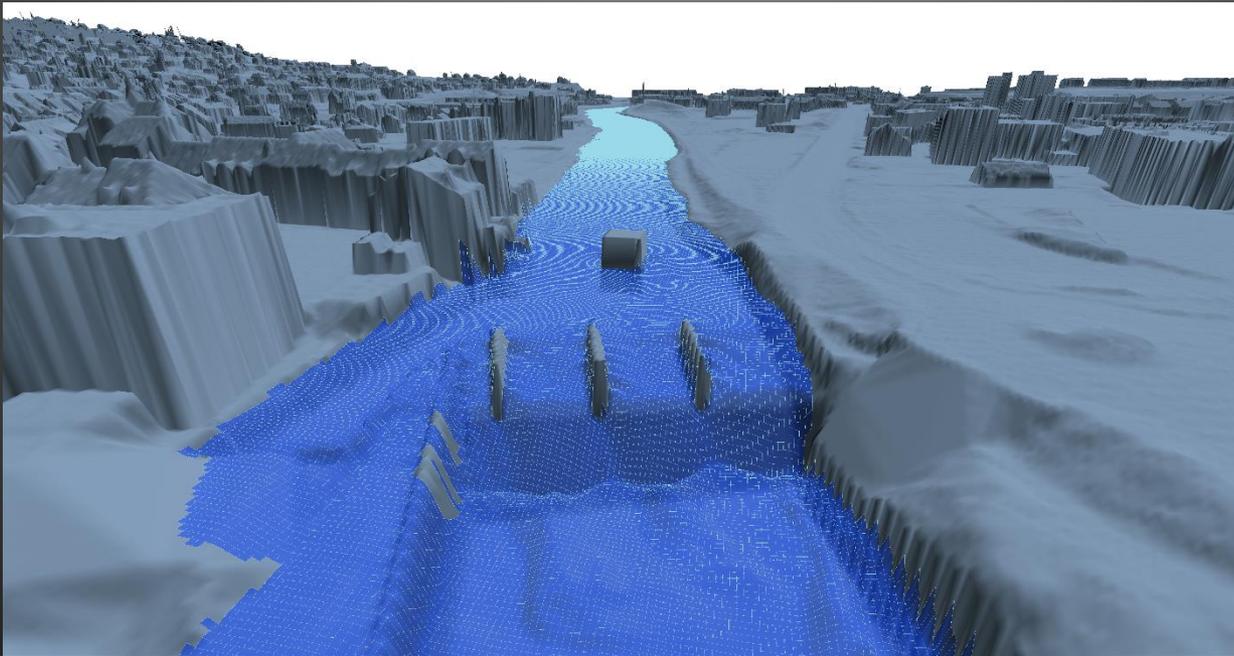


flox-GPU



Inhalt

- Einführung flox-GPU
- Pre-Processing (Gittererstellung)
- Modellaufbau, Modellstart
- Während der Modellierung (Laufzeit)
- Post-Processing
- Exkurs: Berechnung, Sonstiges
- Beispiele

flox-GPU

Einführung

Schwierigkeiten bei der numerischen Modellierung von Gewässern

Modellgenauigkeit:

- Genauigkeit des physikalischen Modells (Software)
- Genauigkeit des numerischen Modells (Software)
- Genauigkeit des Operats:
 - DTM/Rechennetze
 - Energieverluste (Strickler-Beiwerte)
 - Input-Ganglinien (Hydrologie), etc.

Schwierigkeiten bei der numerischen Modellierung von Gewässern

Interpretation, Verarbeitung und Verständnis:

- Weiterverarbeitung zu den Ingenieurprodukten?
- Wie plausibilisiert man die Ergebnisse?
- Was kann man kontrollieren, was muss man glauben?
- Wie überprüft man die Resultate?
- Wie findet man Fehler?

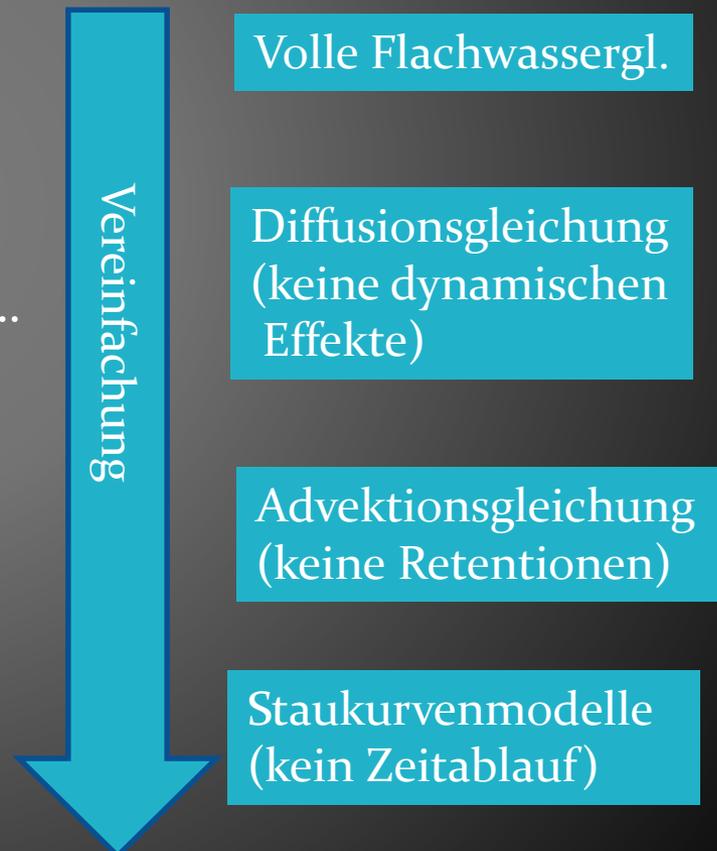
Situation war lange eine Minimierung von Kompromissen

Entweder:

- Anspruchsvolle, genaue Hydraulik
- Langsam
- Nur wenig Modellläufe möglich
- Aufwändiges Pre- und Post-Processing
- Software: BASEMENT, Flumen, HydroAS, ...

Oder:

- Vereinfachte, ungenaue Hydraulik
- Schnell
- Viele Szenarien
- Einfaches Pre- und Post-Processing
- Software: TriPad, FloodArea, ...



Was ist flox-GPU?

- Auf Geschwindigkeit optimierte 2D Modellierungssoftware (Verwendet NVIDIA-Grafikkarten)
 - Basiert auf Rasterdaten (Vereinfacht Pre- und Postprocessing)
- Kompromisse minimieren: möglichst schnell, genau und einfach

Kennzeichen flox-GPU

- Lösung der tiefengemittelten Flachwassergleichungen (FWG) (ohne Vereinfachungen)
- Moderne Numerik: Finite-Volumen, Zeit-explizit, shock-capturing, second-order, central-upwind (basiert auf Brodtkorb et al. 2011)
- Verwendet Raster als Rechennetz
- Rechnet auf GPU(s) → NVIDIA-Grafikkarte ist notwendig!
- Eigenentwicklung Ingenieure Bart AG (B. Loretz, C. Volz)
- Seit 2015 im Praxis-Einsatz
(v.a. Gefahrenkartierung, Wasserbau, Objektschutz)

Flachwassergleichungen

$$\begin{aligned}\frac{\partial}{\partial t} h + \frac{\partial}{\partial x} (uh) + \frac{\partial}{\partial y} (vh) &= 0, \\ \frac{\partial}{\partial t} (uh) + \frac{\partial}{\partial x} \left(u^2 + \frac{1}{2}gh^2\right) + \frac{\partial}{\partial y} (uvh) &= 0, \\ \frac{\partial}{\partial t} (vh) + \frac{\partial}{\partial y} (uvh) + \frac{\partial}{\partial y} \left(v^2h + \frac{1}{2}gh^2\right) &= 0,\end{aligned}$$

3 Gleichungen für 3 Erhaltungsgrößen:

- Massenerhaltung: Wassertiefe h
- Impulserhaltung in x- und y-Richtung: Intensitäten hu , hv
(u = Geschwindigkeit x-Richtung
 v = Geschwindigkeit y-Richtung)

Empirische Schliessbedingung:

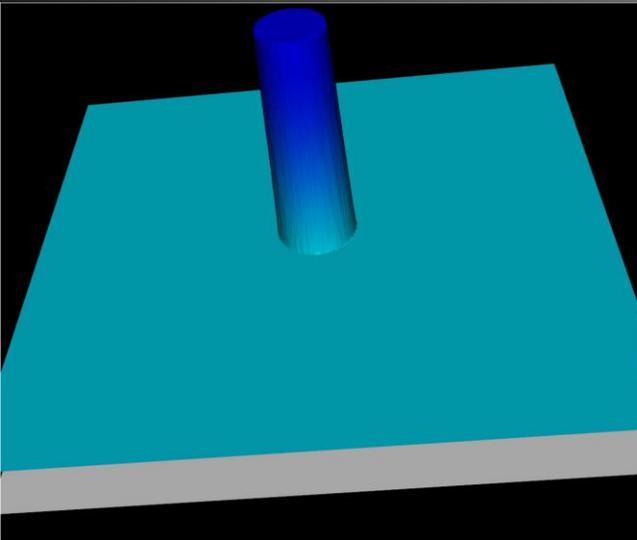
- Energieverluste (Strickler-Beiwert)

Features und Module

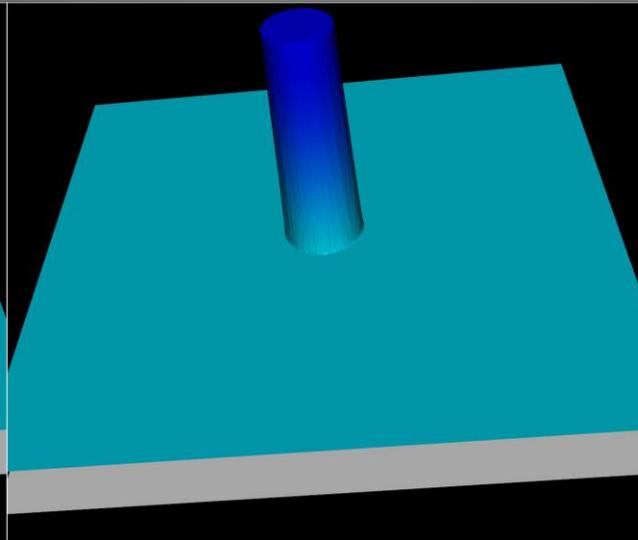
	TriPad	Flumen	BASEMENT	flox-GPU
Durchlässe	+	+	vereinfacht	+
Brücken	+	+	-	-
Wehre	-	+	+	-
Randbedingungen	vereinfacht	+	+	vereinfacht
Anfangsbedingungen/ Hotstart	-	+	+	+
Detailanalyse	-	-	+	+
Messstellen	-	+	+	+
Geschiebe	-	+	+	-

Validiert an Standard-Tests 2D Dammbrechwelle

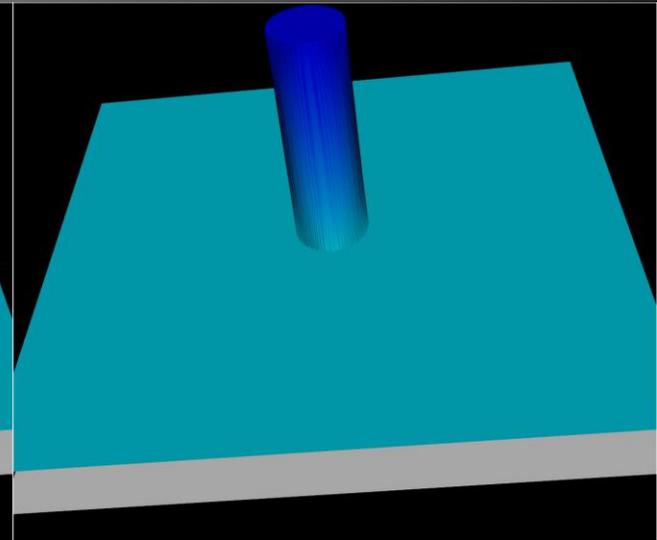
BASEMENT

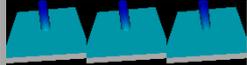


Flumen



flox-GPU



	TriPad	Flumen	Basement	flox-GPU
 10 s	-	340 s (x 0.029)	637 s (x 0.016)	4.2 s (x 2.4)

flox-GPU: Anwendungsbereiche

- 2D Gerinnehydraulik (stationär + instationär)
- Retentionen (statisch + dynamisch), Überflutungen
- Wellenausbreitung, Dammbbruch
- Schiessende und strömende Fliesszustände, inkl. Fliesswechsel
- Grossskalige, lange Simulationen
- Hoch aufgelöste Simulationen
- Abflussbildung (Beregnung)

Bsp.: Dammbbruchwelle



Anwendungsgrenzen

- Fließzustände mit 3D-Effekten:
 - Wasserwalzen, Tosbecken, etc.
 - 3D Sekundärströmungen (z.B. Kurvenströmung)
 - Dichte-Strömungen (z.B. See-Modellierung)
 - Nahfeld-Untersuchungen (z.B. 3D Anströmung Turbine)
- Sehr steile Gerinne führen zu Abweichungen (z.B. Schussrinnen)
- Reibungsverluste nur über Strickler-Beiwert
 - Gültigkeit von Strickler (turbulente Strömung, Überdeckung,...)
 - Keine Turbulenzmodellierung (z.B. k-epsilon Modell)
 - Keine Murgangs-Mechanik (Annäherung über tiefe Strickler-Beiwerte)

Warum GPUs?

