Matematisk modellering og numeriske metoder

Lektion 17

Morten Grud Rasmussen

16. november 2015

1 Mangeskridtsmetoder til løsning af førsteordens ODE'er

[Bogens afsnit 21.2 side 908]

1.1 Adams-Bashforth-metoder

Som tidligere betragter vi et IVP af typen

$$y'(x) = f(x, y(x)) \qquad \text{med} \qquad y(x_0) = y_0. \tag{1}$$

De numeriske metoder til løsning af problemer som ovenstående, vi hidtil har set på, har været såkaldte *enkeltskridtsmetoder*, idet de har baseret sig på at udregne næste skridt ud fra det foregående, hvilket har den fordel, at man kan benytte sig af samme fremgangsmåde fra begyndelsespunktet og frem. Adams-Bashforth-metoder går ud på at udnytte, at man med kendskab til flere tidligere skridt kan lave en polynomiumsapproksimation p_k af funktionen $x \mapsto f(x, y(x)) \in \mathbb{R}$, som jo afhænger af den ukendte funktion y. Her angiver k graden af polynomiet. Integrerer vi ODE'en i (1) over $[x_n, x_{n+1}]$, får vi

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} y'(x)\,\mathrm{d}x = \int_{x_n}^{x_{n+1}} f(x, y(x))\,\mathrm{d}x \approx \int_{x_n}^{x_{n+1}} p_k(x)\,\mathrm{d}x. \tag{2}$$

Hvis vi eksempelvis antager, at vi kender de fire punkter

$$y_n \approx y(x_n), \quad y_{n-1} \approx y(x_{n-1}), \quad y_{n-2} \approx y(x_{n-2}) \quad \text{og} \quad y_{n-3} \approx y(x_{n-3})$$

og dermed

$$\begin{aligned} f_n &= f(x_n, y_n) \approx f(x_n, y(x_n)), \\ f_{n-1} &= f(x_{n-1}, y_{n-1}) \approx f(x_{n-1}, y(x_{n-1})), \\ f_{n-2} &= f(x_{n-2}, y_{n-2}) \approx f(x_{n-2}, y(x_{n-2})), \end{aligned}$$

og

$$f_{n-3} = f(x_{n-3}, y_{n-3}) \approx f(x_{n-3}, y(x_{n-3})),$$

hvor som sædvanlig $x_n = x_0 + nh$, så kan vi approksimere $f(x, y(x))$ med et tredjegradspolynomium p_3 gennem (x_i, f_i) for $i = n-3, n-2, n-1, n$.

Efter en omgang beregningsmæssig lirumlarum når man frem til

$$\int_{x_n}^{x_{n+1}} p_3(x)\,\mathrm{d}x = \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$$

og dermed vha. (2)

$$y_{n+1} = y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}),$$

som kan vises at have lokal diskretiseringsfejl af orden 5 og dermed global diskretiseringsfejl af orden 4.

1.2 Adams-Moulton-metoder

Som i skridtet fra Newton-metoden til den baglæns Newton-metode kan man ændre lidt i en Adams-Bashforth-metode og få en forbedret metode for den pris, at man går fra en eksplicit til en implicit metode. Vi skal altså i stedet for at bruge punkterne x_i for $i = n-3, n-2, n-1, n$ tage udgangspunkt i x_i for $i = n-2, n-1, n, n+1$. Efter lidt fiksfakseri, hvor vi integrerer tredjegradspolynomiet $\tilde{p}_3$ gennem (x_i, f_i) for $i = n-2, n-1, n, n+1$ over intervallet $[x_n, x_{n+1}]$ fås

$$y_{n+1} = y_n + \frac{h}{24}(9f_{n+1} + 19f_n - 5f_{n-1} + f_{n-2}), \tag{3}$$

hvor $f_{n+1} = f(x_{n+1}, y_{n+1})$ og de andre f_i'er er som før. Denne formel kaldes en *Adams-Moulton-formel* og er som sagt implicit, idet y_{n+1} også indgår på højresiden gennem f_{n+1}. Er f en tilpas rar funktion, kan man nu isolere y_{n+1}, men er dette ikke muligt (eller hvis det er for besværligt), så genanvender vi bare tricket fra Heuns og lignende metoder, hvor vi erstatter $f_{n+1} = f(x_{n+1}, y_{n+1})$ med $\tilde{f}_{n+1} = f(x_{n+1}, \tilde{y}_{n+1})$, hvor $\tilde{y}_{n+1}$ er en *prædiktor*. Normalt anvendes i udgangspunktet prædiktoren

$$\tilde{y}_{n+1} = y_n + \frac{h}{24}(55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3})$$

