Hydroinformatik II - SoSe 2024

HyBHW-S2-01-V13: Gerinnehydraulik - Übungen

Prof. Dr.-Ing. habil. Olaf Kolditz

[1]Helmholtz Centre for Environmental Research – UFZ, Leipzig

[2]Technische Universität Dresden – TUD, Dresden

[3]Center for Advanced Water Research – CAWR

[4]TUBAF-UFZ Center for Environmental Geosciences – C-EGS, Freiberg / Leipzig

Dresden, 12.07.2024

TECHNISCHE UNIVERSITÄT DRESDEN

Zeitplan: Hydroinformatik II - SoSe 2024

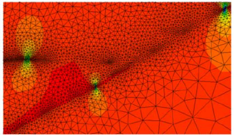
Datum	HI	II	Thema	Typ
14.06.2024	14	2-01	Einführung in die Lehrveranstaltung - Teil 2	L
14.06.2024	15	2-02	Werkzeuge | Tools	L
14.06.2024	16	2-03	Grundlagen: Kontinuumsmechanik	L
21.06.2024	17	2-04	Grundlagen: Hydromechanik	L
21.06.2024	18	2-05	Grundlagen: Partielle Partialgleichungen	L
21.06.2024	19	2-06	Übung: Analytische Lösungen	E
28.06.2024*	20	2-07	Grundlagen: Näherungsverfahren	L
28.06.2024*	21	2-08	Übung: Jupyter Diffusionsprozess	E
02.07.2024*	22	2-09	Numerik: Finite-Differenzen-Methode (explizit)	L
02.07.2024*	23	2-10	Numerik: Finite-Differenzen-Methode (implizit)	L
12.07.2024	24	2-11	Übung: Finite-Differenzen-Methoden	E
12.07.2024	25	2-12	Grundlagen: Gerinnehydraulik	L
12.07.2024	26	2-13	Übung: Gerinnehydraulik	E
19.07.2024	27	2-14	Ausblick: Grundwassermodellierung	E
19.07.2024	28	2-15	Klausur/Beleg: Besprechung zur Vorbereitung	L

*online Vorlesung

TECHNISCHE UNIVERSITÄT DRESDEN

Konzept

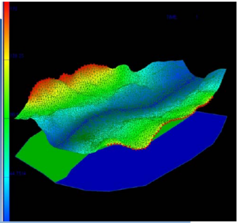
$$\frac{d\psi}{dt} = \frac{\partial\psi}{\partial t} + \mathbf{v}^E \nabla\psi$$



Basics

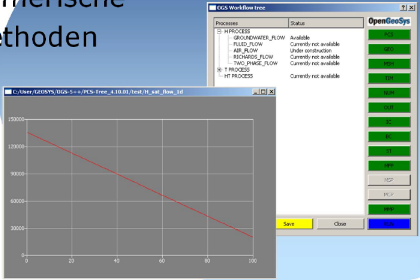
Mechanik

Anwendung



Numerische

Methoden



Programmierung

Visual C++

Prozessverständnis

TECHNISCHE UNIVERSITÄT DRESDEN

Hydroinformatik - Anwendungen

1. Abfallwirtschaft: Diffusionsprozesse
2. **Hydrology: Gerinnehydraulik (→ this)**
3. Grundwasserwirtschaft: Grundwasserhydraulik (→ next)

TECHNISCHE UNIVERSITÄT DRESDEN

Wiederholung:
BHYWI-08-12
Gerinnehydraulik - Grundlagen

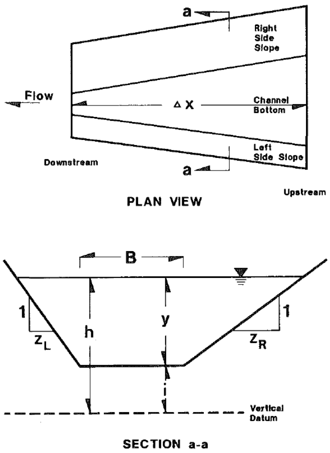


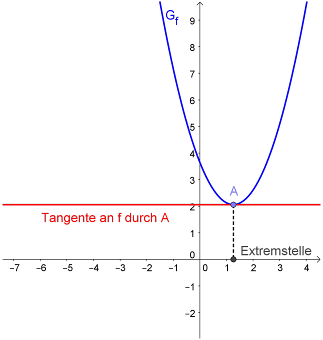
Fig.: Trapezgerinne Paine (1992)

