(i);

%Längst upp i högra hörnet, specialfall
i=antalkolumner;
b1(i)=-alpha*u(i)+beta(i)*(u(i+1)-u(i-
1))+gamma(i)*u(i+antalkolumner);
b2(i)=-alpha*v(i)+beta(i)*(v(i+1)-v(i-
1))+gamma(i)*v(i+antalkolumner);
bh(i)=gamma(i)*h(i)+beta(i)*(h(i+1)-h(i));
c1(i)=kappa*h(i+1)-kappa*h(i);
c2(i)=eta*h(i);

%Längst ner i vänstra hörnet, specialfall
i=antalterm-antalkolumner+1;
b1(i)=-alpha*u(i)+beta(i)*(-u(i-1)+u(i+1))-
gamma(i)*u(i-antalkolumner);
b2(i)=-alpha*v(i)+beta(i)*(-v(i-1)+v(i+1))-
gamma(i)*v(i-antalkolumner);
bh(i)=-gamma(i)*h(i)+beta(i)*(h(i)-h(i-1));
c1(i)=-kappa*h(i-1)+kappa*h(i);
c2(i)=-eta*h(i);

%Längst ner i högra hörnet, specialfall
i=antalterm;
b1(i)=-alpha*u(i)-beta(i)*u(i-1)-gamma(i)*u(i-
antalkolumner);
b2(i)=-alpha*v(i)-beta(i)*v(i-1)-gamma(i)*v(i-
antalkolumner);
bh(i)=-beta(i)*h(i)-gamma(i)*h(i);
c1(i)=-kappa*h(i);
c2(i)=-eta*h(i);

%Övre randen
for i=2:antalkolumner-1;
b1(i)=-alpha*u(i)+beta(i)*(u(i+1)-u(i-1));
b2(i)=-alpha*v(i)+beta(i)*(v(i+1)-v(i-1));
bh(i)=gamma(i)*h(i);
c2(i)=eta*h(i);
end

%Nedre randen
for i=antalterm-antalkolumner+2:antalterm-1
b1(i)=-alpha*u(i)+beta(i)*(u(i+1)-u(i-1));
b2(i)=-alpha*v(i)+beta(i)*(v(i+1)-v(i-1));
bh(i)=-gamma(i)*h(i);
c2(i)=-eta*h(i);
end

%Högra randen
for i=2*antalkolumner:antalkolumner:antalterm-
antalkolumner
b1(i)=-
alpha*u(i)+beta(i)*u(i+1)+gamma(i)*(u(i+antalkolumner)-
u(i-antalkolumner));
b2(i)=-
alpha*v(i)+beta(i)*v(i+1)+gamma(i)*(v(i+antalkolumner)-
v(i-antalkolumner));
end

%Vänstra randen
for i=antalkolumner+1:antalkolumner:antalterm-
2*antalkolumner+1
b1(i)=-alpha*u(i)-beta(i)*u(i-
1)+gamma(i)*(u(i+antalkolumner)-u(i-antalkolumner));
b2(i)=-alpha*v(i)-beta(i)*v(i-
1)+gamma(i)*(v(i+antalkolumner)-v(i-antalkolumner));
end

```

## Rander.m

%% %% SPECIFIKA RÄNDER FÖR VATTENPARKEN  
INLOPP/UTLOPP

%% %% INFLÖDES RANDVILLKOR %% %%

```

j=10;
for i=(j-1)*antalkolumner+3:(j-1)*antalkolumner+4
b1(i)=-alpha*u(i);
b2(i)=alpha*(inflode-v(i));
end
j=11;
for i=(j-1)*antalkolumner+3:(j-1)*antalkolumner+7
b1(i)=-alpha*u(i);
b2(i)=alpha*(inflode-v(i));
end

```

%% %% UTFLÖDES RANDVILLKOR %% %%

for j=70:71

```

i=(j-1)*antalkolumner+124;
%bh(i)=(alpha-my(i))*(hojdkonst-h(i));
b1(i)=alpha*(utflode-u(i));
b2(i)=alpha*(utflode-v(i));
end

```

for j=69:70

```

i=(j-1)*antalkolumner+125;
b1(i)=alpha*(utflode-u(i));
b2(i)=alpha*(utflode-v(i));
end

```

for j=67:69

```

i=(j-1)*antalkolumner+126;
b1(i)=alpha*(utflode-u(i));
b2(i)=alpha*(utflode-v(i));
end

```

for j=66:67

```

i=(j-1)*antalkolumner+127;
b1(i)=alpha*(utflode-u(i));
b2(i)=alpha*(utflode-v(i));
end

```

for j=65:66

```

i=(j-1)*antalkolumner+128;
b1(i)=alpha*(utflode-u(i));
b2(i)=alpha*(utflode-v(i));
end

```

j=65;

i=(j-1)\*antalkolumner+129;

```

b1(i)=alpha*(utflode-u(i));
b2(i)=alpha*(utflode-v(i));

```

i=65\*antalrader+129;