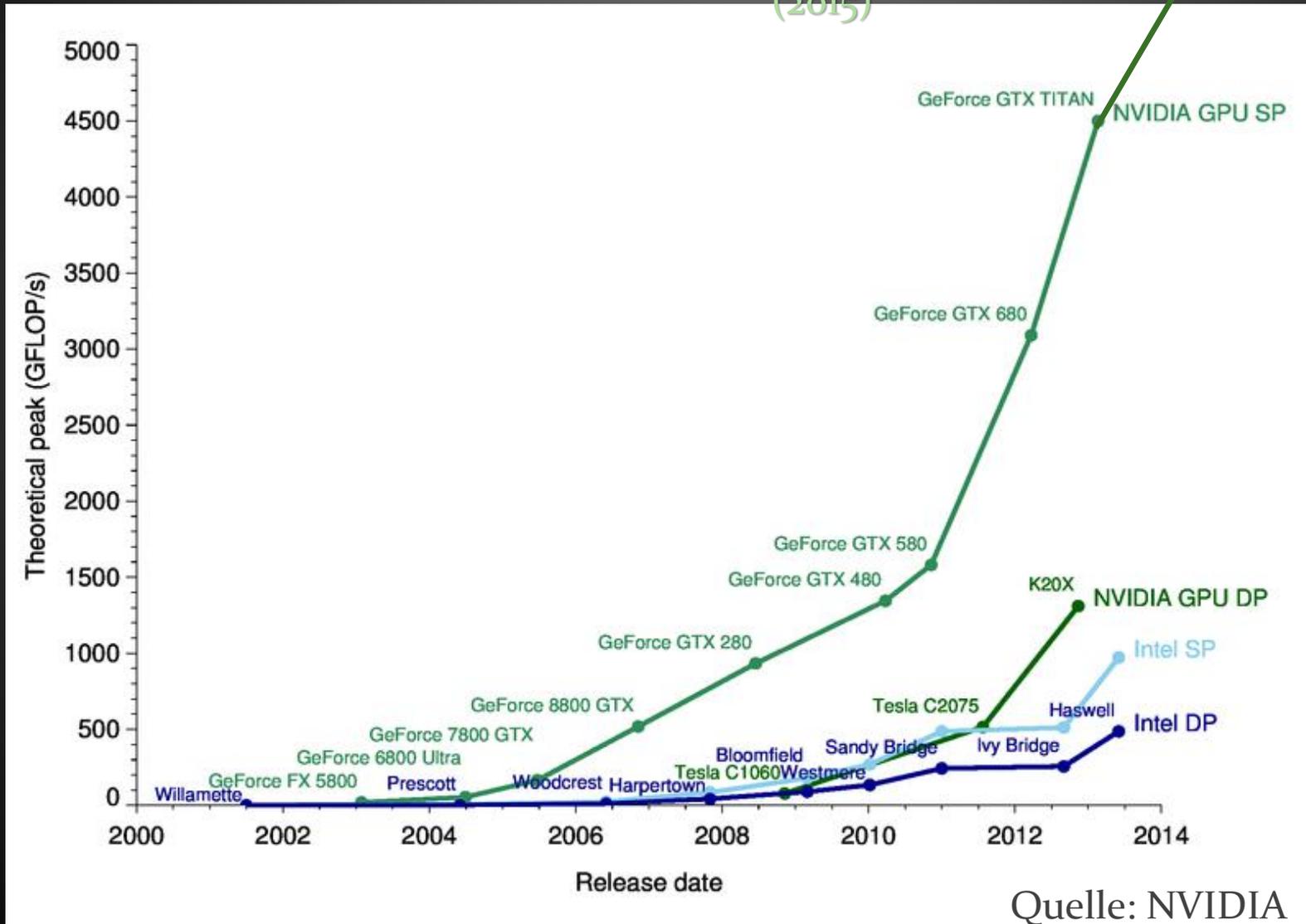


- Geeignet, viele gleichartige Operationen parallel auszuführen
- Lösung der FWG lässt sich effizient parallelisieren
→ Insbesondere mit strukturierten Gitter (Raster)
- Verhältnis Leistung/Kosten hervorragend
→ (Single-Precision, «Gamer-Karten»)

NVIDIA Titan Xp 12.1 TFLOP/s (2017)

Leistung spricht für sich

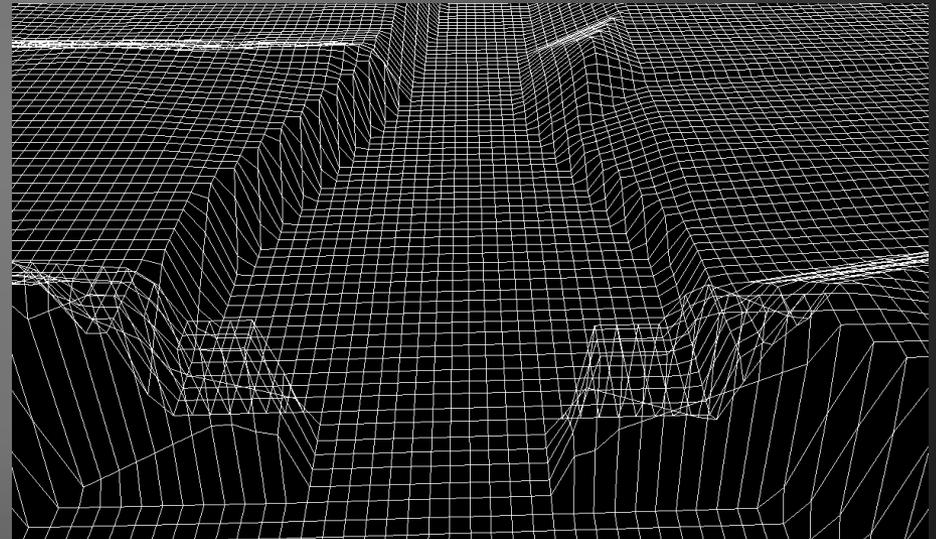
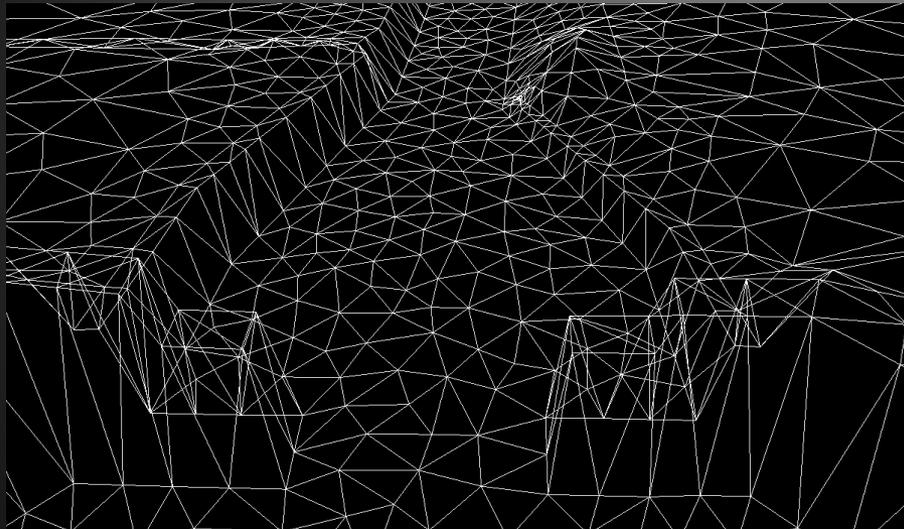
GeForce GTX Titan X @ 6144 GFLOP/s
(2015)



Quelle: NVIDIA

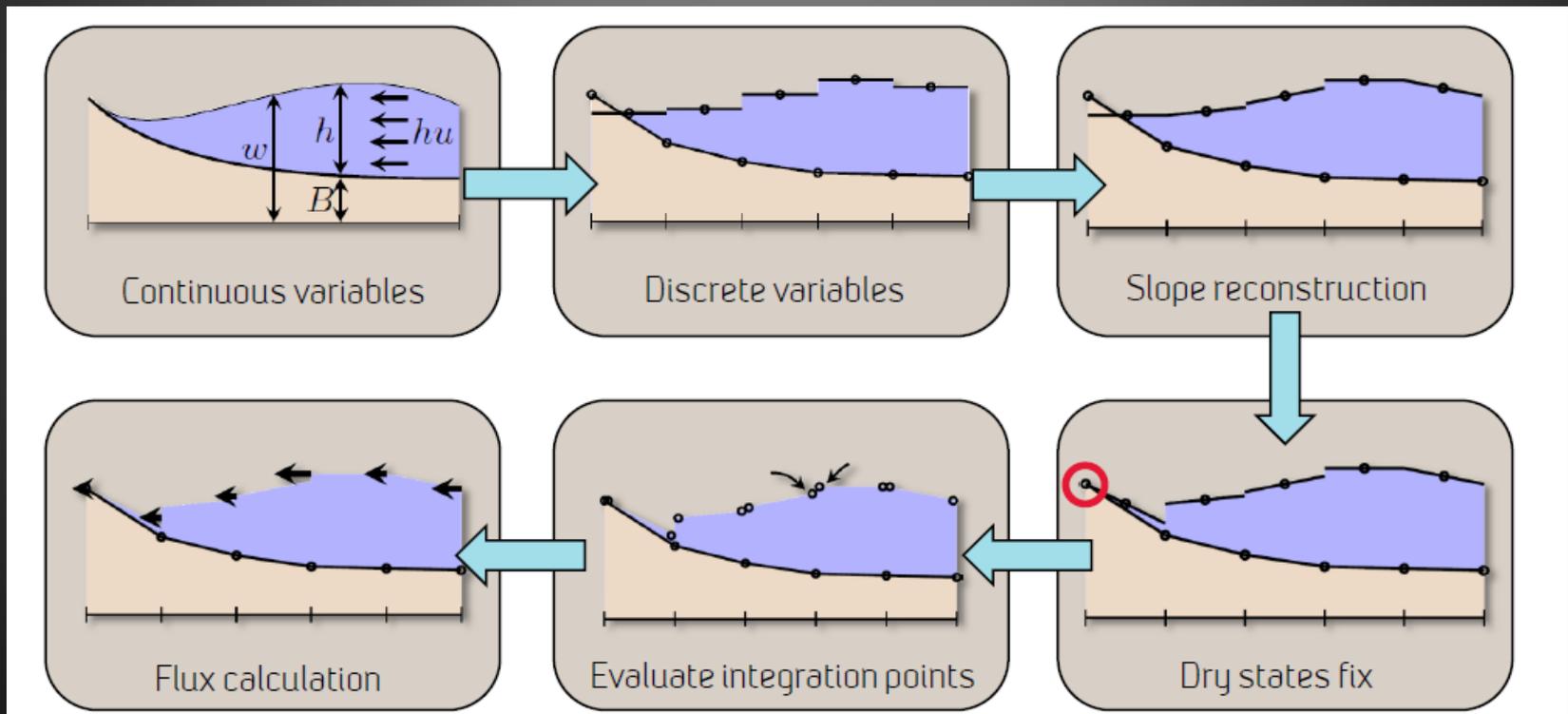
Rechennetze/Raster

- Strukturierte Gitter oder unstrukturierte Gitter?



Kleine Theorie flox-GPU

- Berechnungsablauf

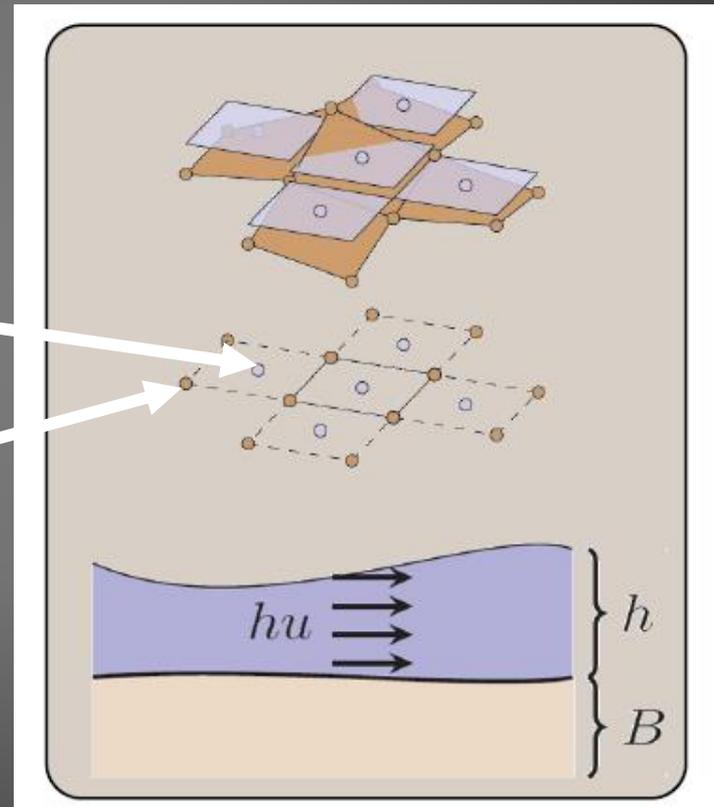


Versetzte Gitter

- Geländehöhen Zell-Ecken
- Variablen in Zell-Mittelpunkten (Wasserstand, etc.)