$$y \equiv z$$

Newton-Verfahren #2

Funktional: Energieerhaltung (potentielle + kinetische + Wärme-Energie (Reibung))

$$f(h) = \left(h + \frac{Q^2}{2gA^2}\right)\Big|_{i+1} - \left(h + \frac{Q^2}{2gA^2}\right)\Big|_{i} + (x_{i+1} - x_i)\frac{S_{f,i+1} + S_{f,i}}{2} \tag{1}$$



$$\begin{aligned} f'(h) &= \frac{df}{dh} = 0 \\ &\approx \frac{f(h_{k+1}) - f(h_k)}{h_{k+1} - h_k} \\ h_{k+1} &= h_k - \frac{f(h_k)}{f'(h)} = h_k - \frac{N}{D} \end{aligned} \tag{2}$$

Quelle: `https://menzelths.github.io/Mathematik-Kursstufe/Ableitungen_Extremstellen.html`

Newton-Verfahren #3

$$f(h) = \underbrace{\left(h + \frac{Q^2}{2gA^2}\right)|_{i+1}}_{N_1} - \underbrace{\left(h + \frac{Q^2}{2gA^2}\right)|_{i}}_{N_2} + \underbrace{(x_{i+1} - x_i)\frac{S_{f,i+1} + S_{f,i}}{2}}_{N_3} \tag{3}$$

$$f'(h) = \underbrace{\frac{d}{dh}\left(h + \frac{Q^2}{2gA^2}\right)|_{i+1}}_{D_1} - \underbrace{\frac{d}{dh}\left(h + \frac{Q^2}{2gA^2}\right)|_{i}}_{D_2} + \underbrace{\frac{x_{i+1} - x_i}{2}\frac{d}{dh}(S_{f,i+1} + S_{f,i})}_{D_3} \tag{4}$$

Newton Verfahren #4

$$D_1 = \frac{d}{dh}\left(h + \frac{Q^2}{2gA^2}\right) = 1 - \frac{Q^2}{gA^3}\frac{dA}{dh} \tag{5}$$

$$\frac{dA}{dh} = \frac{d}{dh}\left(y(B + yC_4)\right) = B + 2yC_4 \tag{6}$$

$$D_1 = 1 - \frac{Q^2}{gA^3}B + 2yC_4 \tag{7}$$

Newton Verfahren #5

Bleibt noch die Differenzierung der Streckenverluste

$$D_3 = \frac{dS_f}{dh} = S_f' = \frac{d}{dh}\left(\frac{Q}{AR^{2/3}}\right)^2 \tag{8}$$

$$\frac{d}{dh}\left(\frac{Q}{AR^{2/3}}\right)^2 = Q^2 \frac{d}{dh}\left(A^{-2}R^{-4/3}\right) \tag{9}$$

$$\ldots = Q^2\left(\frac{dA^{-2}}{dh}R^{-4/3}\right) + \left(A^{-2}\frac{dR^{-4/3}}{dh}\right) \tag{10}$$

$$\ldots = Q^2\left(-2A^{-3}\frac{dA}{dh}R^{-4/3}\right) + \left(A^{-2}\frac{(-4)}{3}R^{-7/3}\frac{dR}{dh}\right) \tag{11}$$

TECHNISCHE UNIVERSITÄT DRESDEN

Newton Verfahren #6

Bleibt noch die Differenzierung der Streckenverluste

$$\frac{dS_f}{dh} = S_f' = \frac{d}{dh}\left(\frac{Q}{AR^{2/3}}\right)^2 \tag{12}$$