```

b1(i)=-alpha*u(i);
b2(i)=-alpha*v(i);
i=66*antalrader+128;

```

```

b1(i)=-alpha*u(i);
b2(i)=-alpha*v(i);
i=70*antalrader+125;

```

```

b1(i)=-alpha*u(i);
b2(i)=-alpha*v(i);
i=69*antalrader+126;

```

```

b1(i)=-alpha*u(i);
b2(i)=-alpha*v(i);
i=67*antalrader+127;

```

```

b1(i)=-alpha*u(i);
b2(i)=-alpha*v(i);

```

## inreranderspecial2.m

%% %% RANDVILLKOR FÖR GRÄNSEN  
LAND/VATTEN

% inre området av den vänstra delen av dammen, innan  
överfallet,



```

%strandkant1 används
for j=2:antalrader-1
    for i=(j-1)*antalkolumner+2:(j-1)*antalkolumner+x-1
        strandkant=strandkant1;
        if Hb(i)>=strandkant

            randvillkor
            end
        end
    end
end

%inre området för den högra delen av dammen, efter
överfallet,
%strandkant2 används
for j=2:antalrader-1
    for i=(j-1)*antalkolumner+x:j*antalkolumner-1
        strandkant=strandkant2;
        if Hb(i)>=strandkant

            randvillkor
            end
        end
    end
end

```

### randvillkor.m

```

%Land
    if Hb(i-antalkolumner-
1)>=strandkant&Hb(i-antalkolumner)>=strandkant&Hb(i-
antalkolumner+1)>=strandkant...
        &Hb(i+1)>=strandkant&Hb(i-
1)>=strandkant&Hb(i+antalkolumner+1)>=strandkant...

&Hb(i+antalkolumner)>=strandkant&Hb(i+antalkolumner
-1)>=strandkant

        b1(i)=-
alpha*u(i);%+beta(i)*(u(i+1)-u(i-
1))+gamma(i)*(u(i+antalkolumner)-u(i-antalkolumner));
        b2(i)=-
alpha*v(i);%+beta(i)*(v(i+1)-v(i-
1))+gamma(i)*(v(i+antalkolumner)-v(i-antalkolumner));
        bh(i)=(alpha-my(i))*(Hb(i)-h(i));
        end

        %Raka ränder
        if Hb(i-antalkolumner-
1)>=strandkant&Hb(i-antalkolumner)>=strandkant&Hb(i-
antalkolumner+1)>=strandkant...
            &Hb(i+1)>=strandkant&Hb(i-
1)>=strandkant&Hb(i+antalkolumner-1)<strandkant&...

Hb(i+antalkolumner)<strandkant&Hb(i+antalkolumner+1)
<strandkant

            b1(i)=-alpha*u(i);
            b2(i)=-alpha*v(i);

        end
        if Hb(i-
1)>=strandkant&Hb(i+antalkolumner-
1)>=strandkant&Hb(i+antalkolumner)>=strandkant...

&Hb(i+antalkolumner+1)>=strandkant&Hb(i+1)>=strandk
ant&Hb(i-antalkolumner+1)<strandkant...
            &Hb(i-
antalkolumner)<strandkant&Hb(i-antalkolumner-
1)<strandkant

            b1(i)=-alpha*u(i);
            b2(i)=-alpha*v(i);

        end
end

```

```

        if
Hb(i+antalkolumner)>=strandkant&Hb(i+antalkolumner-
1)>=strandkant&Hb(i-1)>=strandkant...
            &Hb(i-antalkolumner-
1)>=strandkant&Hb(i-antalkolumner)>=strandkant...
            &Hb(i-
antalkolumner+1)<strandkant&Hb(i+1)<strandkant&Hb(i
+antalkolumner+1)<strandkant

            b1(i)=-alpha*u(i);
            b2(i)=-alpha*v(i);

        end
        if Hb(i-
antalkolumner)>=strandkant&Hb(i-
antalkolumner+1)>=strandkant&Hb(i+1)>=strandkant...

&Hb(i+antalkolumner+1)>=strandkant&Hb(i+antalkolum
ner)>=strandkant...
            &Hb(i+antalkolumner-
1)<strandkant&Hb(i-1)<strandkant&Hb(i-antalkolumner-
1)<strandkant

            b1(i)=-alpha*u(i);
            b2(i)=-alpha*v(i);

        end

        %Raka ränder vid inåtgående hörn
        if Hb(i+1)>=strandkant&&Hb(i-
antalkolumner+1)>=strandkant&&Hb(i-
antalkolumner)>=strandkant...
            &&Hb(i-antalkolumner-
1)>=strandkant&&Hb(i-
1)>=strandkant&&Hb(i+antalkolumner-1)>=strandkant...

&&Hb(i+antalkolumner)<strandkant&&Hb(i+antalkolum
ner+1)<strandkant

            b1(i)=-alpha*u(i);
            b2(i)=-alpha*v(i);

        end

        if Hb(i+1)<strandkant&&Hb(i-
antalkolumner+1)>=strandkant&&Hb(i-
antalkolumner)>=strandkant...
            &&Hb(i-antalkolumner-
1)>=strandkant&&Hb(i-
1)>=strandkant&&Hb(i+antalkolumner-1)>=strandkant...

