BHYWI-08: Semester-Fahrplan

Vorlesungen

Datum	V	Thema
13.04.2018	01	Einführung / Qt Installation
20.04.2018	02	Grundlagen: Kontinuumsmechanik
27.04.2018	03	Grundlagen: Hydromechanik
04.05.2018	04	Grundlagen: Partielle Differentialgleichungen
11.05.2018	05	Grundlagen: Numerik, Qt Übung: Funktionsrechner
18.05.2018	06	Numerik: Finite Differenzen Methode I (explizit)
01.06.2018	07	Numerik: Finite Differenzen Methode II (implizit)
08.06.2018	08	Gerinnehydraulik: Theorie – Grundlagen
15.06.2018	09	Gerinnehydraulik: Programmierung, Übung 1
22.06.2018	10	Gerinnehydraulik: Programmierung, Übung 2
29.06.2018	11	Grundwassermodellierung: Catchment Übung
06.07.2018	12	Grundwassermodellierung: Datenbasierte Methoden I
13.07.2018	13	Grundwassermodellierung: Datenbasierte Methoden II
20.07.2018	14	Klausurvorbereitung

TECHNISCHE UNIVERSITÄT DRESDEN

Hydroinformatik II
"Prozesssimulation und Systemanalyse"
BHYWI-08-07 @ 2018
Finite-Differenzen-Verfahren II

Olaf Kolditz

*Helmholtz Centre for Environmental Research – UFZ

[1]Technische Universität Dresden – TUDD

[2]Centre for Advanced Water Research – CAWR

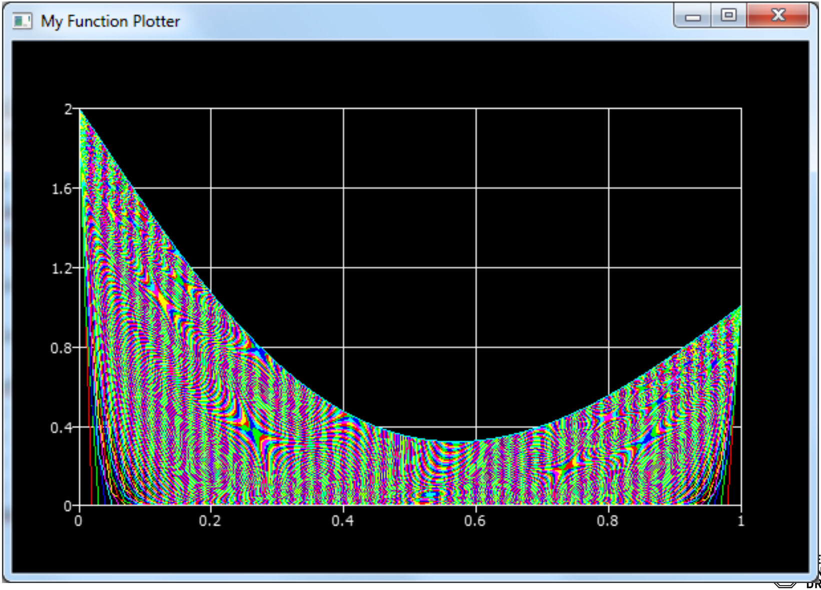
08.06.2018 - Dresden

Fahrplan fr heute ...
Implicit FDM for diffusion equation (section 4.2)