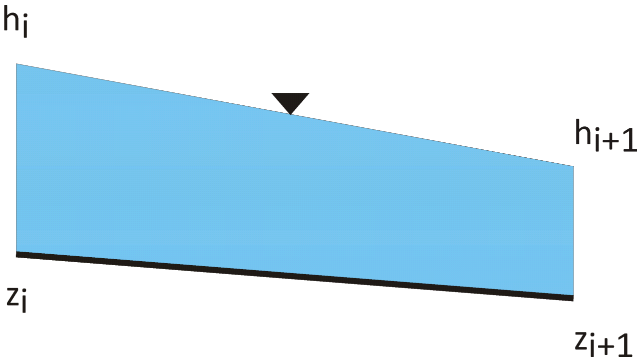
$$\begin{aligned} S_f' &= \left[Q^2(By + C_4y^2)^{10/3}\frac{4}{3}(B + C_5y)^{1/3}C_5\right] \\ &+ \left[(B + yC_5)^{4/3}\frac{-10Q^2}{3}(By + C_4y^2)^{13/3}(B + 2C_4y)\right] \end{aligned} \tag{13}$$

$$\begin{aligned} S_f' &= \frac{4}{3}Q^2C_5(By + C_4y^2)^{-10/3}(B + C_5y)^{1/3} \\ &- \frac{10}{3}Q^2(B + 2C_4y)(B + C_5y)^{4/3}(By + C_4y^2)^{-13/3} \end{aligned} \tag{14}$$

Übungen BHYWI-08-11 Gerinnehydraulik

Levels of programming ...

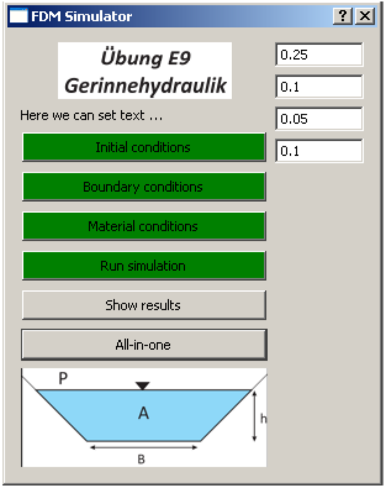
- **Q&D**: Übung 1 (BHYWI-08-05-E): Funktionalität
- **OOP**: Übung 2 (BHYWI-08-06-E): Modularität
- **GUI**: Übung 3 (BHYWI-08-07-E): Interaktion (Ausgabe)
- GUI: Übung 4 (BHYWI-08-08-E): Interaktion (Eingabe)



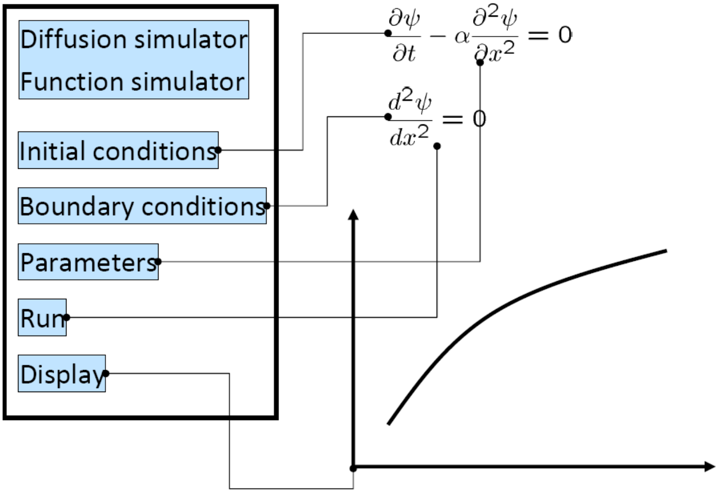
BHYWI-08-05-E > BHYWI-08-11A
BHYWI-08-06-E > BHYWI-08-11B
BHYWI-08-07-E > BHYWI-08-11C
BHYWI-08-08-E > BHYWI-08-11D

Levels of programming ...

- **Q&D**: Übung 1 (BHYWI-08-05-E): Funktionalität
- **OOP**: Übung 2 (BHYWI-08-06-E): Modularität
- **GUI**: Übung 3 (BHYWI-08-07-E): Interaktion (Ausgabe)
- GUI: Übung 4 (BHYWI-08-08-E): Interaktion (Eingabe)



Übungen BHYWI-08-11A-D Gerinnehydraulik


Diffusion simulator
Function simulator
Initial conditions
Boundary conditions
Parameters
Run
Display
$\frac{\partial \psi}{\partial t} - \alpha \frac{\partial^2 \psi}{\partial x^2} = 0$
$\frac{d^2 \psi}{dx^2} = 0$