&&Hb(+antalkolumner)>=strandkant&&Hb(i+antalkolum
ner+1)<strandkant

            b2(i)=-alpha*v(i);
            b1(i)=-alpha*u(i);

        end

        if Hb(i+1)>=strandkant&&Hb(i-
antalkolumner+1)>=strandkant&&Hb(i-
antalkolumner)>=strandkant...
            &&Hb(i-antalkolumner-
1)>=strandkant&&Hb(i-
1)>=strandkant&&Hb(i+antalkolumner-1)<strandkant...

&&Hb(i+antalkolumner)<strandkant&&Hb(i+antalkolum
ner+1)>=strandkant

            b1(i)=-alpha*u(i);
            b2(i)=-alpha*v(i);

        end
end

```



```

end

% Inåtgående hörn
if
Hb(i+antalkolumner)>=strandkant&Hb(i+antalkolumner-
1)>=strandkant&Hb(i-1)>=strandkant...
    &Hb(i-antalkolumner-
1)>=strandkant&Hb(i-antalkolumner)>=strandkant...
    &Hb(i-
antalkolumner+1)>=strandkant&Hb(i+1)>=strandkant&H
b(i+antalkolumner+1)<strandkant

    b1(i)=-alpha*u(i);
    b2(i)=-alpha*v(i);
    bh(i)=(alpha-
my(i))*(h(i+1)+h(i+antalkolumner))/2-h(i));
end
if Hb(i-1)>=strandkant&Hb(i-
antalkolumner-1)>=strandkant&Hb(i-
antalkolumner)>=strandkant...
    &Hb(i-
antalkolumner+1)>=strandkant&Hb(i+1)>=strandkant&H
b(i+antalkolumner+1)>=strandkant...

&Hb(i+antalkolumner)>=strandkant&Hb(i+antalkolumner
-1)<strandkant

    b1(i)=-alpha*u(i);
    b2(i)=-alpha*v(i);

    bh(i)=(alpha-my(i))*(h(i-
1)+h(i+antalkolumner))/2-h(i));
end
if
Hb(i+1)>=strandkant&Hb(i+antalkolumner+1)>=strandka
nt&Hb(i+antalkolumner)>=strandkant...
    &Hb(i+antalkolumner-
1)>=strandkant&Hb(i-1)>=strandkant&Hb(i-
antalkolumner-1)>=strandkant...
    &Hb(i-
antalkolumner)>=strandkant&Hb(i-
antalkolumner+1)<strandkant

    b1(i)=-alpha*u(i);
    b2(i)=-alpha*v(i);

    bh(i)=(alpha-my(i))*(h(i-
antalkolumner)+h(i+1))/2-h(i));
end
if Hb(i-
antalkolumner)>=strandkant&Hb(i-
antalkolumner+1)>=strandkant&Hb(i+1)>=strandkant...

&Hb(i+antalkolumner+1)>=strandkant&Hb(i+antalkolum
ner)>=strandkant...
    &Hb(i+antalkolumner-
1)>=strandkant&Hb(i-1)>=strandkant&Hb(i-
antalkolumner-1)<strandkant

    b1(i)=-alpha*u(i);
    b2(i)=-alpha*v(i);
    bh(i)=(alpha-my(i))*(h(i-
antalkolumner)+h(i-1))/2-h(i));
end

```

### TempMatriser.m

```

A1temp=sparse(antalterm,antalterm);
A2temp=sparse(antalterm,antalterm);
AHtemp=sparse(antalterm,antalterm);

```

```

for i=antalkolumner+1:antalterm-antalkolumner
    A1temp(i,i)=alpha;% huvuddiagonalerna
    A1temp(i,i+1)=beta(i);% överdiagonalerna
    A1temp(i,i-1)=-beta(i);% underdiagonalerna

```

```

    A1temp(i,i+antalkolumner)=gamma(i);% superöverdiagona
    lerna
    A1temp(i,i-antalkolumner)=-
    gamma(i);% superunderdiagonalerna
end

```

```

for i=2:antalkolumner
    A1temp(i,i)=alpha;% huvuddiagonalerna
    A1temp(i,i+1)=beta(i);% överdiagonalerna
    A1temp(i,i-1)=-beta(i);% underdiagonalerna

```

```

A1temp(i,i+antalkolumner)=gamma(i);% superöverdiagona
lerna
end

```

```

for i=antalterm-antalkolumner+1:antalterm-1
    A1temp(i,i)=alpha;% huvuddiagonalerna
    A1temp(i,i+1)=beta(i);% överdiagonalerna
    A1temp(i,i-1)=-beta(i);% underdiagonalerna
    A1temp(i,i-antalkolumner)=-
    gamma(i);% superunderdiagonalerna
end

```

```

i=1;
A1temp(i,i)=alpha;

```

```

i=antalterm;
A1temp(i,i)=alpha;

```

```

A2temp=A1;
AHtemp=A1;

```