Variablen
(Cell-Center)

Geländehöhen
(Cell-Vertex)



→ DTM-Raster um 1 Zelle grösser
in Breite und Höhe als Variablen-Raster

Lösungsverfahren

- Zeit-explizite Simulation:
 - Zustand Q_{t+dt} basiert auf Zustand Q_t → Auflösen nach Unbekannten
 - Einfache Numerik, schnell, aber kleine Zeitschritte (CFL-Kriterium)!
- Numerisches Verfahren
 - 1. Ordnung Genauigkeit: EULER
(schnell, tendiert zu Oszillationen)
 - 2. Ordnung Genauigkeit: RUNGE-KUTTA
(2-stufiges Verfahren, langsamer, stabiler)

CFL Kriterium

- Zeitschritt unterliegt Stabilitätskriterium (CFL)

$$CFL = \frac{\text{Ausbreitungsgeschwindigkeit}}{\text{Netzgeschwindigkeit}} = \frac{v+c}{\Delta x / \Delta t} < 1.0 !$$

- CFL-Zahl (<1.0) wird vorgegeben → Zeitschritt
- Kleine Zellen führen zu kleinen Zeitschritten (=lange Simulationen)
- Hohe Fließgeschwindigkeit führt zu kleinen Zeitschritten

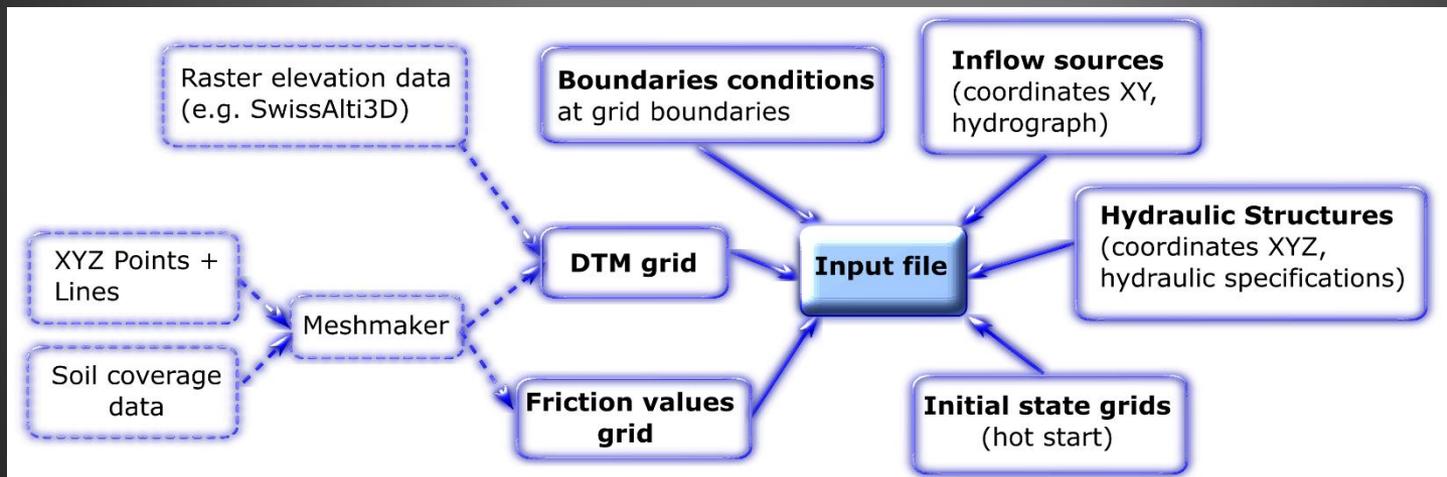
Kleines Fazit:

- flox-GPU basiert auf denselben Gleichungen wie Flumen/Basement
- Numerik ist robust und ausgereift
- «Lightweight» aber mit wichtigsten Funktionalitäten (Randbedingungen, Quellen, Durchlässe, Messstellen, etc.)
- Die Daten müssen i.A. hoch aufgelöst werden
- Trotz hoher Auflösung ist die Performance dank GPU hoch
- Verwendung von Raster
 - Einfachheit
 - Pre- und Postprocessing mit Standard GIS

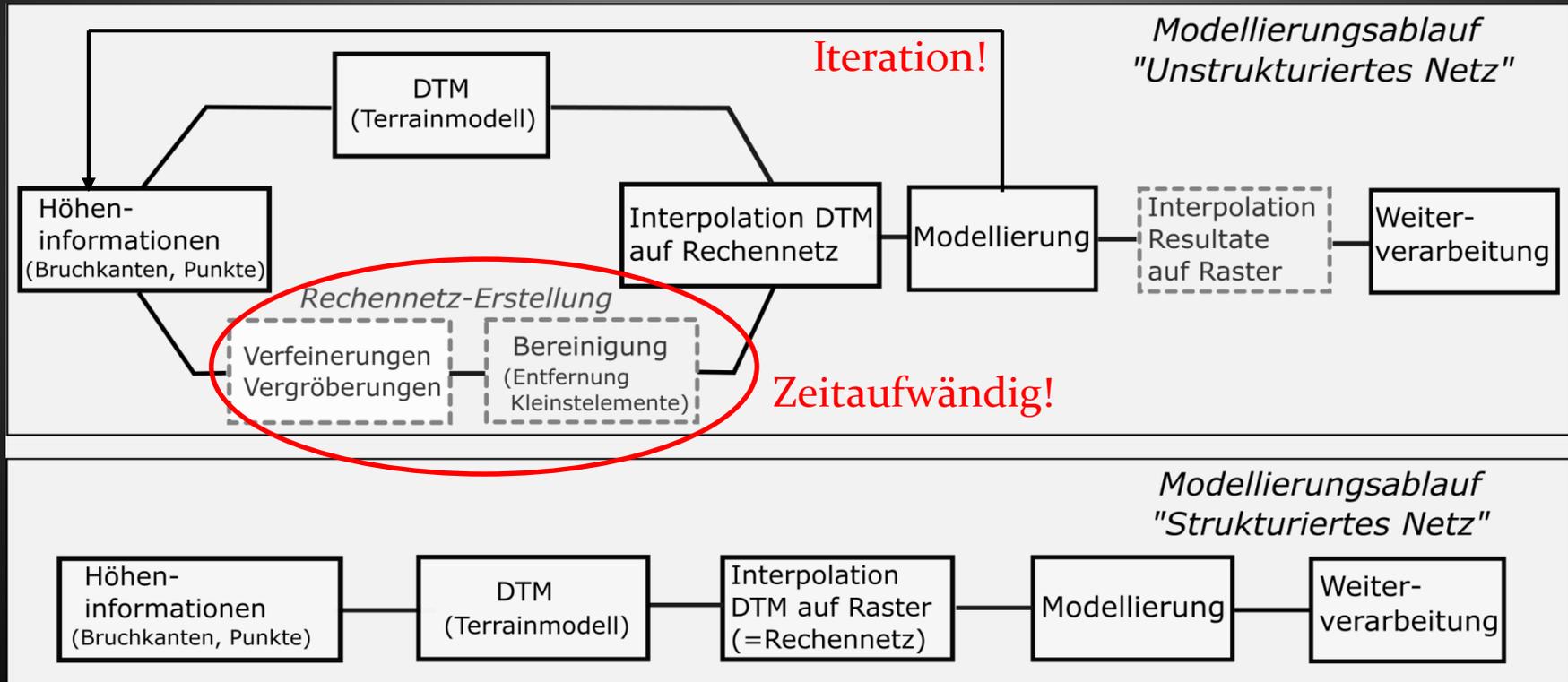
flox-GPU

Pre-Processing (Gittererstellung)

Pre-Processing: Übersicht



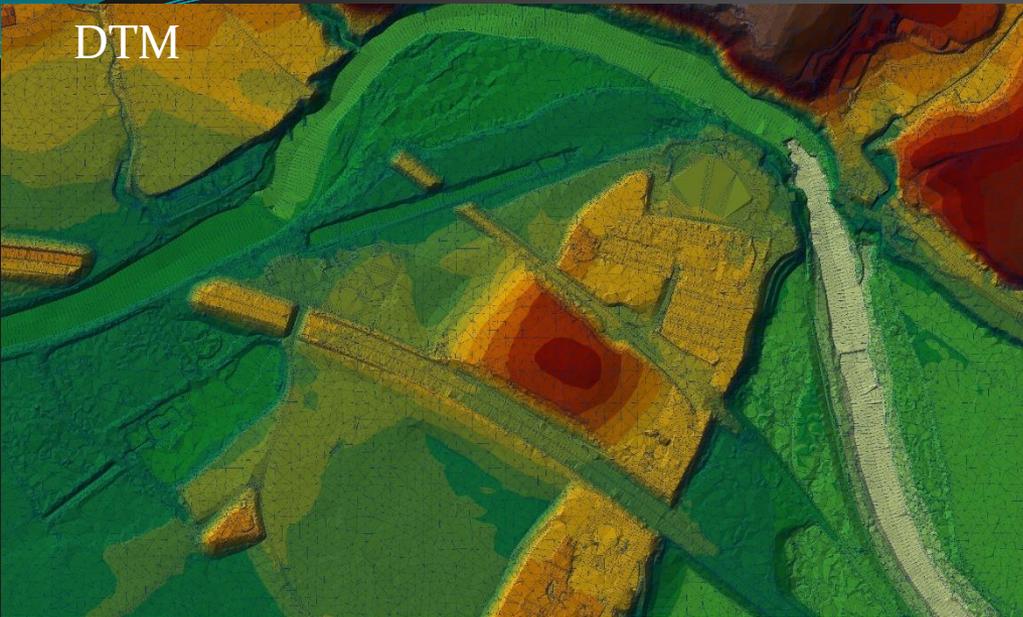
Gittererstellung: Unstrukturierte Netze vs. Strukturierte Netze



Exkurs: Unstrukturierte Netze (I)

- Motto: «So wenig wie möglich, so viel wie nötig»
- Netze sind Triangulationen von Punkt und Liniendaten
- 2-stufiges Vorgehen: DTM-Netz / Rechennetz
- Rechennetze:
 - Ausdünnung Bruchkanten (Nester), wegen CFL-Kriterium **(AUFWÄNDIG!)**
 - Qualitätskriterien
 - Informationsverlust

DTM

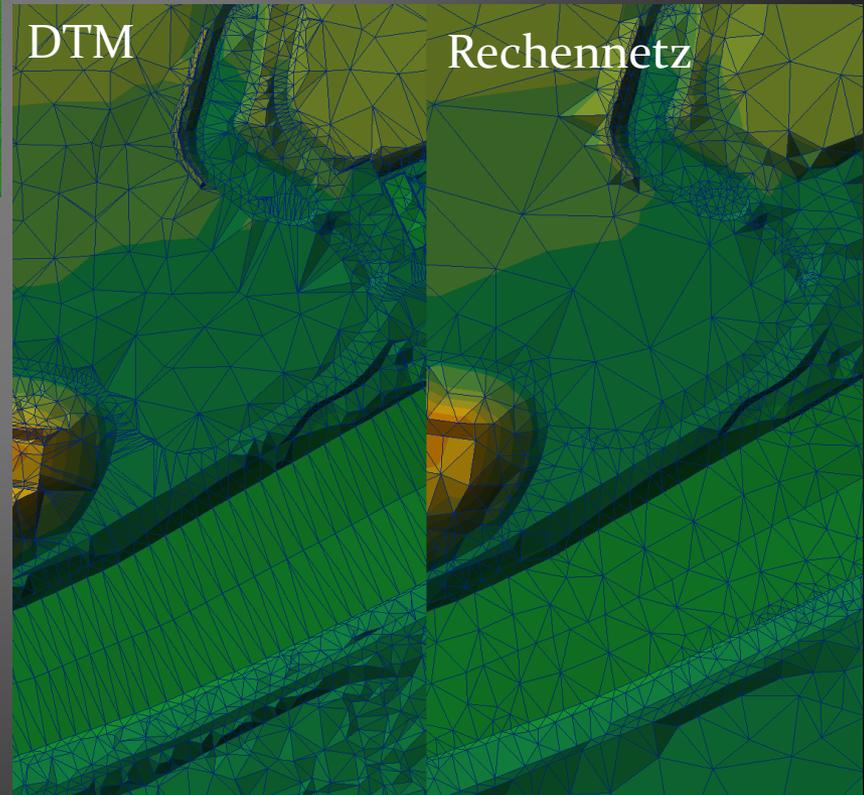


Unstrukturierte Netze (II) DTM und Rechennetz

Rechennetz



DTM

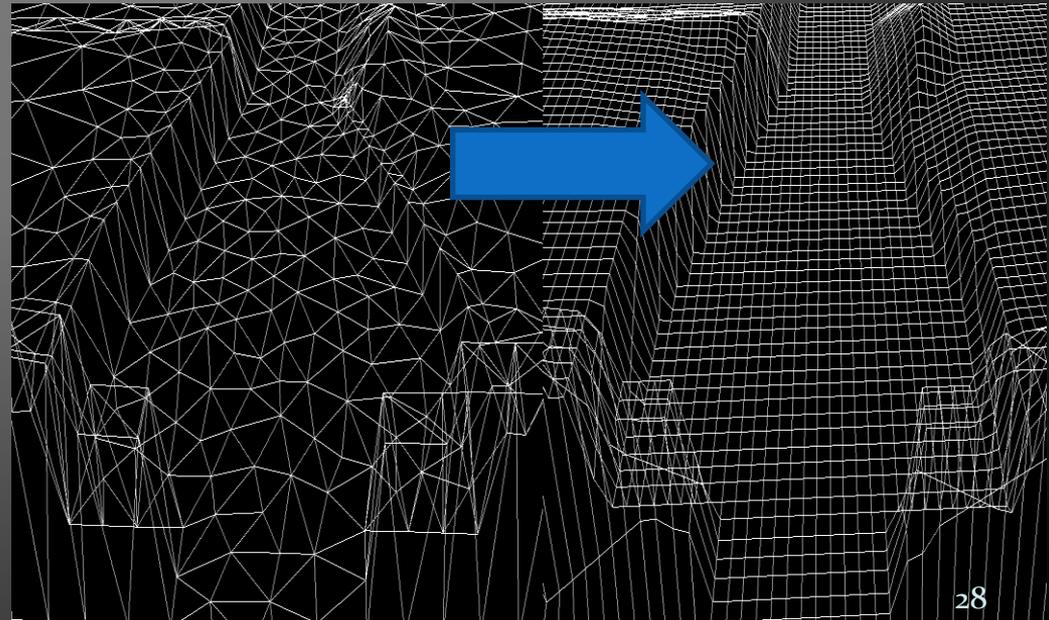
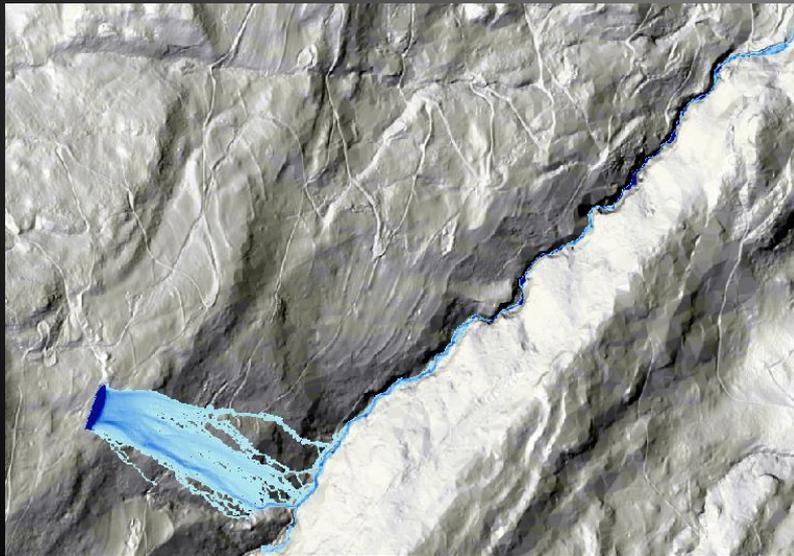


Rechennetz

Strukturierte Gitter (flox-GPU)

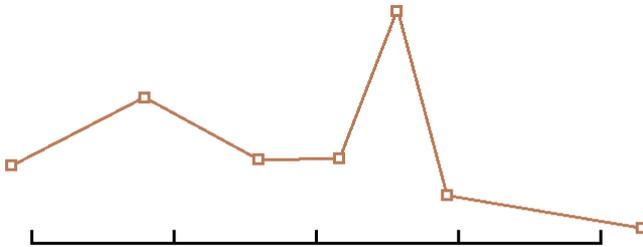
- Direkt auf Raster Modellieren (SwissAlti3D, Drohne, etc.)