Programmstruktur

BHYWI-08-11A

```
int main(int argc, char *argv[])
{
  // 1-Geometrie
  // 2-Anfangsbedingungen
  // 3-Randbedingungen
  // 4-Parameter
  // 5-Berechnungsgroessen
  // 6-Berechnung (Newton-Verfahren)
  // 7-Ausgabe der Ergebnisse
    // File
    // x-y Plot
}

class channel (:: geo, ic, bc, mat, run, out)
```

Listing: Programmstruktur und OOP

TECHNISCHE UNIVERSITÄT DRESDEN

Newton-Verfahren

BHYWI-08-11A

$$h_{k+1} = h_k - \frac{f(h_k)}{f'(h_k)} = h_k - \frac{N}{D} \tag{15}$$

```
void channel::newton-step
{...
  double N,N1,N2,N3,D,D1,D2,D21,D22;
  for(int i=0;i<n-1;i++)
  {
    N1 = pow(discharge,2)/pow(wetted_cross_section[i+1],2) + gravity*u_old[i+1];
    N2 = pow(discharge,2)/pow(wetted_cross_section[i],2) + gravity*u_old[i];
    N3 = gravity*(bed_slope - (friction_slope[i+1]+friction_slope[i])/2.)*(x[i+1]-x[i]);
    N = N1 - N2 - N3;
    D1 = pow(discharge,2)/pow(wetted_cross_section[i],3) * (bottom_width+2.*m*u_old[i]) - gravity;
    D21 = friction_law_exponent*2.*(sqrt(1+m*m))/wetted_perimeter[i];
    D22 = (1.+friction_law_exponent)/wetted_cross_section[i] * (bottom_width+2.*m*u_old[i]);
    D2 = gravity*friction_slope[i]*(D21-D22)*(x[i+1]-x[i]);
    D = D1 + D2;
    u_new[i] = u_old[i] - N/D;
  }
...
}
```

Listing: Newton-Schritt

Definition der konstanten Modellparameter

BHYWI-08-11A

```
// Parameter
double discharge = 0.05; // Volumenfliessrate [m3/s]
double gravity = 9.81; // [m/s2]
double friction_law_exponent = 0.5; // Chezy, Manning-Strickler [-]
double error_tolerance = 1e-3; // [m]
double bed_slope = 0.0004; // [m/m]
double bottom_width = 1.; // [m]
double m = 1.; //
double friction_coefficient = 10.; //
```

Definition der funktionalen Modellparameter

BHYWI-08-11A

```
// Berechnungsgroessen
  // Newton-Verfahren
double wetted_perimeter[n];
double wetted_cross_section[n];
double water_level_elevation[n];
  // Abgeleitete Groessen (Ausgabe)
double hydraulic_radius[n];
double flow_velocity[n];
double Froude_number[n];
double friction_slope[n];
```

Berechnung der funktionalen Modellparameter

BHYWI-08-11A

```
for(int i=0;i<n;i++)
{
  wetted_perimeter[i] = bottom_width + 2.*sqrt(1.+m*m)*u_old[i];
  wetted_cross_section[i] = (bottom_width + m*u_old[i])*u_old[i];
  hydraulic_radius[i] = wetted_cross_section[i] / wetted_perimeter[i];
  water_level_elevation[i] = bottom_elevation[i] + u_old[i];
  flow_velocity[i] = discharge/wetted_cross_section[i];
  Froude_number[i] = flow_velocity[i]/(sqrt(gravity*
  wetted_cross_section[i]\
                     /sqrt(bottom_width*bottom_width+4.*m*
  wetted_cross_section[i])));
  friction_slope[i] = pow(flow_velocity[i]/(friction_coefficient*
                      pow(hydraulic_radius[i],friction_law_exponent))
  ,2);
}
```

Ausgabe > File

BHYWI-08-11A

```cpp
void channel::out
{...
  ofstream out_file("out.txt");
  out_file.precision(4);
  out_file << "Water depth (old):\t";
  for(int i=0;i<n;i++)
  {
    out_file << "\t" << u_old[i] << " ";
  }
  out_file << endl;
...
  out_file << "Water depth (new):\t";
  for(int i=0;i<n;i++)
  {
    out_file << "\t" << u_new[i] << " ";
  }
  out_file << endl;
  out_file.close();
...}
}
```

