# Avsluttende prosjekt - INF5620

Henrik Andersen Sveinsson

December 5, 2013

Prosjektet går ut på å løse den ikke-lineære diffusjonslikningen:

$$\rho u_t = \nabla \cdot (\alpha(u)\nabla u) + f(\mathbf{x}, t) \tag{1}$$

med initialbetingelse:

$$u(\mathbf{x}, 0) = I(\mathbf{x}) \tag{2}$$

og Neumannbetingelse:

$$\frac{\partial u}{\partial n} = 0 \tag{3}$$

## 1 Tidsdiskretisering

Bruker backward Euler på den tidsderiverte:

$$\rho \frac{u - u_1}{\Delta t} = \nabla \cdot (\alpha(u)\nabla u) + f(\mathbf{x}, t) \tag{4}$$

$$\rho u - \Delta t \nabla \cdot (\alpha(u)\nabla u) = \rho u_1 + \Delta t f(\mathbf{x}, t) \tag{5}$$

Den variasjonelle formuleringen for det romlige problemet i hvert tidssteg blir:

$$\rho(u, v) - \Delta t \left[ (\alpha(u)\nabla u, \nabla v) + \cancel{\left( \alpha(u)\frac{\partial u}{\partial n}, v \right)_{\partial\Omega}} \right] = \rho(u_1, v) + \Delta t (f, v) \tag{6}$$

Der Neumannbetingelsen eliminerer grenseflateintegralet. Merk at i den siste likningen er $u = \sum c_j \psi_j$, og likningen setter residualet ortogonalt på løsningen i den opprinnelige likningen.

### 1.1 Variasjonell formulering av initialbetingelsen

$$u_e(\mathbf{x}, 0) \approx u(\mathbf{x}, 0) = \sum c_j \psi_j \tag{7}$$

Setter:

$$R = u(\mathbf{x}, 0) - I(\mathbf{x}) \tag{8}$$

Krever at:

$$\int_\Omega R v = 0 \tag{9}$$

Altså at:

$$(u, v) = (I, v) \tag{10}$$

Med elementfunksjoner får vi at initialbetingelsen blir:

$$u(\mathbf{x}, 0) = \sum I(\mathbf{x}_{\nu(j)})\varphi_{\nu(j)} \tag{11}$$

## 2 Picard-iterasjon

Lineariserer den variasjonelle formuleringen for å kunne anvende Picard-iterasjon.

$$\rho(u, v) - \Delta t(\alpha(u_{-})\nabla u, \nabla v) = \rho(u_1, v) + \Delta t(f, v) \tag{12}$$

Setter så inn at $v$ er alle $\psi$ i et eller annet rom, som er det samme rommet som vi ekspanderer $u$ i.

$$\sum_{j\in\mathcal{I}_s} \left[\rho(\psi_j, \psi_i) - \Delta t(\alpha(u_{-})\nabla\psi_j, \nabla\psi_i)\right] c_j = \rho(u_{-}, \psi_i) + \Delta t(f, \psi_i) \quad \forall \quad i \tag{13}$$

Picard-iterasjon på denne, vil si å løse dette likningssettet med hensyn på $u = \sum c_j\psi_j$, og oppdatere $u_{-}$.

## 3 Implementasjon

Tidsintegrasjon med 1-stegs picarditerasjon er implementert. E/h er konstant (vist ved testing for varierende oppløsning).

## 4 Manufactured solution

Likningen skal løses i 1D på $\Omega = [0, 1]$. Den ikke-lineære faktoren er $\alpha = 1 + u^2$ og løsningen skal være:

$$u(x, t) = t\int_0^x q(1-q)dq = tx^2\left(\frac{1}{2} - \frac{x}{3}\right) \tag{14}$$

Med sympy vises det da at kildeleddet f skal være:

$$f(x, t) = -\frac{1}{3}\rho x^3 + \frac{1}{2}\rho x^2 + \frac{8}{9}t^3x^7 - \frac{28}{9}t^3x^6 + \frac{7}{2}t^3x^5 - \frac{5}{4}t^3x^4 + 2tx - t \tag{15}$$

Forskjellen mellom løsningen her og løsningen i FEniCS bør være av orden $\Delta t$, siden jeg bruker et Backward Euler skjema.

Innsetting av dette i den numeriske metoden gir en feil som går som $\Delta t$, som vist i figur 1.

Dette viser at feilen som følge av picarditerasjonen ikke er av større orden enn feilen Backward Euler gir. Jeg anser at denne også løser oppgave g, selv om jeg ikke har satt inn den korrigerte produserte løsningen.

## 5 Feilkilder i programmene

Feilkilder:

- Programmeringsblunders

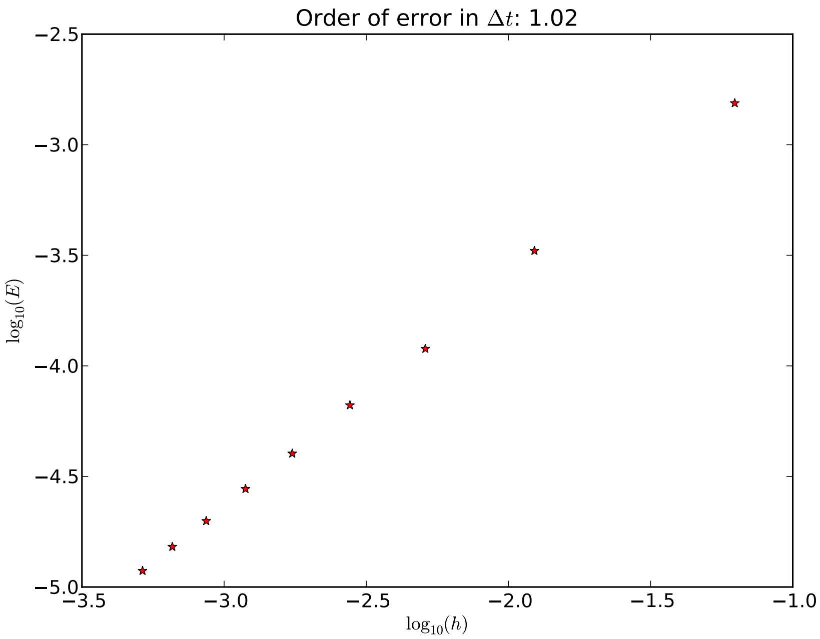


Figure 1: Figuren viser at feilen med *manufactured solution* nesten er i orden $\Delta t$.
.

- Trunkeringsfeil
- Avrundingsfeil (sannsynligvis små)
- Picarditerasjon