- Erstellung DTM-Netz wie bisher (kein «Rechennetz» nötig, keine Qualitätskriterien!)
- DTM-Netz wird auf Raster interpoliert

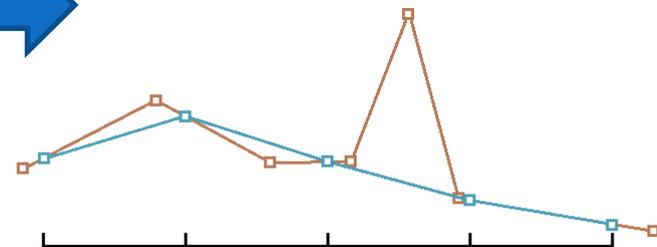


Richtige Zellgrösse? Beispiel: grobes Raster

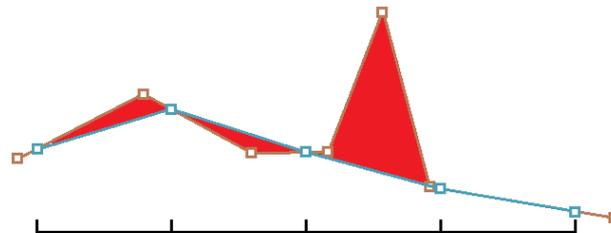
Geländehöhen Vektordaten



Geländehöhen Raster

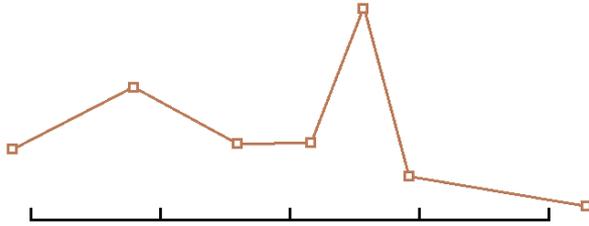


Genauigkeitsverluste
hoch

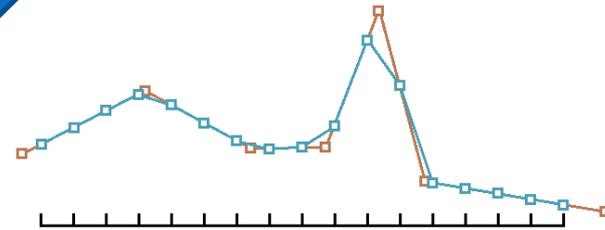


Richtige Zellgrösse? Beispiel: feineres Raster

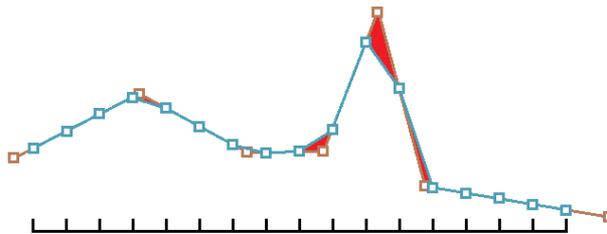
Geländehöhen Vektordaten



Geländehöhen Raster



Genauigkeitsverluste
gering



Hinweis:

- Viele Interpolationsmethoden für Raster
- Bruchkanten können erzwungen werden (z.B. «meshmaker»)

Mesherstellung für Strukturierte Gitter

- Kritischste Stelle im Netz definiert Zellgrösse für ganzes Grid
- Halbe Zellgrösse führt zu ca. 8-fachem Rechenaufwand (4-fache Anzahl Zellen, halb so grosser Zeitschritt)
- Schwierig: Systeme mit sehr kleinen und grossen Komponenten (kleine Bäche und grosse Domain)
- Manchmal sinnvoll schmale Strukturen (z.B. Mauer) künstlich zu verbreitern

Alles Raster oder was?

Flächige Eingangsdaten ebenfalls als Raster:

- Strickler-Beiwerte
- Anfangsbedingungen/Hotstart (h , h_u , h_v)
- Berechnungsdaten

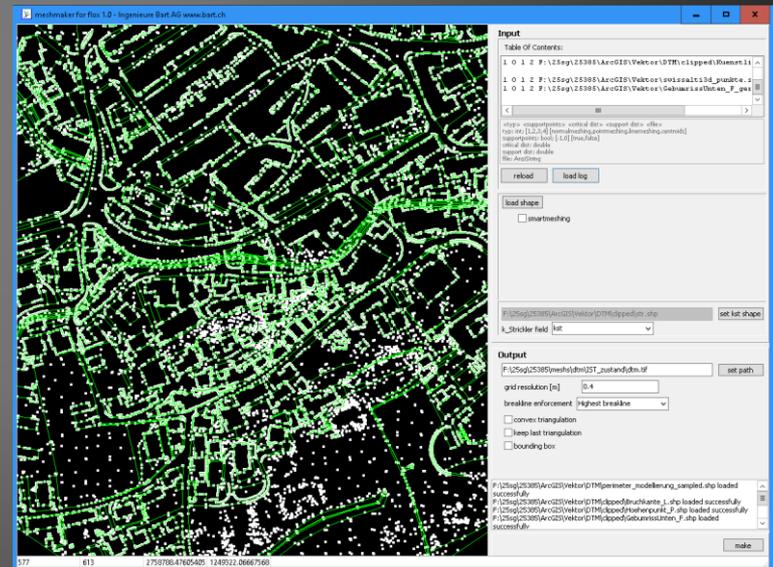
Ausgabedateien als Raster

Hilfsprogramm DTM-Erstellung aus Shape-Dateien:

«meshmaker» kostenfrei verfügbar

- Triangularisierung von Bruchkanten/Punkten (mit TRIANGLE)
- Mappen auf DTM-Raster
- Erstellt Raster mit Strickler-Beiwerten
- Schnell + Effizient
- Quickguide:

www.flox-gpu.ch/clients/quickguide_meshmaker.pdf



flox-GPU

Modellaufbau, Modellstart

Eingabedatei

- Steuerung über Text-Datei mit Schlüsselwörtern (Tags)
- «#» für Kommentare
- Reihenfolge Tags ist egal
- Gross-/Kleinschreibung ist egal
- Leerzeilen sind egal
- «include» für Include-Datei (Einbindung zusätzlicher Eingabedatei)
- Quickguide:

http://www.flox-gpu.ch/files/reference_web.txt

```
BATCH
project_id 25359
GQ 1179
scen_id 0
return_period 30
probability 1
run_name nm
cfl 0.999

progress_images true
output B

runtime 9

email volz@bart.ch

### NACH MASSNAHME ###

##mesh F:\25sg\25359\meshs\dtm\nm\dtm_nm.tif
##k_strickler_grid F:\25sg\25359\meshs\dtm\nm\dtm_nm_kst.tif

mesh F:\25sg\25359\meshs\dtm\nm\dtm_nm_grob.tif
k_strickler_grid F:\25sg\25359\meshs\dtm\nm\dtm_nm_grob_kst.tif

path c:\flox\25359\HQ30_saar_nm

##### RANDBEDINGUNGEN #####

N non_ref
E non_ref
S non_ref
W non_ref
```


Datenablage:

measures	31.07.2017 02:52	Dateiordner	
observers	31.07.2017 02:52	Dateiordner	
progress	31.07.2017 03:45	Dateiordner	
d_max.tif	31.07.2017 13:19	TIFF-Bild	4'373 KB
d_max.tif.aux.xml	31.07.2017 13:19	XML-Dokument	1 KB
hillshade.png	31.07.2017 13:19	PNG-Bild	24'428 KB
hillshade.png.aux.xml	31.07.2017 13:19	XML-Dokument	1 KB
i_max.tif	31.07.2017 13:19	TIFF-Bild	4'470 KB
i_max.tif.aux.xml	31.07.2017 13:19	XML-Dokument	1 KB
in.txt	31.07.2017 13:19	TXT-Datei	2 KB
progress_log.txt	31.07.2017 13:19	TXT-Datei	8 KB
summary.txt	31.07.2017 13:19	TXT-Datei	1 KB
system.png	31.07.2017 13:19	PNG-Bild	32'771 KB
system.png.aux.xml	31.07.2017 13:19	XML-Dokument	1 KB
time_log.txt	31.07.2017 13:19	TXT-Datei	410 KB
v_max.tif	31.07.2017 13:19	TIFF-Bild	4'541 KB
v_max.tif.aux.xml	31.07.2017 13:19	XML-Dokument	1 KB
w_max.tif	31.07.2017 13:19	TIFF-Bild	3'068 KB
w_max.tif.aux.xml	31.07.2017 13:19	XML-Dokument	1 KB

- «path» Tag in Eingabedatei legt Ort der Datenablage fest
- **Hauptverzeichnis:** allgemeine Informationen, Resultate, Kontrollbilder und Logs
- «measures»: Messungen als Text-Dateien (z.B. Durchflussmessung)
- «observers»: Detailbilder für Auswertungen (z.B. Video, Kontrolle)
- «progress»: Bilder zur Fortschrittsanzeige, CFL-Bedingung

Meta-Informationen zur Simulation

- Optionale Daten
- Hilfreich für automatische Weiterverarbeitung
- Verwaltung von grosser Anzahl an Szenarien / Modellläufen
- Derzeit Fokus auf Gefahrenkartierungen

```
# META INFORMATIONEN  
  
project_id 25359  
GQ 1179  
scen_id 0  
return_period 300  
probability 1  
run_name kleinbach
```

Allgemeine Parameter

- Auswahl GPU
- Festlegung CFL-Zahl (<1.0)
- Simulationszeit («runtime» [h], «runtime_s» [s])
- Lösungsverfahren
(«euler true» = EULER, «euler false» = RUNGE_KUTTA (default))
- «Advanced» Parameter → siehe Referenz

```
# PARAMETER

gpu 0
cfl 0.9
runtime 5.5

mesh F:\mesh\dtm.tif

path C:\flox\results
|
```

```
# Friction  
  
k_strickler 33  
k_strickler_grid F:\25sg\25359\meshs\dtm\kst.tif  
  
|
```

Reibungsverluste

- Reibungsverluste definiert über Strickler-Beiwert [$\text{m}^{1/3}/\text{s}$]
- Default-Wert: «k_strickler»
 - gilt auf ganzer Domain (sofern nicht überschrieben)
- Detailangabe: «k_strickler_grid»
 - Zellgenaue Angabe
 - Berücksichtigung lokaler 3D Energieverluste möglich (z.B. bei Absturz, Rampen, Brücken, ...)
 - Nodata-Werte = Default-Wert
- Erhöhung Rauigkeit bei «steilen Zellen»
 - Berücksichtigung der tatsächlichen Fläche der Sohlreibung

Quellen

- Tag «source»
- Angabe als Punkt-Koordinate
- Angabe der Zeitreihe: Zeit , Quelle [m³/s]
(Zeiteinheit [s] oder [h], je nach «runtime» oder «runtime_s»)
- Zugabe erfolgt ohne Impuls
 - Strömung muss erst beschleunigen
 - Möglichst ausserhalb des relevanten Perimeters setzen
 - Hohe Wasserstände möglich → kleine Zeitschritte
 - Zugabe auf mehrere Zellen verteilen mit «source_radius»

```
# Zugabe Quelle
source 2754610.18 1228467.35
0.000 0.000000
114.770 31.150000
459.080 0.0000000
end_source

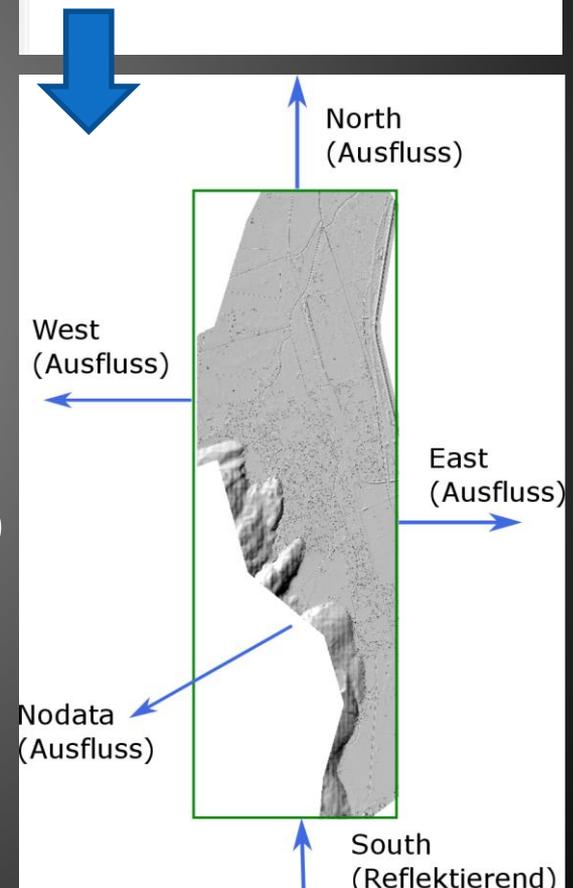
# Zugaben auf 3x3 Zellen verteilen
source_radius 1
|
```

