

DATORÖVNING 4 — FIXPUNKTSITERATION

Allmänt. Dokumentera ditt arbete i ett pdf-dokument. Spara alla filer så att du kan läsa på inför tentamen. Det räcker inte att din kompis har filerna. Datorövningarna examineras vid tentamen. Samarbete uppmuntras, men detta är inget grupparbete. Varje student måste göra sina egna datorprogram och sina egna dokument.

Mål.

Matematik: Fixpunktsekvation $x = g(x)$. Fixpunktsiteration.
Programmering: Funktion. Iteration.

Litteratur. Adams, kap. 4. Jönsson, kap. 6, 7.

Instruktioner. Skriv funktionsfiler (m-filer) för de nämnda funktionerna. Dokumentera allt ditt arbete i ett word-dokument. Exportera slutligen word-dokumentet till pdf-format. Det görs med 'print' och skrivare 'PDF995'. Slutprodukten skall alltså vara pdf. Visa upp för din lärare som prickar av att du gjort övningen.

Skapa en ny filkatalog ("directory") **studio-4** för denna övning.

Uppgifter.

1. Fixpunktsalgoritmen är

- (a) Givet: en kontraktion $g : I \rightarrow I$, en startpunkt $x_0 \in I$ och en tolerans TOL.
- (b) Start: $i = 1$, $x_i = x_0$, $x_{i+1} = g(x_i)$.
- (c) Om $|x_{i+1} - x_i| \leq \text{TOL}$, stoppa. Annars, $i = i + 1$, $x_{i+1} = g(x_i)$, upprepa (c).

Skriv ett program som implementerar algoritmen enligt programskalet [fixpoint1.m](#). Kopiera filen genom att klicka på denna länk eller gå till länken "matlab" på kurshemsidan och ladda ned den därifrån. Använd **while**-loop.

2. Använd programmet på följande fixpunktsekvationer.

- (1) $x = x/2 + 1/x$
- (2) $x = 2/x$
- (3) $\cos(x) = x$

Du måste själv hitta en startpunkt.

3. Observera hur de beräknade följderna konvergerar eller divergerar. Hur många steg behövs för att felet ska bli mindre än 10^{-10} ?

4. Det är onödigt att spara hela följden x_i . Istället för i och x_i räcker det att använda två variabler x och x_1 :

- (a) Givet: en kontraktion $g : I \rightarrow I$, en startpunkt $x_0 \in I$ och en tolerans TOL.
- (b) Start: $x_1 = x_0$, $x = g(x_1)$.
- (c) Om $|x_1 - x| \leq \text{TOL}$, stoppa. Annars, $x_1 = x$, $x = g(x_1)$, upprepa (c).

Skriv ett program som implementerar denna algoritm enligt programskalet [fixpoint.m](#).

5. System av ekvationer. En viktig fördel med fixpunktsiterationen jämfört med bisektionsalgoritmen är att den fungerar även för system av ekvationer. Det gör inte bisektion. Nackdelen är att det är ofta svårt att skriva ekvationssystemet på fixpunktsform så att iterationen konvergerar.

Prova nu ditt program `fixpoint.m` på följande ekvationssystem:

$$5x_1 + 2x_2 = -2$$

$$x_1 + 6x_2 = 1$$

Skriv om på fixpunktform genom att lösa ut x_1, x_2 ur diagonaltermerna:

$$x_1 = (-2 - 2x_2)/5$$

$$x_2 = (1 - x_1)/6$$

Detta blir bra (liten Lipschitz-konstant, $L < 1$) eftersom diagonaltermerna är stora. Skriv sedan en funktionsfil

```
function y=gfunk1(x)
    y(1,1)=(-2-2*x(2))/5;
    y(2,1)=(1-x(1))/6;
```

Kör sedan `fixpoint` med en kolonnvektor som startpunkt:

```
>> x=fixpoint(@gfunk1,[0;0],1e-7)
```

Lösningen är $(-0.5, 0.25)$.

/stig