Listing: Test Ausgabe

out.txt - Editor

Datei Bearbeiten Format Ansicht ?

```
0.25
0.1
Iteration: 0
Water depth (old):                          0.25      0.25      0.25      0.25      0.25      0.25      0.25      0.25      0.25      0.25      0.1
Wetted perimeter:                           1.71      1.71      1.71      1.71      1.71      1.71      1.71      1.71      1.71      1.71      1.28
Wetted cross section:             0.313     0.313     0.313     0.313     0.313     0.313     0.313     0.313     0.313     0.313     0.11
Hydraulic radius:                           0.183     0.183     0.183     0.183     0.183     0.183     0.183     0.183     0.183     0.183     0.0857
Water level elevation:            0.29      0.286     0.282     0.278     0.274     0.27      0.266     0.262     0.258     0.254     0.1
Flow velocity:                    0.16      0.16      0.16      0.16      0.16      0.16      0.16      0.16      0.16      0.16      0.455
Froude number:                    0.112     0.112     0.112     0.112     0.112     0.112     0.112     0.112     0.112     0.112     0.479
Friction slope:                   0.0014    0.0014    0.0014    0.0014    0.0014    0.0014    0.0014    0.0014    0.0014    0.0014    0.0241
Water depth (new):                          0.259     0.259     0.259     0.259     0.259     0.259     0.259     0.259     0.259     0.242     0.1
Iteration: 1
Water depth (old):                          0.259     0.259     0.259     0.259     0.259     0.259     0.259     0.259     0.259     0.242     0.1
Wetted perimeter:                           1.73      1.73      1.73      1.73      1.73      1.73      1.73      1.73      1.73      1.69      1.28
Wetted cross section:             0.326     0.326     0.326     0.326     0.326     0.326     0.326     0.326     0.326     0.301     0.11
Hydraulic radius:                           0.188     0.188     0.188     0.188     0.188     0.188     0.188     0.188     0.188     0.179     0.0857
Water level elevation:            0.299     0.295     0.291     0.287     0.283     0.279     0.275     0.271     0.267     0.246     0.1
Flow velocity:                    0.153     0.153     0.153     0.153     0.153     0.153     0.153     0.153     0.153     0.166     0.455
Froude number:                    0.105     0.105     0.105     0.105     0.105     0.105     0.105     0.105     0.105     0.118     0.479
Friction slope:                   0.00125             0.00125             0.00125             0.00125             0.00125             0.00125
Water depth (new):                          0.267     0.267     0.267     0.267     0.267     0.267     0.267     0.267     0.253     0.242     0.1
Iteration: 2
Water depth (old):                          0.267     0.267     0.267     0.267     0.267     0.267     0.267     0.267     0.253     0.242     0.1
Wetted perimeter:                           1.76      1.76      1.76      1.76      1.76      1.76      1.76      1.76      1.72      1.69      1.28
Wetted cross section:             0.339     0.339     0.339     0.339     0.339     0.339     0.339     0.339     0.317     0.301     0.11
Hydraulic radius:                           0.193     0.193     0.193     0.193     0.193     0.193     0.193     0.193     0.185     0.179     0.0857
Water level elevation:            0.307     0.303     0.299     0.295     0.291     0.287     0.283     0.279     0.261     0.246     0.1
Flow velocity:                    0.148     0.148     0.148     0.148     0.148     0.148     0.148     0.148     0.158     0.166     0.455
Froude number:                    0.1       0.1       0.1       0.1       0.1       0.1       0.1       0.1       0.11      0.118     0.479
Friction slope:                   0.00113             0.00113             0.00113             0.00113             0.00113             0.00113
Water depth (new):                          0.274     0.274     0.274     0.274     0.274     0.274     0.274     0.262     0.253     0.242     0.1
Iteration: 3
Water depth (old):                          0.274     0.274     0.274     0.274     0.274     0.274     0.274     0.262     0.253     0.242     0.1
Wetted perimeter:                           1.78      1.78      1.78      1.78      1.78      1.78      1.78      1.74      1.72      1.69      1.28
Wetted cross section:             0.349     0.349     0.349     0.349     0.349     0.349     0.349     0.331     0.317     0.301     0.11
Hydraulic radius:                           0.197     0.197     0.197     0.197     0.197     0.197     0.197     0.19      0.185     0.179     0.0857
```