Randbedingungen (RB)

- Sehr einfache Behandlung, dafür effizient!
- 2 Typen:
 - Reflektierender Rand = Ausfluss nicht möglich
 - Nicht-reflektierender Rand = Wasser fließt aus
- RB nur am Domainrand (East, West, North, South)
- Beeinflussung vermeiden
→ weit genug entfernt von interessierendem Perimeter
- Besonderheit:
Nodata-Zellen im Raster als Ausflussrand

```
# RANDBEDINGUNGEN
```

```
N non_ref  
E non_ref  
S ref  
W non_ref
```



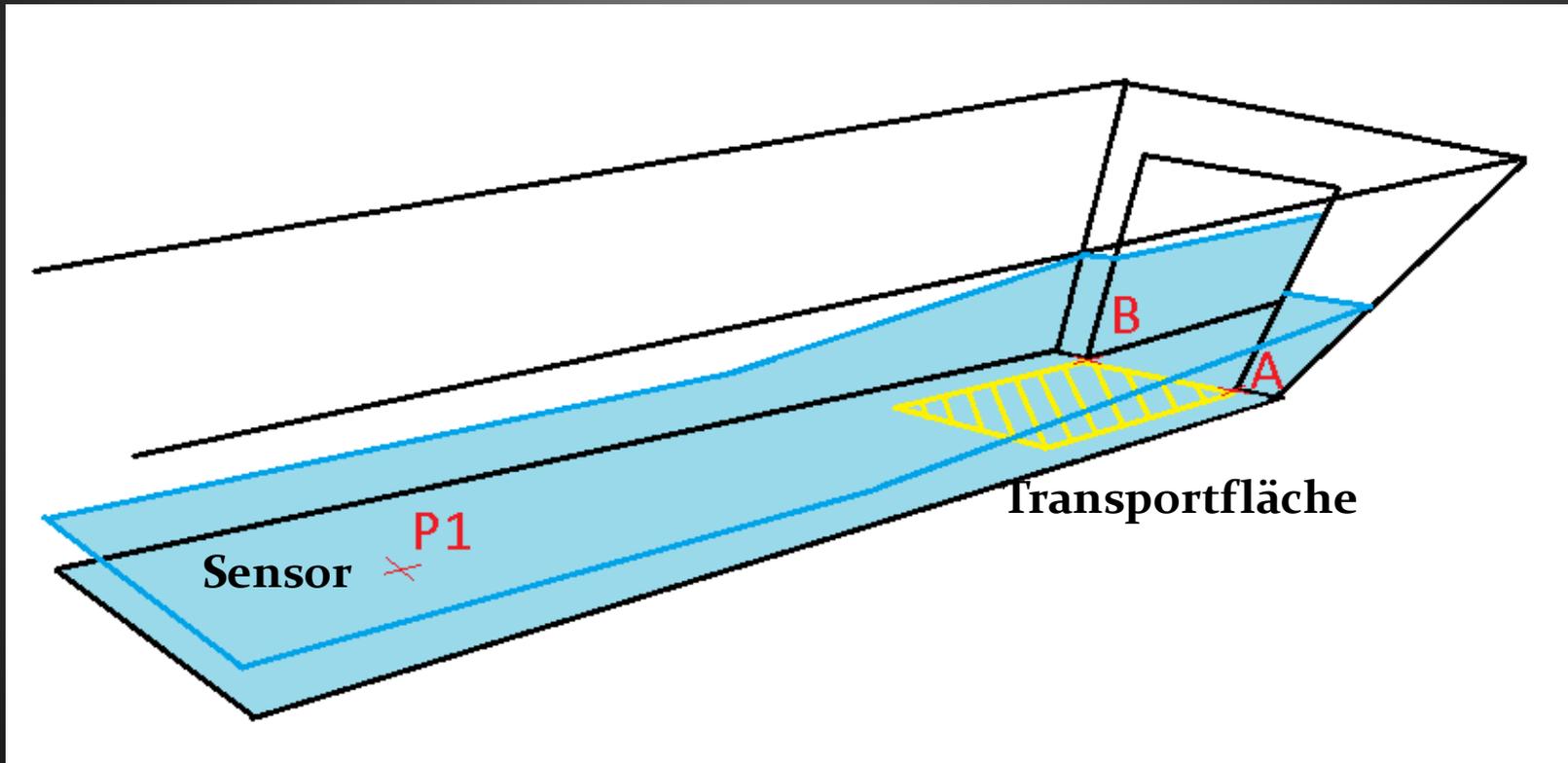
Anfangsbedingungen (AB)

- Wasserstände (und Intensitäten) können für Modellstart als Raster definiert werden (z.B. Reservoir-Wasserstand für Dambruchmodell)
- Achtung: Es müssen sinnvolle Werte vorgegeben werden, kein Check!
- Siehe «Hotstart» in Abschnitt Post-Processing

Durchlässe

- Durchlässe in flox-GPU sind eine 1D\2D Kopplung
- Als solche sind sie fehleranfällig und können Oszillationen verursachen
- 1D Rechnungen nach dem Hager Modell führen zu einer Verschiebung von Wassersäulen und einer Anpassung von Fließgeschwindigkeiten (2D)
- Durchlässe funktionieren über 2 «Sensoren» und 2 Transportflächen
- Numerisch: berechnete Durchflüsse werden über Zeitschritte gemittelt um Oszillationen zu verringern

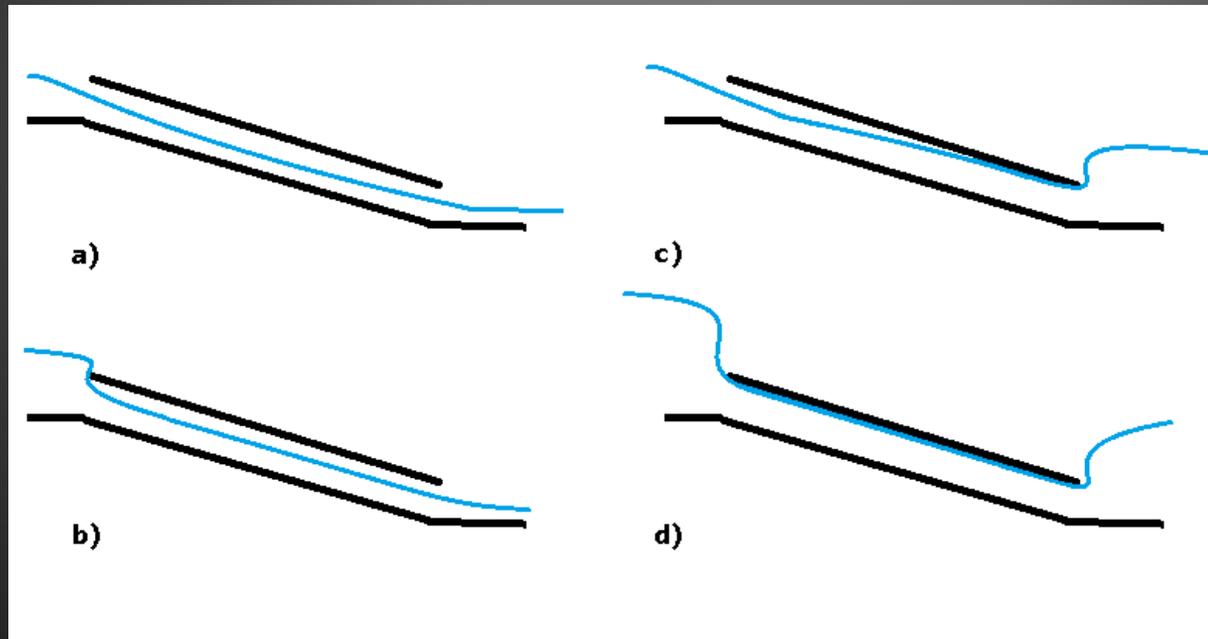
Durchlassmodell basierend auf Hager (1998)



```
culvert p1.x p1.y A1.x A1.y B1.x B1.y p2.x p2.y A2.x A2.y B2.x B2.y width height kst streamline_factor inlet_loss method name
```

Durchlassmodell basierend auf Hager (1998)

- Es werden 4 grundlegende Fälle unterschieden:



	Auslass nicht eingestaut	Auslass eingestaut
Einlass nicht eingestaut	a)	c)
Einlass eingestaut	b)	d)

Durchlassmodell basierend auf Hager (1998)

Grundlegende Formeln, Hager/Beffa

	Rectangular culvert	Pipe
$Q_{critical}$	$\left(\frac{2}{3}\right)^3 \times \sqrt{g \times H_E^3} \times w$	$\left(\frac{3H_E}{5D}\right)^5 \times \sqrt{g \times D^5}$
$Q_{uniform}$	$A \times \sqrt{\frac{\Delta Z_E}{c_0}}$	$A \times \sqrt{\frac{\Delta Z_E}{c_0}}$
Q_{gated}	$w \times h \times c_g \times \sqrt{2g \times (H_E - c_g \times h)}$	$\frac{\pi}{4} D^2 \times c_g \times \sqrt{2g \times (H_E - c_g \times D)}$
$Q_{pressurized}$	$A \times \sqrt{\frac{\Delta Z_E}{c_1}}$	$A \times \sqrt{\frac{\Delta Z_E}{c_1}}$

Durchlassmodell basierend auf Hager (1998)

	$ratio_{inlet} < 1.0$	$1.0 < ratio_{inlet} < 1.2$	$1.2 < ratio_{inlet}$
$ratio_{outlet} < 1.0$	o	1	4
$1.0 < ratio_{outlet} < 1.2$		2	5
$1.2 < ratio_{outlet}$		3	6

Case	Handling
o	$\min(Q_{critical}, Q_{uniform})$
1	intermediate(0,4)
2	intermediate(0,1,3,5)
3	intermediate(0,6)
4	$\min(Q_{gated}, Q_{uniform})$
5	intermediate(4,6)
6	$Q_{pressurized}$

Brücken

- Derzeit noch kein eigenes Brückenmodul
- Empfohlen:
 - Einbau grosser Elemente ins Raster (z.B. Querschnittseinengung)
 - Nicht abgebildete Elemente als lokale Rauigkeitserhöhung (Energieverluste durch Pfeiler, Widerlager, Brückenplatte, ...)



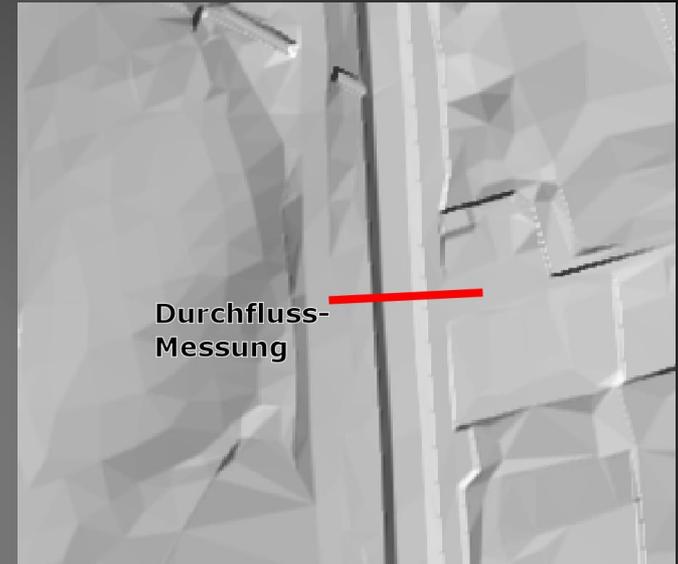
Optionale Ausgaben

- «measure» = Wieviel Durchfluss durch Linie?

```
# Durchfluss Messung
measure m.txt
342.3 532.2
430.01 536.4
end_measure
|
```

Ausgabedatei

Koordinaten
Anfangs- und
Endpunkt



- «observer» = Zeitlicher Verlauf Fließgrößen = Raster (t)

```
# Detailauswertung
observer <xll> <yll> <width> <height> <values> <fps> <name> <start=0> <end=end_time>
```

values = [hwuvtfE] (h w u v Tau Froude Energylevel)

fps = «frames per second» (Anzahl Ausgaben = fps x Dauer Ausgabe in [s])

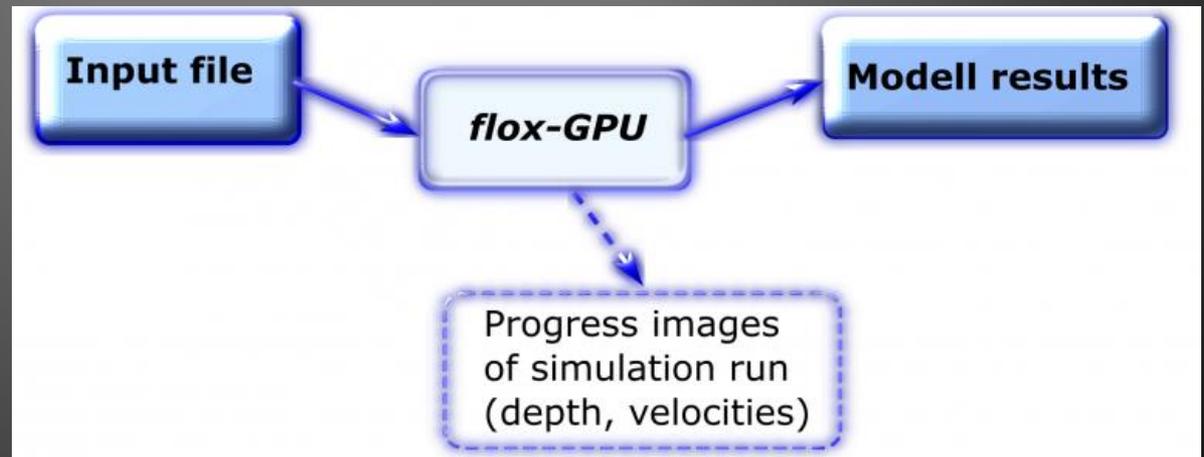
- Hotstart-Raster = «output huv»

flox-GPU

Während der Modellierung
(Laufzeit)

