Transformer och differentialekvationer för M3 2009/2010

Inlämningsuppgift 3. Sturm-Liouvilles egenvärdesproblem.

I denna uppgift skall du arbeta med följande egenvärdesproblem av Sturm-Liouvilles typ:

$$\begin{cases} -y''(x) = \lambda(a + bx + cx^2)y(x) \\ a_1y'(0) - a_2y(0) = 0 \\ b_1y'(1) + b_2y(1) = 0 \end{cases}$$

De värden på $a, b, c, \ldots$ som du skall använda finner du i ditt personnummer: $xxxabc$–$a_1a_2b_1b_2$. Alla nollor måste du ersätta med 10:or.

1. Bestäm med metoden i kompendiet sid. 54–55 dels ett intervall som innehåller det minsta egenvärdet, dels ett intervall som innehåller det näst minsta egenvärdet till egenvärdesproblemet ovan. Observera att formeln på sidan 55 inte kan tillämpas. Man måste "lösa" en transcendent ekvation $f(x) = 0$. Det finns exempelvis ett Matlab-kommando 'fsolve' och ett Mathematica-kommando 'FindRoot' som kan användas. Egentligen är man inte så intresserad av någon större precision i lösningen, så det kan räcka med att man plottar $f(x)$ och läser av intervall som innehåller rötterna. Det är i så fall bäst att skriva ekvationen så att $f(x)$ inte har några singulariteter.

2. Bestäm med hjälp av iteration och Schwarzkvoter det minsta egenvärdet till problemet ovan. Låt startvärdet $u_0(x)$ vara ett andragradspolynom som uppfyller randvillkoren. Utnyttja de intervall du bestämde i deluppgift 1 dels för att kontrollera rimligheten i det värde för $\lambda_{\min}$ som iterationsmetoden ger dig, dels för att uppskatta noggrannheten i detta värde med hjälp av Sats 5.7b i kompendiet. Iterationen skall avbrytas då relativa felet i egenvärdet är (enligt Sats 5.7b) högst 10^{-6}. Rita även upp några grafer till $u_{k-1}(x)/u_k(x)$, för att se hur de allt bättre approximerar λ.

Anmärkning: Vill man bygga upp en lista av Schwarzkonstanter i Mathematica har man glädje av kommandot `AppendTo`, som utökar en lista med ett element på slutet. Ett enkelt exempel får illustrera idén. Iterera formeln $s_{k+1} = \frac{s_k}{2} + \frac{1}{s_k}$ utgående från $s_0 = 1$ och bygg upp en lista av resultaten. Skriv

```
s={1.}
```

upprepa sedan kommandot

```
s = AppendTo[s, s[[-1]]/2 + 1/s[[-1]]]
```

Mer elegant och praktiskt är kanske att definiera en procedur för iterationen. I Mathematica är en procedur helt enkelt en följd av kommandon inuti en parentes:

```
s={1.};
iter:=(s1=s[[-1]]/2+1/s[[-1]];AppendTo[s,s1])
```

Man behöver då bara upprepa kommandot `iter`.

I Maple kan man till exempel skriva

```
s:=[1];
iter:= s->[op(s), evalf(s[-1]/2+1/s[-1])];
```

Upprepa sedan `s:=iter(s);`.