fra Adams-Bashforth-metoden. Da vi kan estimere fejlen i det $(n+1)$'ste skridt ε_{n+1} ved

$$\varepsilon_{n+1} \approx \frac{1}{15}(y_{n+1} - \tilde{y}_{n+1}),$$

kan man, hvis fejlen i y_{n+1} estimeres til at være uacceptabelt stor, bruge y_{n+1} som en *ny* prædiktor $\tilde{y}_{n+1}$ og genindsætte i (3). Denne proces kan gentages, indtil fejlen estimeres til at være under en passende valgt tolerance. Denne *prædiktor-korrektor*-metode kaldes *Adams-Moulton-metoden af fjerde orden*. Adams-Moulton-metoden er generelt meget mere præcis end en Adams-Bashforth-metode af samme orden og er desuden numerisk stabil.

1.3 Eksempel

Eksempel 1.1 (Bogens Example 1 side 911)**.** Vi vil gentage succesen fra sidste lektion med IVP'et $y'(x) = x + y(x)$ og $y(0) = 0$, blot hvor vi anvender Adams-Moulton-metoden af fjerde orden, og skridtlængden sætter vi til $h = 0.2$. Det viser sig fluks at være umuligt, idet vi kun kender $y_0 = 0$. Vi finder derfor y_1, y_2 og y_3 vha. RK4 og kan nu anvende Adams-Moulton-metoden. Det kan ses af nedenstående tabel, at præcisionen øges betragteligt af korrektionen i forhold til prædiktoren.

n	x_n	RK4-opstart	Værdi af prædiktor	Korrigeret værdi	Eksakt værdi	Fejl
0	0.0	0.000000			0.000000	0.000000
1	0.2	0.021400			0.021403	0.000003
2	0.4	0.091818			0.091825	0.000007
3	0.6	0.222107			0.222119	0.000012
4	0.8		0.425361	0.425529	0.425541	0.000012
5	1.0		0.718066	0.718270	0.718282	0.000012
6	1.2		1.119855	1.120106	1.120117	0.000011
7	1.4		1.654885	1.655191	1.655200	0.000009
8	1.6		2.352653	2.353026	2.353032	0.000006
9	1.8		3.249190	3.249646	3.249647	0.000001
10	2.0		4.388505	4.389062	4.389056	-0.000006

2 Metoder til førsteordenssystemer og højereordens ODE'er

[Bogens afsnit 21.3 side 912]

2.1 Repetition af systemer af ODE'er

Vi genkalder fra lektion 6, hvad et system af førsteordens ODE'er er, og hvordan en n'te-ordens ODE kan konverteres til et system af førsteordens ODE'er. Lad $Y = \begin{pmatrix} y_1 & y_2 & \cdots & y_n \end{pmatrix}$, være en vektorfunktion. Da er

$$Y'(x) = F(x, Y(x)) \tag{4}$$

et *system af førsteordens ODE'er*, evt. med *begyndelsesbetingelse* $Y(x_0) = Y_0 = \begin{pmatrix} y_{00} & y_{10} & \cdots & y_{n0} \end{pmatrix}$. Et *$n$'te-ordens IVP*

$$y^{(n)} = f(x, y(x), y'(x), \ldots, y^{(n-1)}) \qquad \text{med} \qquad y(x_0) = K_1, y'(x_0) = K_2, \ldots, y^{(n-1)}(x_0) = K_n$$

kan omskrives til et system af førsteordens ODE'er med begyndelsesbetingelse

$$Y(x_0) = \begin{pmatrix} y_1(x_0) & y_2(x_0) & \cdots & y_n(x_0) \end{pmatrix} = \begin{pmatrix} K_1 & K_2 & \cdots & K_n \end{pmatrix}$$

ved at skrive

$$y_1 = y, \quad y_2 = y', \quad \cdots, \quad y_n = y^{(n-1)}$$

og sætte

$$\begin{aligned} y_1' &= y_2 \\ y_2' &= y_3 \\ &\vdots \\ y_{n-1}' &= y_n \\ y_n' &= f(x, y_1, \ldots, y_n), \end{aligned}$$

dvs. systemet er på formen (4) med $Y = \begin{pmatrix} y_1 & \cdots & y_n \end{pmatrix}$, $F = \begin{pmatrix} f_1 & \cdots & f_n \end{pmatrix}$ og $f_i(x, y_1, \ldots, y_n) = y_{i+1}$ for $i = 1, \ldots, n-1$ og $f_n(x, y_1, \ldots, y_n) = f(x, y_1, y_2, \ldots, y_n)$.