Während der Modellierung

- Speicher
- Kontrollbilder, Fortschrittsanzeige
- Allgemeine Logs
- Measures
- Durchlässe

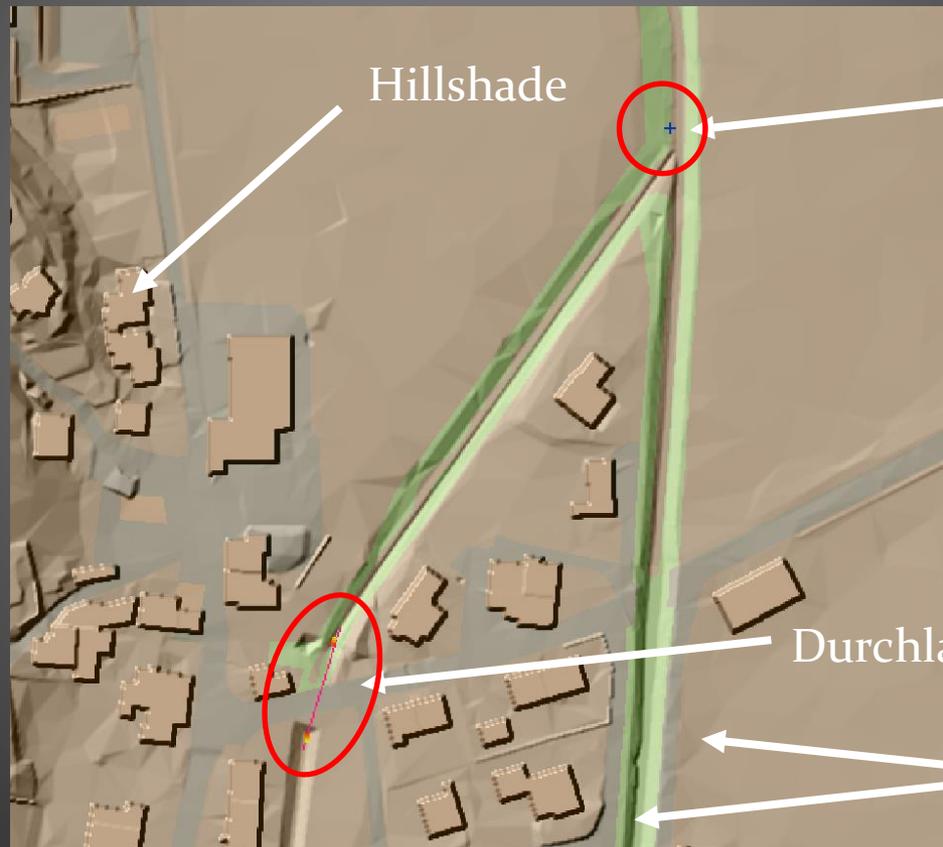


Speicher

- Duales Speichermodell: CPU/GPU
- flox-GPU alloziert Speicher für die Modellierung dynamisch
- Das Rechnernetz muss immer komplett geladen werden
- Zustandsvariablen der Modellierung werden nach Bedarf reserviert
- Logs/externe Programme geben Aufschluss über Ressourcenverwendung

Kontrolle: Alles richtig aufgesetzt?

«system.png»



Hillshade

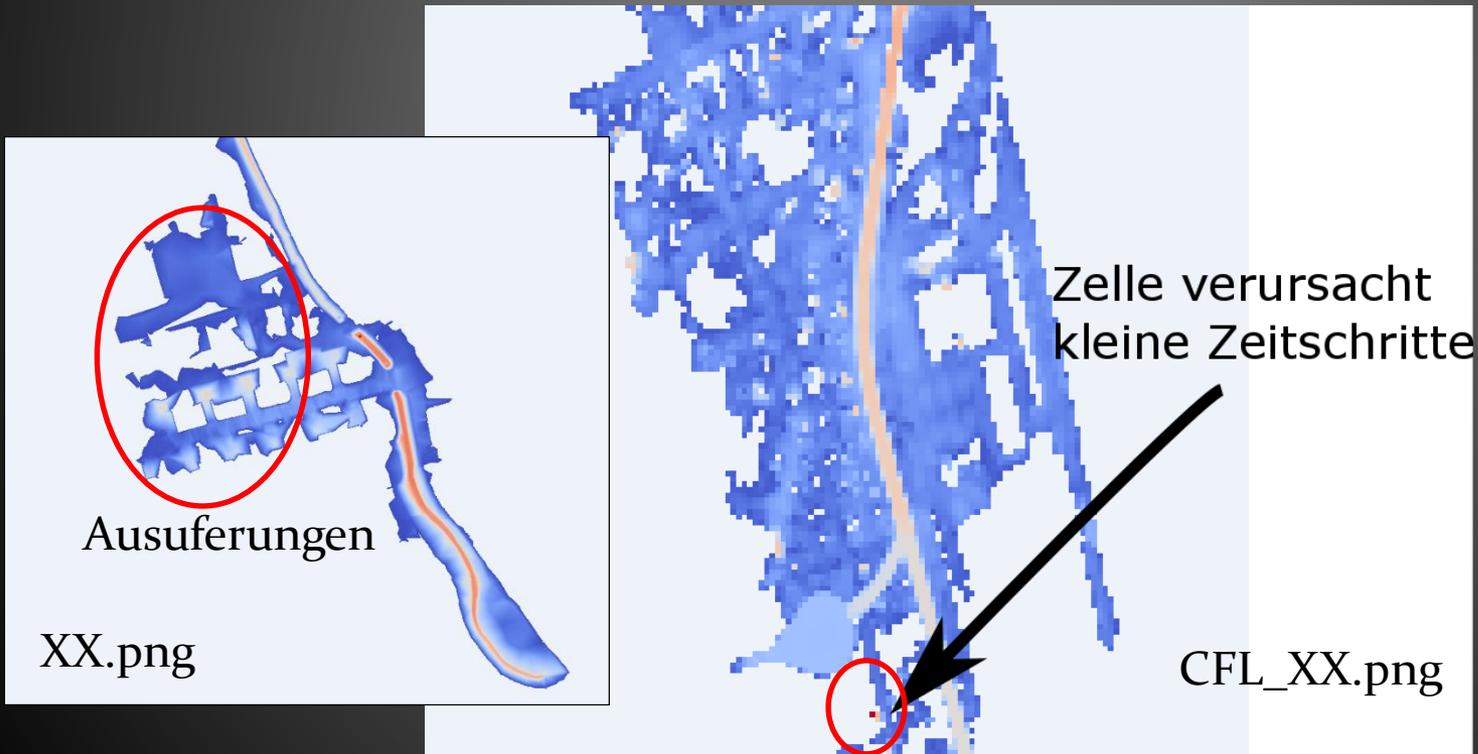
Quelle

Durchlass

Farben =
Strickler-Beiwerte

Kontrolle: Ausuferungen und CFL-Bedingung «CFL_XX.png», «XX.png»

- Update mit jedem % Rechenfortschritt in progress-Ordner



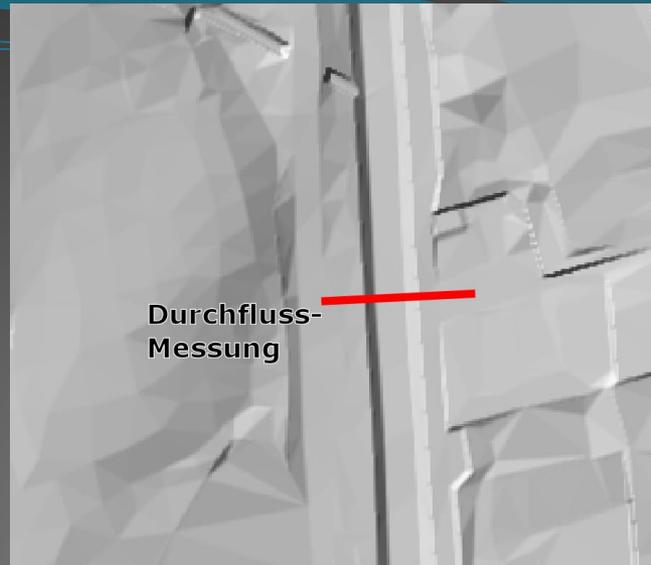
Kontrolle Simulationsparameter: «progress_log.txt»

- Update mit jedem % Rechenfortschritt
- Misst Fortschritt, Performance und Systemnutzung

Fortschritt [%] Echte Laufzeit [s] Berechnungs-Schritt [-] Modellzeit [s] Zeitschrittgröße [s]

prog	date	time	runtime[s]	step	model_time	time_ratio	timestep[s]	used_blocks	max_blocks
0	19.09.2017	16:46:13	0	0	0	0	0	0	10000
1	19.09.2017	16:46:15	1.6	2340	72.007613	43.667443	0.021986	111	10000
2	19.09.2017	16:46:18	5	6610	144.000406	28.611247	0.01494	217	10000
3	19.09.2017	16:46:22	9.4	11757	216.003	23.010866	0.013445	299	10000
4	19.09.2017	16:46:28	15	17287	288.002007	19.225768	0.012708	437	10000
5	19.09.2017	16:46:35	21.6	23310	360.003962	16.649892	0.011127	508	10000
6	19.09.2017	16:46:43	29.9	30215	432.000694	14.429845	0.009758	628	10000
7	19.09.2017	16:46:53	39.8	37776	504.002342	12.650661	0.009409	789	10000
8	19.09.2017	16:47:04	51.2	45539	576.000179	11.250003	0.009128	995	10000
9	19.09.2017	16:47:17	64.4	53538	648.008114	10.055836	0.008894	1158	10000
10	19.09.2017	16:47:32	79.3	61710	720.004178	9.075607	0.008735	1332	10000
11	19.09.2017	16:47:48	95.4	69977	792.007114	8.303963	0.008715	1508	10000
12	19.09.2017	16:48:06	112.7	78141	864.004464	7.667023	0.008904	1709	10000
13	19.09.2017	16:48:24	131.3	86158	936.004824	7.127772	0.009102	1938	10000
14	19.09.2017	16:48:44	151.2	94077	1008.00194	6.668532	0.009057	2136	10000
15	19.09.2017	16:49:05	172	102081	1080.00322	6.278614	0.008939	2216	10000
16	19.09.2017	16:49:27	193.8	110125	1152.00759	5.94287	0.008976	2322	10000
17	19.09.2017	16:49:49	216.6	118145	1224.00568	5.651805	0.008991	2446	10000
18	19.09.2017	16:50:13	240.1	126136	1296.0031	5.387885	0.008938	2548	10000

Measure Log



- Gemessen wird der (Betrag) des Durchflusses durch jede Zellkante
- Logdatei:
 - Abgelegt im Verzeichnis «measures»
 - 1. Spalte Zeit [t]
 - 2. Spalte Durchfluss [m³/s]

Zeit [s]	Discharge [m ³ /s]
602.077026	25.102558
602.144348	25.111506
602.211731	25.117708
602.279053	25.127325
602.346375	25.139839
602.413757	25.151834
602.481079	25.159863
602.548401	25.16523
602.615784	25.171732
602.683105	25.17918
602.750488	25.184649
602.81781	25.185274
602.885193	25.187275
602.952515	25.189474
603.019897	25.19043
603.08728	25.185343
603.154602	25.177147
603.221985	25.171064
603.289368	25.163061
603.35675	25.151247
603.424072	25.134405
603.491455	25.121
603.558838	25.113903
603.626221	25.111485
603.693604	25.111612
603.760986	25.114098
603.828369	25.123873
603.895752	25.139006
603.963135	25.157047
604.030518	25.176023

Durchlass Log

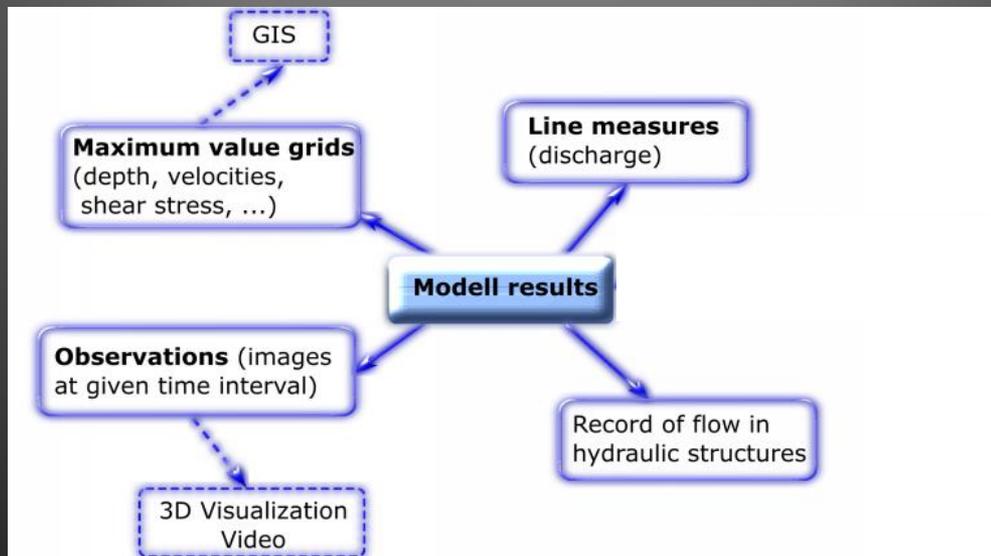
time	The time this step was logged. If you want more frequent output you can change the parameter <code>culvert_logtime</code> to a lower value (see reference). Note that no log will be written before either inlet or outlet sensor measure a flow depth > 0.
Q (avg)	This is the actual average discharge transferred since the last logstep.
Q_supposed (avg)	This is the smoothed out discharge of the last 5 timesteps. It is what would be transferred in this timestep if this amount was available in the managed zone.
Q_momentarily	The calculated discharge for this timestep
Q_uniform	Uniform discharge
Q_critical	Critical discharge
Q_gated	Gated discharge
Q_pressurized	Pressurized discharge
h1	Flow depth at inlet
v1	Flow velocity at inlet
h2	Flow depth at outlet
v2	Flow velocity at outlet
A	Wetted area of culvert
U	Wetted perimeter of culvert
flag	Case according to flow condition (see table above)

flox-GPU

Post-Processing

Allgemeines

- Resultate weisen die gleiche (hohe) Auflösung auf wie der Input
- Ergebnisse als Raster (→ Einfache Weiterverarbeitung)
- Massive Datenmengen → Komprimierung der Raster
- Hohe Auflösung des Outputs ist Fluch und Segen zugleich