1. L08-07: Implicit FDM for diffusion equation
2. E08-07: Implementation of iFDM
3. Qt Basics

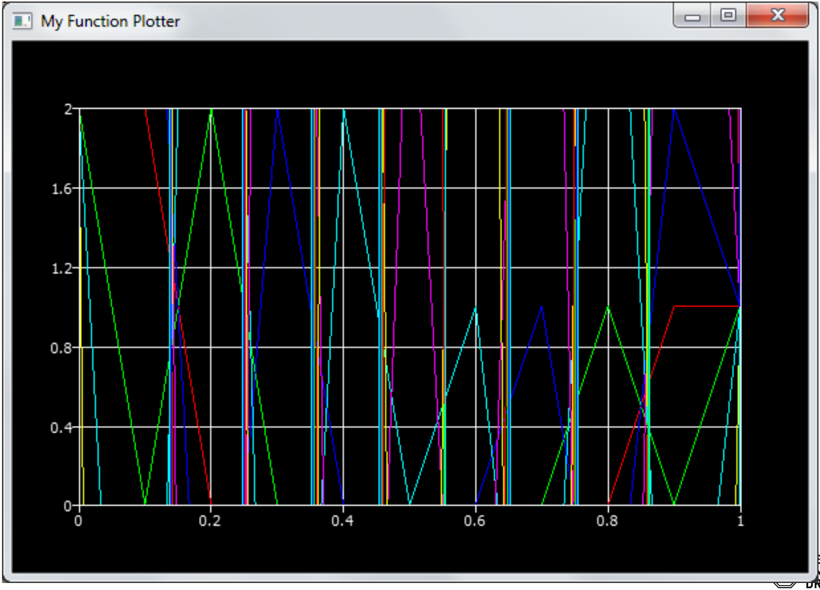
TECHNISCHE UNIVERSITÄT DRESDEN

Letzte Vorlesung: explizite FDM



TECHNISCHE UNIVERSITÄT DRESDEN

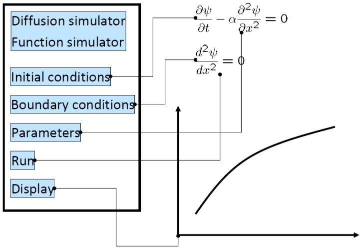
Letzte Vorlesung: explizite FDM - Zeitschrittbegrenzung

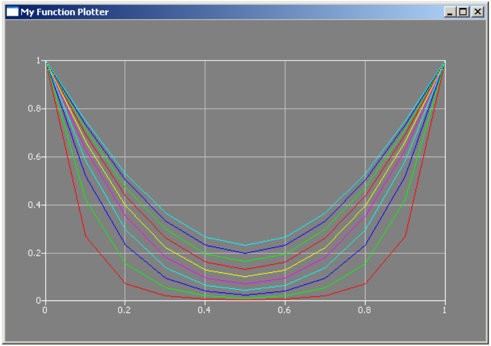


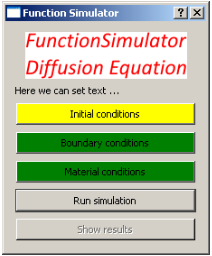
Letzte Vorlesung: explizite FDM - Zeitschrittbegrenzung

$$Ne = \alpha \frac{\Delta t}{\Delta x^2} \leq 0.5 \qquad (1)$$

Ziel der Vorlesung







TECHNISCHE UNIVERSITÄT DRESDEN

Implicite FDM - Theory #1 (Skript 4.2)

- PDE for diffusion processes

$$\frac{\partial u}{\partial t} - \alpha \frac{\partial^2 u}{\partial x^2} = 0 \tag{2}$$

- Time discretization

$$\left[\frac{\partial u}{\partial t}\right]_j^n \approx \frac{u_j^{n+1} - u_j^n}{\Delta t} \tag{3}$$

- Forward time / centered space

$$\left[\frac{\partial^2 u}{\partial x^2}\right]_j^{n+1} \approx \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{\Delta x^2} \tag{4}$$

- (Current time / centered space)

$$\left[\frac{\partial^2 u}{\partial x^2}\right]_j^n \approx \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} \tag{5}$$

TECHNISCHE UNIVERSITÄT DRESDEN

Implicite FDM - Theory #2 (Skript 4.2)

- Substitute into PDE

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} - \alpha \frac{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}{\Delta x^2} = 0 \tag{6}$$

- Algebraic equation (index notation)

$$\frac{\alpha \Delta t}{\Delta x^2} \left(-u_{j-1}^{n+1} + 2u_j^{n+1} - u_{j+1}^{n+1}\right) + u_j^{n+1} = u_j^n \tag{7}$$

- Algebraic equation (matrix notation)

$$\mathbf{Ax} = \mathbf{b} \tag{8}$$

- Explain steps with black board

TECHNISCHE UNIVERSITÄT DRESDEN

- Algebraic equation (matrix notation)