2.2 Euler-metoden

Euler-metoden kan anvendes nærmest uændret på systemer, idet vi nu blot sætter

$$Y_{n+1} = Y_n + hF(x_n, Y_n),$$

med eneste forskel, at der er tale om en følge af vektorer. Men da en n'te-ordens ODE kan skrives som et system af førsteordens ODE'er, kan vi altså også anvende Euler-metoden på højereordens-systemer.

Eksempel 2.1 (Example 1 bogen side 913)**.** Vi ser på andenordens IVP'et

$$y''(x) + 2y'(x) + 0.75y(x) = 0 \qquad \text{med} \qquad y(0) = 3 \qquad \text{og} \qquad y'(0) = -2.5.$$

Dette svarer til systemet

$$\begin{aligned} y_1' &= y_2, & y_1(0) &= 3, \\ y_2' &= -2y_2 - 0.75y_1, & y_2(0) &= -2.5. \end{aligned}$$

Euler-metoden bliver altså til

$$\begin{aligned} y_{1,n+1} &= y_{1,n} + hy_{2,n}, & y_{1,0} &= 3, \\ y_{2,n+1} &= y_{2,n} + h(-2y_{2,n} - 0.75y_{1,n}), & y_{2,0} &= -2.5. \end{aligned}$$

Det er altså ikke væsensforskelligt fra tilfældet med én førsteordens ODE, bortset fra, at der nu er to følger, som er koblede.

2.3 Runge-Kutta-metoder

Ligesom med Euler-metoden, som jo er den simpleste Runge-Kutta-metode, kan de andre Runge-Kutta-metoder nemt oversættes til systemer. Vi eksemplificerer med en oversættelse af RK4-metoden. Begyndelsesbetingelsen lyder

$$Y(x_0) = Y_0$$

og de fire prædiktorer er

$$\begin{aligned} K_1 &= hF(x_n, Y_n), \\ K_2 &= hF(x_n + \tfrac{1}{2}h, Y_n + \tfrac{1}{2}K_1), \\ K_3 &= hF(x_n + \tfrac{1}{2}h, Y_n + \tfrac{1}{2}K_2) \end{aligned}$$

og

$$K_4 = hF(x_n + h, Y_n + k_3).$$

Med denne notation bliver RK4 for systemer

$$Y_{n+1} = Y_n + \tfrac{1}{6}(K_1 + 2K_2 + 2K_3 + K_4).$$

2.4 Runge-Kutta-Nyström-metoder

Hvis man tager en andenordens ODE $y''(x) = f(x, y(x), y'(x))$ med tilhørende begyndelsesbetingelser $y(x_0) = y_0$ og $y'(x_0) = y'_0$, omformulerer til et system, anvender en Runge-Kutta-metode, og oversætter tilbage til den oprindelige andenordens ODE, så får man en såkaldt Runge-Kutta-Nyström-metode. Et populært eksemplar er følgende.

$$
\begin{aligned}
k_1 &= \tfrac{1}{2}hf(x_n, y_n, y'_n), & k &= \tfrac{1}{2}h(y'_n + \tfrac{1}{2}k_1), \\
k_2 &= \tfrac{1}{2}hf(x_n + \tfrac{1}{2}h, y_n + k, y'_n + k_1), & & \\
k_3 &= \tfrac{1}{2}hf(x_n + \tfrac{1}{2}h, y_n + k, y'_n + k_2), & l &= h(y'_n + k_3)
\end{aligned}
$$

og

$$k_4 = \tfrac{1}{2}hf(x_n + h, y_n + l, y'_n + 2k_3)$$

hvor vi bemærker, at vi omgår vektorformuleringen, men til gengæld har både k_i'erne og l og k i anden søjle til at holde styr på de to "dimensioner." For at finde approksimationen af y_{n+1} af $y(x_{n+1})$ i punktet $x_{n+1} = x_0 + (n+1)h$ sættes de fire k_i'er ind i

$$y_{n+1} = y_n + h(y'_n + \tfrac{1}{3}(k_1 + k_2 + k_3)),$$

hvor

$$y'_{n+1} = y'_n + \tfrac{1}{3}(k_1 + 2k_2 + 2k_3 + k_4)$$

approksimerer $y'(x_{n+1})$.