«summary.txt»

- Zusammenfassung und Ablage der Simulationsparameter
- Macht Simulation nachvollziehbar
- Achtung: CFL-Wert ist hier $\frac{1}{4}$ des Eingabewerts

```
finished 2017-9-19 16:30:54
host_name BARTWS46
GPU0 GeForce GTX 750 Ti
email

##### META
project 25344
GQ 566
run holzerbach
id 0
J 9999
P 1.000000

##### SIMULATION
infile ..\examples\holzerbach\mod\in_EHQ.txt
outpath c:\flox\work\holzerbach\hq100\
runtime 413.850006
simulated_time 1800.007568
steps 178844
time_ratio 4.349421

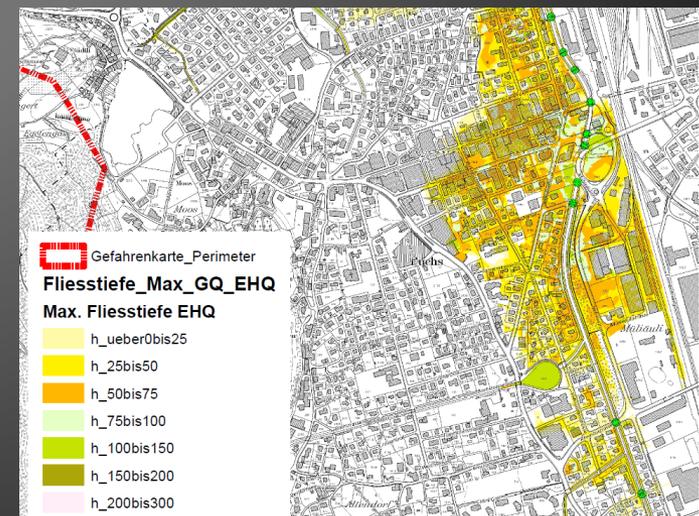
##### ENVIRONMENT
flox_revision 132
CFL 0.250000
cellsize 0.400000
cells_x 1428
cells_y 2162
theta 1.300000
epsilon_edge 0.010000
epsilon_desingularization 0.010000
2nd order Runge-Kutta
```

Resultate: Maximum Raster

- Standard-Output (Maximum-Raster):
 - Max. Wassertiefe d («d_max.tif»)
 - Max. Wasserstand w («w_max.tif»)
 - Max. Geschwindigkeit v («v_max.tif»)
 - Max. Intensität i ($d * v$) («i_max.tif»)

measures	30.11.2016 16:27	Dateiordner	
observers	07.12.2016 16:46	Dateiordner	
progress	07.12.2016 18:06	Dateiordner	
hillshade.png	07.12.2016 16:46	PNG-Bild	1'414 KB
system.png	07.12.2016 16:46	PNG-Bild	1'646 KB
d_max.tif	07.12.2016 18:06	TIFF-Bild	1'682 KB
h.tif	07.12.2016 18:06	TIFF-Bild	1'680 KB
hu.tif	07.12.2016 18:06	TIFF-Bild	1'887 KB
hv.tif	07.12.2016 18:06	TIFF-Bild	1'903 KB
i_max.tif	07.12.2016 18:06	TIFF-Bild	1'696 KB
v_max.tif	07.12.2016 18:06	TIFF-Bild	1'797 KB
w_max.tif	07.12.2016 18:06	TIFF-Bild	1'303 KB
error_log.bt	07.12.2016 16:46	TXT-Datei	0 KB
in.bt	07.12.2016 16:45	TXT-Datei	1 KB
progress_log.bt	07.12.2016 18:06	TXT-Datei	8 KB
summary.bt	07.12.2016 18:06	TXT-Datei	1 KB
time_log.bt	07.12.2016 18:06	TXT-Datei	11 KB

- Ablage in Hauptverzeichnis der Resultate
- Auswertung mit GIS möglich



Resultate: Observer (Detailausgaben)

Zeitlicher Verlauf der Variablen («observer» tag):

- Variablen: [hwuvtfE] (h w u v Tau Froude Energylevel)
- Ablage in «observers» Ordner – für jede Variable ein Unterordner
- Für jeden Ausgabezeitpunkt ein eigenes Raster mit Simulationszeit im Namen
- Detailauswertung im GIS pro Ausgabezeitpunkt möglich
- Zeitliche Visualisierung mit «floxviewer»

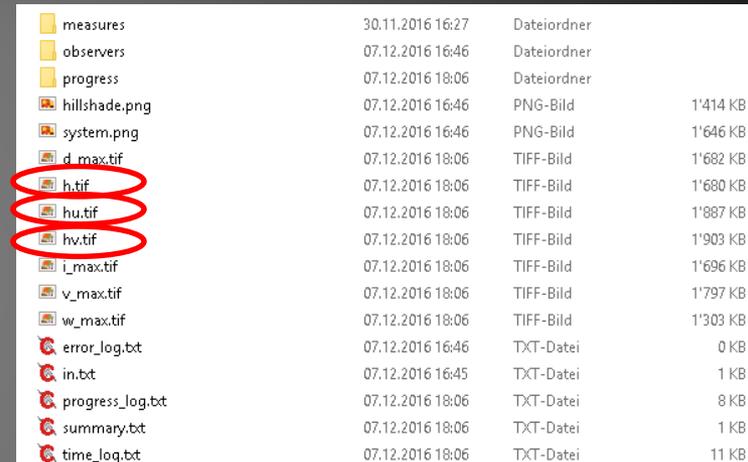
measures	30.11.2016 16:27	Dateiordner	
observers	07.12.2016 16:46	Dateiordner	
progress	07.12.2016 18:06	Dateiordner	
h	07.12.2016 18:05	Dateiordner	
w	20.09.2017 10:50	Dateiordner	
h_10.001136.tif	07.12.2016 16:48	TIFF-Bild	1'193 KB
h_10.001136.tif.aux.xml	07.12.2016 16:48	XML-Dokument	1 KB
h_20.006044.tif	07.12.2016 16:49	TIFF-Bild	1'215 KB
h_20.006044.tif.aux.xml	07.12.2016 16:49	XML-Dokument	1 KB
h_30.020632.tif	07.12.2016 16:50	TIFF-Bild	1'234 KB
h_30.020632.tif.aux.xml	07.12.2016 16:50	XML-Dokument	1 KB
h_40.022739.tif	07.12.2016 16:52	TIFF-Bild	1'252 KB
h_40.022739.tif.aux.xml	07.12.2016 16:52	XML-Dokument	1 KB
h_50.034122.tif	07.12.2016 16:53	TIFF-Bild	1'264 KB
h_50.034122.tif.aux.xml	07.12.2016 16:53	XML-Dokument	1 KB
h_60.041771.tif	07.12.2016 16:54	TIFF-Bild	1'271 KB
h_60.041771.tif.aux.xml	07.12.2016 16:54	XML-Dokument	1 KB
h_70.048973.tif	07.12.2016 16:55	TIFF-Bild	1'278 KB
h_70.048973.tif.aux.xml	07.12.2016 16:55	XML-Dokument	1 KB
h_80.053223.tif	07.12.2016 16:57	TIFF-Bild	1'286 KB
h_80.053223.tif.aux.xml	07.12.2016 16:57	XML-Dokument	1 KB
h_90.060715.tif	07.12.2016 16:59	TIFF-Bild	1'299 KB

Resultate: Hotstart

Endzustand Erhaltungsgrößen nach Simulation:

- Wasserstand h («h.tif»)
- Intensität in x-Richtung («hu.tif»)
- Intensität in y-Richtung («hv.tif»)

(Optionale Ausgabe: «output huv»)



Name	Datum	Typ	Größe
measures	30.11.2016 16:27	Dateiordner	
observers	07.12.2016 16:46	Dateiordner	
progress	07.12.2016 18:06	Dateiordner	
hillshade.png	07.12.2016 16:46	PNG-Bild	1'414 KB
system.png	07.12.2016 16:46	PNG-Bild	1'646 KB
d_max.tif	07.12.2016 18:06	TIFF-Bild	1'682 KB
h.tif	07.12.2016 18:06	TIFF-Bild	1'680 KB
hu.tif	07.12.2016 18:06	TIFF-Bild	1'887 KB
hv.tif	07.12.2016 18:06	TIFF-Bild	1'903 KB
i_max.tif	07.12.2016 18:06	TIFF-Bild	1'696 KB
v_max.tif	07.12.2016 18:06	TIFF-Bild	1'797 KB
w_max.tif	07.12.2016 18:06	TIFF-Bild	1'303 KB
error_log.txt	07.12.2016 16:46	TXT-Datei	0 KB
in.txt	07.12.2016 16:45	TXT-Datei	1 KB
progress_log.txt	07.12.2016 18:06	TXT-Datei	8 KB
summary.txt	07.12.2016 18:06	TXT-Datei	1 KB
time_log.txt	07.12.2016 18:06	TXT-Datei	11 KB

Verwendung der Grids als AB führt zu Fortführung der Simulation:

- Tags in Eingabedatei:
 - «h_grid»
 - «hu_grid»
 - «hv_grid»
- Startzeit der Simulation kann gesetzt werden: «start_time_s»
→ richtige Fortsetzung der Zeitreihen

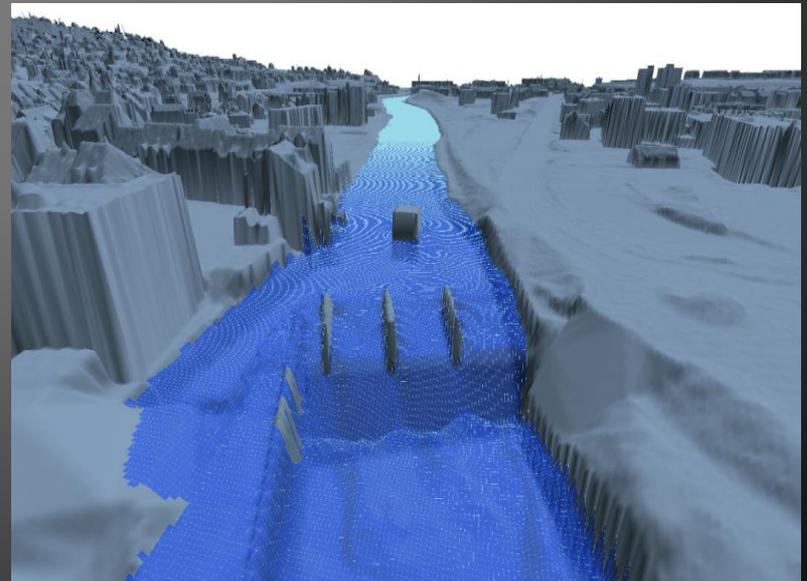
Auswertung mit GIS:

Weiterverarbeitung der Maximum-Raster mit GIS (ArcGIS, QGIS):

- Visualisierung ist einfach
- Punktwerte abfragen
- Querprofile, Längsprofile durch Resultate
- Polygonisieren, Beschneiden, Verschneiden, etc.

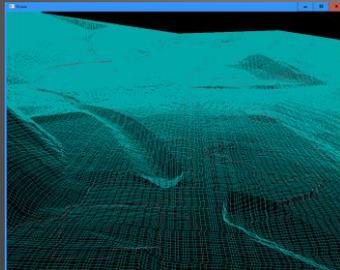
floxviewer: Visualisierung / Kontrolle / Bilderserien (I)

- Noch experimentell, wenig bedienungsfreundlich
- Genaue Darstellung der 3D Geometrie
- Performant dank GPU (OpenGL)
- Ermöglicht Kontrolle Geländehöhen (Vergleich mit Bruchkanten)
- 3D Visualisierung des Fließvorgangs
- Bilderserien (für Videos)



floxviewer: Visualisierung / Kontrolle / Bilderserien (II)

- Start als Konsolenanwendung: «floxviewer.exe [textdatei]»
- Steuerdatei als Textdatei mit Schlüsselwörtern
(Achtung Bug: immer nur 1 Leerzeichen als Abstand!)
- Steuerung:
 - Bewegung im Raum: Pfeiltasten + «q/a» + rechte Maustaste
 - Switchen zwischen Datensätzen «i» und Darstellungsart «o»
 - Start Bilderserie: «n» + Pfad mit Raster-Dateien im Steuerfile («tif_dir»)



```
# Darstellung DTM Grid
grid F:\01Int\1043\examples\holzerbach\dtm\grid\dtm.tif 169 169 169 #RGB: grey

# Darstellung Vektor Linien (Shape-Format)
vector F:\01Int\1043\examples\holzerbach\dtm\shapes\Bruchkante_L2.shp 255 0 0 #RGB: red

# StartPosition
pos 2726338.622 1222117.7 480.6
angle -168.4 -156
tilesize 4000
```

Steuerdatei:

- Darstellung grid: «grid [Pfad-zum-Grid] [RGB]»
- Mehrere Grids können überlagert werden
- Darstellung Shape-Linien: «vector [pfad-shape-datei] [RGB]»
- Angabe Startposition: «pos [x y z]»
- Einfärbung mit RGB Werten (z.B. Grau = 169 169 169, Rot = 255 0 0, Blau = 0 0 255) ⁶⁷

flox-GPU

Exkurs: Berechnung, Sonstiges

Berechnung

- Noch experimentell
- Abflussbildung / Abflusskonzentration berechnet sich in hoher Auflösung
- Angabe von Berechnungs-Grids «hydrology_grids» zu best. Zeitpunkten mit Niederschlagsintensitäten [mm/h]
- Berechnungs-Grid kann andere Ausdehnung und Zellgrößen haben
- Entspricht einer «Quelle» für jede erfasste Zelle

```
# PRECIPITATION

hydrology_grids
0 F:\01Int\1043\examples\regen\regen\regen_100_ezg_407.tif
100 F:\01Int\1043\examples\regen\regen\regen_100_ezg_408.tif
end
```

gridmaker.exe: Erstellt Raster aus TRIANGLE-Dateien

- Konsolenanwendung
- Schnell + effizient
- Möglichkeit Bruchkante zu «erzwingen» beim Mappen auf Raster

kst.exe: Erstellt Strickler-Raster aus Shape-Polygon-Datei

- Konsolenanwendung
- Schnell + effizient
- Einfaches Mappen auf Raster
- Strickler-Raster um 1 Zelle in Breite und Höhe kleiner als DTM!
(Cell-center)

Was ist geplant?