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{9}$$

- Algebraic equation (index notation)

$$Ne\left(-u_{j-1}^{n+1} + 2u_j^{n+1} - u_{j+1}^{n+1}\right) + u_j^{n+1} = u_j^n \tag{10}$$

- Let's take a closer look ...

$$\underbrace{\begin{bmatrix} 1+2Ne & -Ne & & & \\ -Ne & \dots & \dots & & \\ & \dots & \dots & \dots & \\ & & \dots & \dots & -Ne \\ & & & -Ne & 1+2Ne \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} u_0 \\ u_1 \\ \dots \\ u_{n-1} \\ u_n \end{bmatrix}}_{\mathbf{x}} = \underbrace{\begin{bmatrix} b_0 \\ b_1 \\ \dots \\ b_{n-1} \\ b_n \end{bmatrix}}_{\mathbf{b}} \quad (11$$

TECHNISCHE UNIVERSITÄT DRESDEN

Implementation #1

- Data structures (as usual ...)

```
Dialog::Dialog(QWidget *parent) : QDialog(parent)
{
  matrix = new double[n*n];
  vecb = new double[n];
  vecx = new double[n];
}

Dialog::~Dialog()
{
  delete [] matrix;
  delete [] vecb;
  delete [] vecx;
}
```

Implementation #2

- Functions (a pain in the neck ...)

```
AssembleEquationSystem();
Gauss(matrix,vecb,vecx,n);

void Dialog::AssembleEquationSystem()
{...
  int i,j;
  // Matrix entries
  for(i=0;i<n;i++)
  {
    vecb[i] = u_old[i]; // RHS Vektor
    for(j=0;j<n;j++)
    {
      matrix[i*n+j] = 0.0;
      if(i==j) // Hauptdiagonale
        matrix[i*n+j] = 1. + 2.*Ne;
      else if(abs((i-j))==1) // Nebendiagonalen
        matrix[i*n+j] = - Ne;
    }
  }
...}
```

TECHNISCHE UNIVERSITÄT DRESDEN

- Boundary conditions - concept

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{12}$$

$$[1\,0\,0\ldots 0]\begin{bmatrix} u_0 \\ u_1 \\ \ldots \\ u_n \end{bmatrix} = \begin{bmatrix} u_0 \\ 0 \\ \ldots \\ 0 \end{bmatrix} \tag{13}$$

- Boundary conditions - implementation

```
void Dialog::AssembleEquationSystem()
{...
  // Treat boundary conditions
  for(i=0;i<n;i++)
    for(j=0;j<n;j++)
    {
      if(i==0||i==n-1)
        matrix[i*n+j] = 0.0;
    }
  for(i=0;i<n;i++)
  {
    if(i!=0&&i!=n-1)
      continue;
    for(j=0;j<n;j++)
    {
       if(i==j)
         matrix[i*n+j] = 1.0;
       else
         matrix[i*n+j] = 0.0;
    }
  }
}
```

- 1

$$a_{11}u_1 + a_{12}u_2 = b_1 \tag{14}$$

$$a_{21}u_1 + a_{22}u_2 = b_2 \tag{15}$$

- 2

$$a_{21}\frac{a_{11}}{a_{21}}u_1 + a_{22}\frac{a_{11}}{a_{21}}u_2 = \frac{a_{11}}{a_{21}}b_2 \tag{16}$$

- 3

$$\left(\frac{a_{22}a_{11}}{a_{21}} - a_{12}\right)u_2 = \frac{a_{11}}{a_{21}}b_2 - b_1 \tag{17}$$

- 4

$$u_2 = \frac{\frac{a_{11}}{a_{21}}b_2 - b_1}{\frac{a_{22}a_{11}}{a_{21}} - a_{12}} \tag{18}$$