I specialtilfældet $y''(x) = f(x, y(x))$, altså hvor f er uafhængig af y', kan man omgå l og k og får

$$
\begin{aligned}
k_1 &= \tfrac{1}{2}hf(x_n, y_n), \\
k_2 &= \tfrac{1}{2}hf(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}h(y'_n + \tfrac{1}{2}k_1)) = k_3, \\
k_4 &= \tfrac{1}{2}hf(x_n + h, y_n + h(y'_n + k_2), \\
y_{n+1} &= y_n + h(y'_n + \tfrac{1}{3}(k_1 + 2k_2))
\end{aligned}
$$

og

$$y'_{n+1} = y'_n + \tfrac{1}{3}(k_1 + 4k_2 + k_4).$$

2.5 Eksempler

I bogen findes en række eksempler på brugen af ovenstående metoder. Den studerende rådes til selv at studere eksemplerne, hvis der er tvivl om brugen af metoderne.

2.6 Baglæns Euler for systemer

Baglæns Euler kan naturligvis også anvendes på systemer, og igen er formen den samme med eneste forskel værende, at der indgår vektorer i stedet for tal.

$$Y_{n+1} = Y_n + hF(x_{n+1}, Y_{n+1}).$$

Som tidligere for baglæns Euler gælder det, at metoden er implicit, og vi skal altså kunne løse ligningen for Y_{n+1} for at få en eksplicit metode ud af det. Om dette er muligt, afhænger af F, ligesom det før afhang af f. Metodens anvendelighed er tæt knyttet til det faktum, at den er stabil selv for *stive* systemer.

Eksempel 2.2 (Bogens Example 4 side 917)**.** Vi vil løse IVP'et

$$y''(x) + 11y'(x) + 10y(x) = 10x + 11 \qquad \text{med} \qquad y(0) = 2 \qquad \text{og} \qquad y'(0) = -10.$$

Problemet kan naturligvis løses eksakt, men vi vil illustrere, at det er et stift system, som for store skridtlængder h fører til ustabilitet for direkte metoder som eksempelvis RK4, mens det for baglæns Euler er stabilt for alle værdier af h, omend fejlen vokser som funktion af h.

Først skal vi konvertere ODE'en til et førsteordenssystem i $y_1 = y$ og $y_2 = y'$:

$$\begin{aligned} y_1'(x) &= y_2(x), & y_1(0) &= 2, \\ y_2'(x) &= -10y_1(x) - 11y_2(x) + 10x + 11, & y_2(0) &= -10. \end{aligned}$$

Baglæns Euler giver nu

$$\begin{aligned} y_{1,n+1} &= y_{1,n} + hy_{2,n+1}, & y_{1,0} &= 2, \\ y_{2,n+1} &= y_{2,n} + h(-10y_{1,n+1} - 11y_{2,n+1} + 10x_{n+1} + 11), & y_{2,0} &= -10. \end{aligned}$$

Løses ligningssystemet for $y_{1,n+1}$ og $y_{2,n+1}$ (med $x_{n+1} = x_n + h$) fås

$$\begin{aligned} y_{1,n+1} &= \frac{1}{1 + 11h + 10h^2}\big((1 + 11h)y_{1,n} + hy_{2,n} + 10h^2x_n + 11h^2 + 10h^3\big), \\ y_{2,n+1} &= \frac{1}{1 + 11h + 10h^2}\big(-10hy_{1,n} + y_{2,n} + 10hx_n + 11h + 10h^2\big). \end{aligned}$$

Tabellen nedenfor opsummerer resultaterne af at anvende baglæns Euler, Euler og RK4 for forskellige værdier af skridtlængden h. Det ses, at baglæns Euler er stabil for begge (og faktisk for alle) værdier af h, mens Euler-metoden er stabil for $h = 0.1$ men ustabil for $h = 0.2$, og RK4 er stabil for $h = 0.2$ men ustabil for $h = 0.3$.

x	Baglæns Euler $h = 0.2$	Baglæns Euler $h = 0.4$	Almindelig Euler $h = 0.1$	Almindelig Euler $h = 0.2$	RK4 $h = 0.2$	RK4 $h = 0.3$	Eksakt
0.0	2.00000	2.00000	2.00000	2.00000	2.00000	2.00000	2.00000
0.2	1.36667		1.01000	0.00000	1.35207		1.15407
0.4	1.20556	1.31429	1.56100	2.04000	1.18144		1.08864
0.6	1.21574		1.13144	0.11200	1.18585	3.03947	1.15129
0.8	1.29460	1.35020	1.23047	2.20960	1.26168		1.24966
1.0	1.40599		1.34868	0.32768	1.37200		1.36792
1.2	1.53627	1.57243	1.48243	2.46214	1.50257	5.07569	1.50120
1.4	1.67954		1.62877	0.60972	1.64706		1.64660
1.6	1.83272	1.86191	1.78530	2.76777	1.80205		1.80190
1.8	1.99386		1.95009	0.93422	1.96535	8.72329	1.96530
2.0	2.16152	2.18625	2.12158	3.10737	2.13536		2.13534