Nahziele:

- Brückenmodul
- Schütz- und Wehrmodul
- Erweiterung Messstellen («Measures»)

Fernziele:

- Programmoberfläche
- Szenarienverwaltung
- Multi-GPU (mehrere GPUs für eine Simulation verwenden)
- Komplexeres Hydrologie-Modul
- Ankopplung Kanalnetzrechnung

flox-GPU: Beispiele

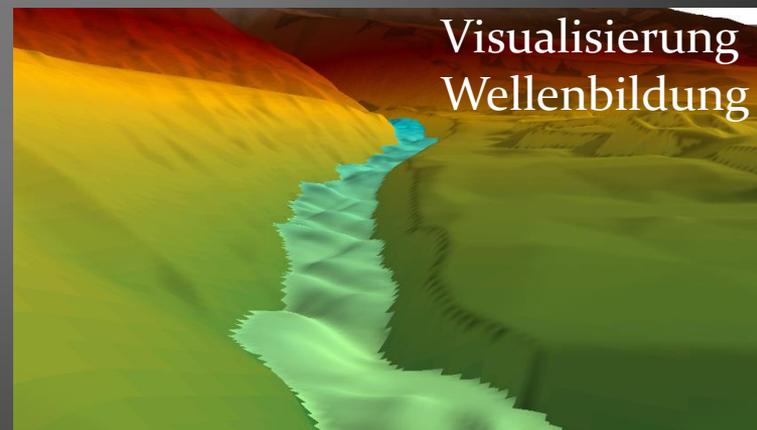
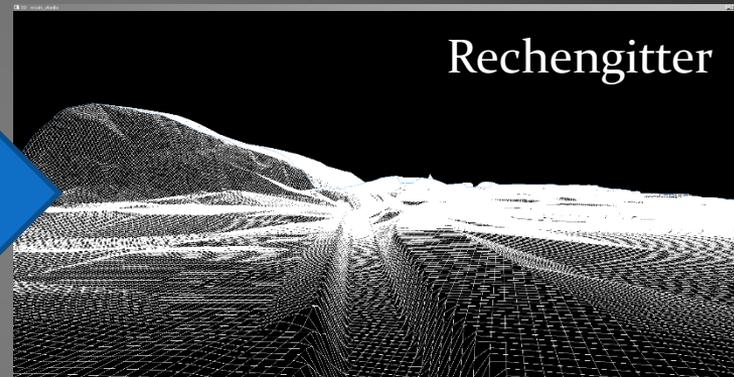
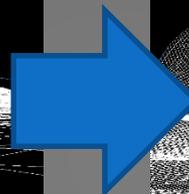
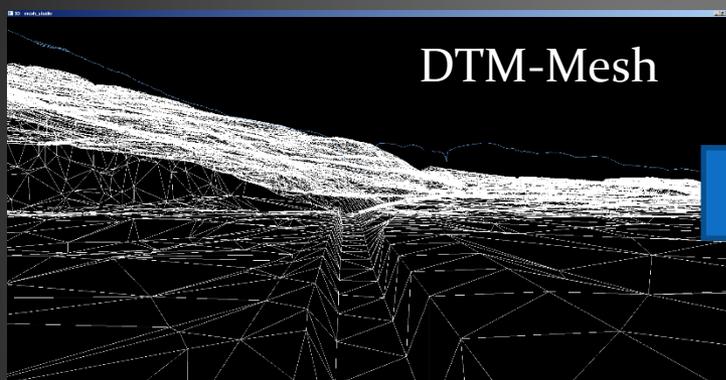
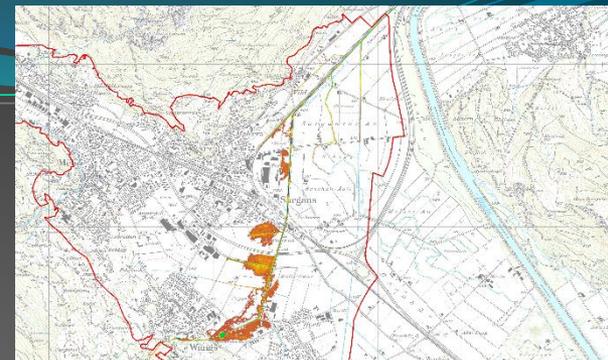
Beispiel 1: Grossbach

- Q_{\max} ca. 50 m³/s (EHQ)
- Basis Ganglinie ca. 3.5 h
- ca. 24 Durchlässe
- 5 Szenarien
- Besonderheit: grosser, flacher Perimeter
 - Bach (und Vorfluter) ca. 5.5 km zu modellieren
 - Benetzte Fläche ca. 500'000 m²
 - Retentionen im Landwirtschaftsland

Grossbach Kennwerte

- Simulierte Zeit: 10 h
- Rechenzeit: 2.5 h (Echtzeitfaktor 4x)
- Zellgrösse 0.16 m² (0.4 x 0.4 m)
- 2.9 Mio benetzte Zellen

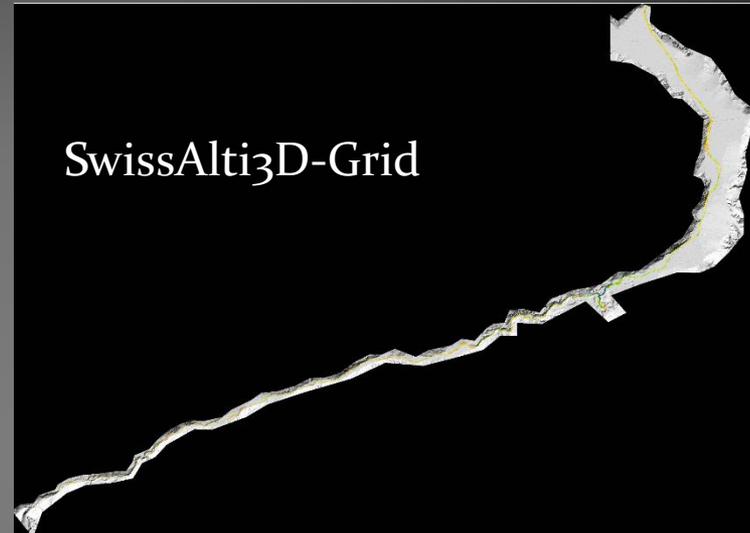
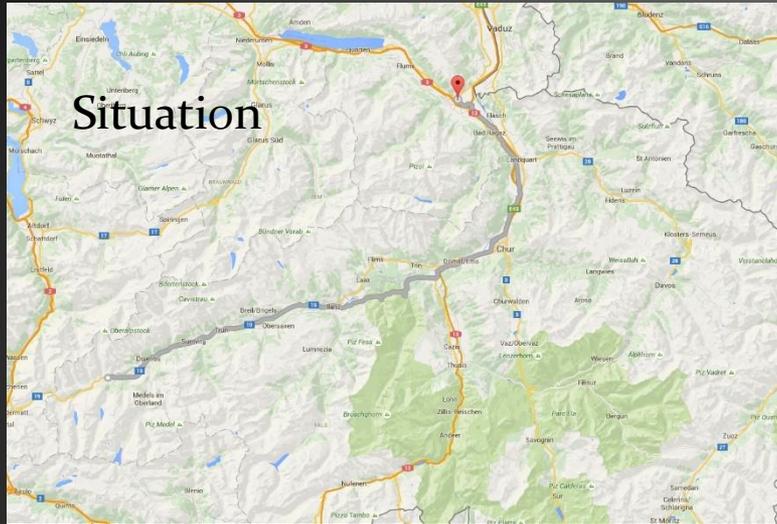
Grossbach-Modell



Bsp. Rhein:

- Modellierung des Rheins von Sedrun bis Sargans (~100km)
- Zellgrösse 4m^2 (2m x 2m), SwissAlti3D
- Blockganglinie von 100 h und $Q = 232 \text{ m}^3/\text{s}$ (mehrere Quellen)
- kaum Ausuferungen
- Netzgrösse: 54 M Zellen
- Benetzte Zellen: 1.75 M (7km^2)
- Berechnete Zellen (pro Schritt): ~3.7 M
- Zeitschritte: 17.6 M
- Berechnungsschritte: 65 Billionen
- Laufzeit 17:55 (Echtzeitfaktor 5.6x)

Rhein-Modell



Beispiel 3: Abflussbildung mit flox-GPU

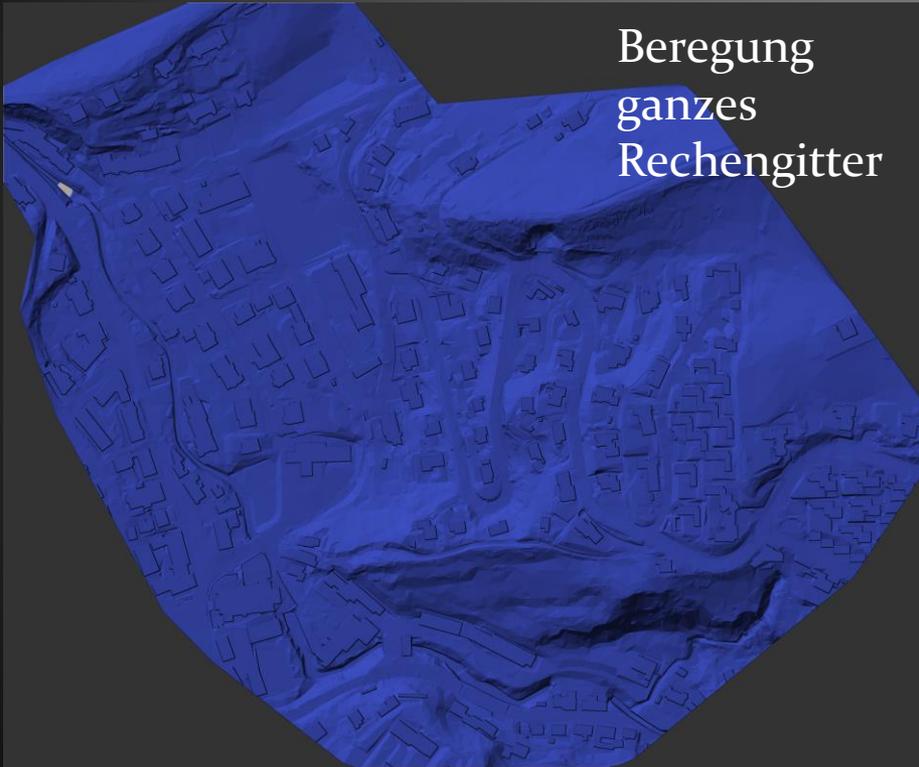
- Die hohe Leistung von flox-GPU ermöglicht das flächendeckende Berechnen von grösseren Gebieten
- Modul bisher ein Prototyp

Beispiel 3: St. Georgen

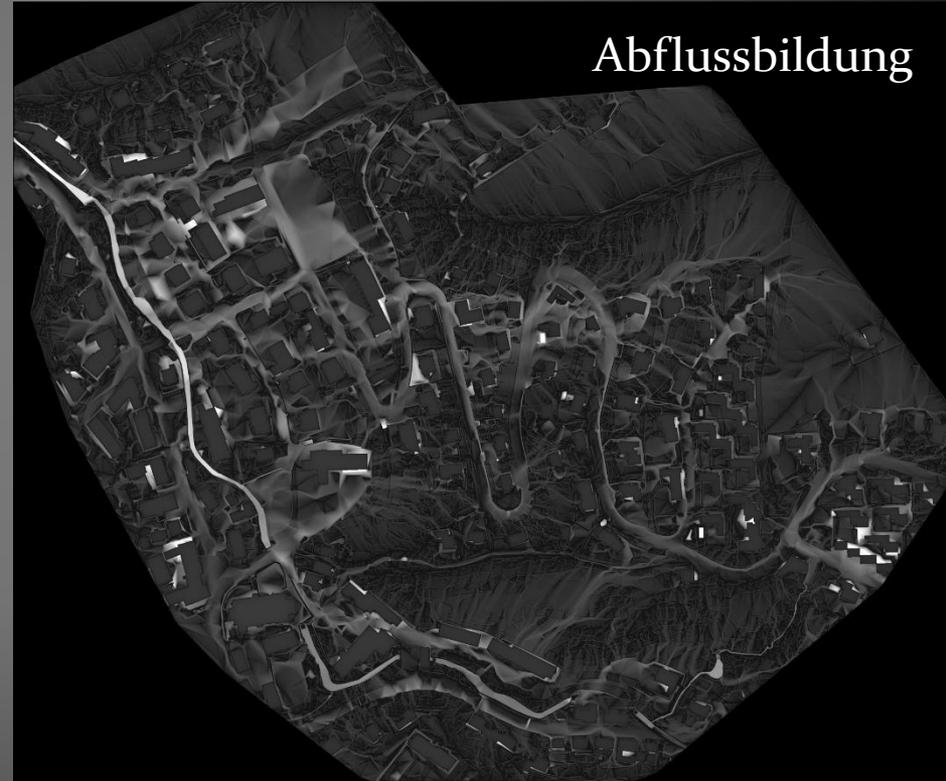
- Zellgrösse 0.04 m^2 ($0.2\text{m} \times 0.2\text{m}$)
- Gesamtfläche ca. 0.3 km^2
- 2h Beregnungszeit mit 50 mm/h
- Rechenzeit ca. 1:08 (Echtzeitfaktor = 1.8)

Beregnung St. Georgen

Beregnung
ganzes
Rechengitter



Abflussbildung



The End



Christian Volz
volz@bart.ch

flox-GPU
flox@bart.ch

Ingenieure Bart AG
info@bart.ch