Implementation #5

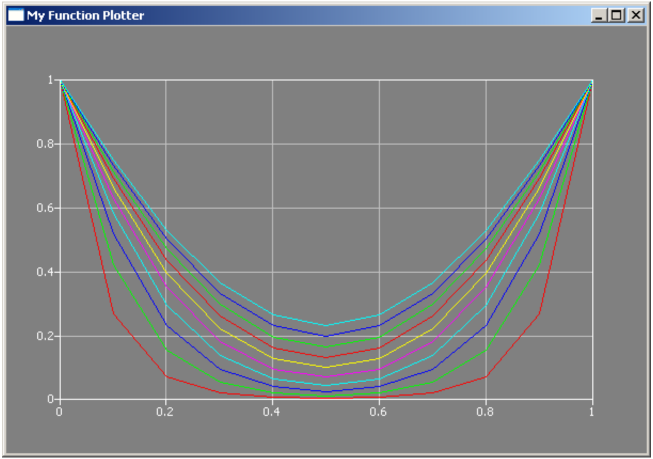
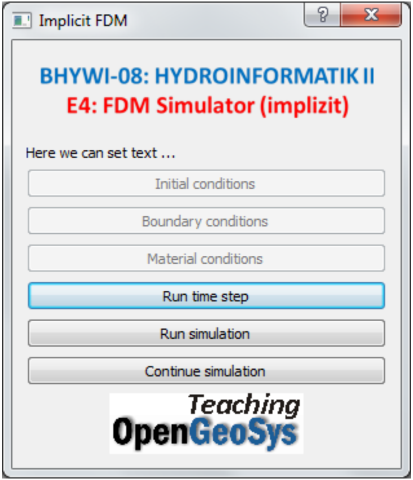


Figure: Zeitliche Entwicklung des Diffusionsprofils - implizites Verfahren (Wahoo...)

Implementation #6: Run multiple time steps



Einzelne Zeitschritte

Mehrere Zeitschritte

Weiterfhren der Berechnung

```
void RunTimeStep();
void RunTimeLoop();
void ContinueTimeLoop();
```

TECHNISCHE UNIVERSITÄT DRESDEN

C++Basics (Star Wars ...)

- ... just a star (*)

```
void Dialog::on_pushButtonSHO_clicked()
{
  Plotter *plotter = new Plotter;
  ...
  plotter->show();
}
```

- ... for better plotting (in your life)

```
void Dialog::SHOBetter()
{
  ...
  plotterAIO->show();
}
```

TECHNISCHE UNIVERSITÄT DRESDEN

C++Basics (... the difference (the whole story))

- Plotter declaration

```
class Dialog : public QDialog
{...
private:
  Plotter *plotterAIO;
}
```

- Plotter declaration ... otherwise

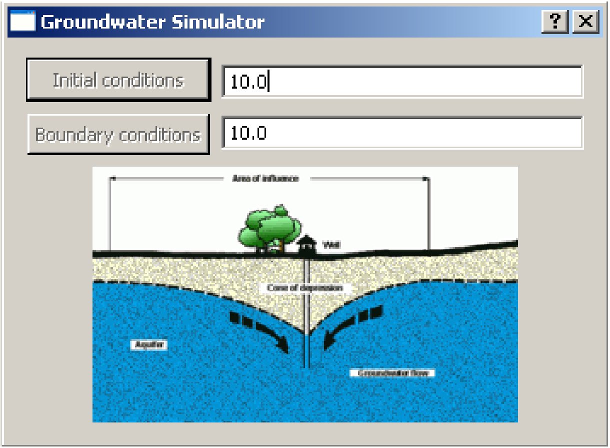
```
Dialog::Dialog(QWidget *parent) : QDialog(parent)
{...
  plotterAIO = new Plotter;
}
```

- Finally ready to use ...

```
void Dialog::SHOBetter()
{...
  plotterAIO->show();
}
```

TECHNISCHE UNIVERSITÄT DRESDEN

- QLineEdit



TECHNISCHE UNIVERSITÄT DRESDEN

Datum	E	Übungen
11.05.2018	01	Qt: Hallo World
11.05.2018	02	Qt: Funktionsrechner
18.05.2018	03	Qt: Explizite Finite-Differenzen-Methode
01.06.2018	04	Qt: Implizite Finite-Differenzen-Methode
15.06.2018	05	Qt: Gerinnehydraulik I (QAD)
22.06.2018	06	Qt: Gerinnehydraulik II (OOP)
22.06.2018	07	Qt: Gerinnehydraulik III (interaktiv)
29.06.2018	08	Qt: Gerinnehydraulik IV (interaktiv)