Berechnung: Newtonverfahren

BHYWI-08-11B

$$h_{k+1} = h_k - \frac{f(h_k)}{f'(h_k)} = h_k - \frac{N}{D} \tag{16}$$

```
// Newton iteration loop
  for(int k=0;k<kn;k++)
  {...
    RunNewtonStep();
  ...}
```

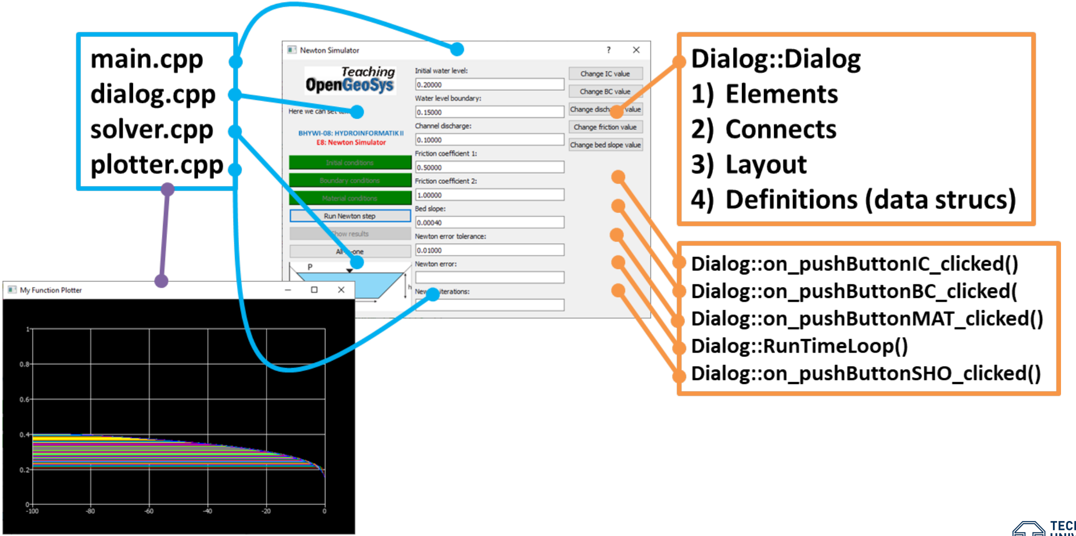
$$\text{bis} \quad \| h_{k+1} - h_k \| < \epsilon \tag{17}$$

TECHNISCHE UNIVERSITÄT DRESDEN

Anlage
BHYWI-08-11C
Gerinnehydraulik
GUI

TECHNISCHE UNIVERSITÄT DRESDEN

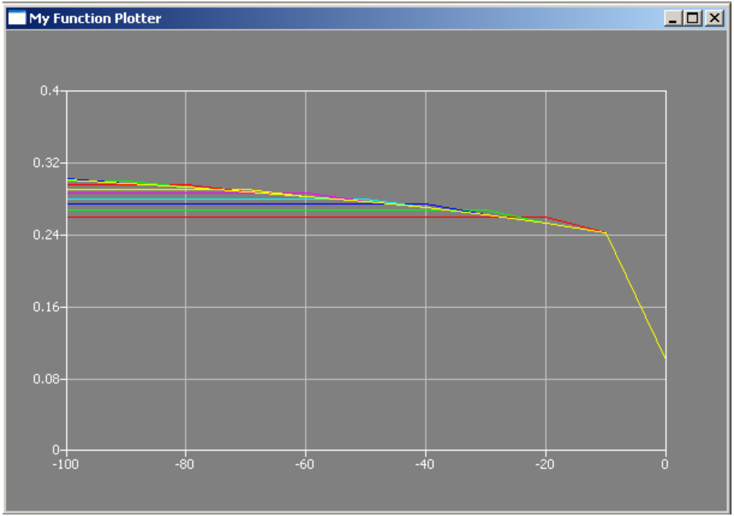
Qt Prinzip



TECHNISCHE UNIVERSITÄT DRESDEN

plotter()

BHYWI-08-11C



TECHNISCHE UNIVERSITÄT DRESDEN

Anlage
EX10-gerinne-python
Gerinnehydraulik
Newton-Verfahren

EX10-gerinne-python