

Numerisk løsning av ordinære differensialligninger (ODE)

1 Bakgrunn

Et system av m første ordens ordinære differensialligninger er gitt ved

$$\begin{aligned}x'_1 &= f_1(t, x_1, \dots, x_m), \\x'_2 &= f_2(t, x_1, \dots, x_m), \\&\vdots \\x'_m &= f_m(t, x_1, \dots, x_m),\end{aligned}$$

eller på vektorform

$$\mathbf{x}' = \mathbf{f}(t, \mathbf{x}),$$

hvor

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}, \quad \mathbf{f}(t, \mathbf{x}) = \begin{bmatrix} f_1(t, x_1, \dots, x_m) \\ f_2(t, x_1, \dots, x_m) \\ \vdots \\ f_m(t, x_1, \dots, x_m) \end{bmatrix}.$$

Dette kalles et *initialverdiproblem* (IVP) hvis løsningen er gitt i et punkt t_0 ,

$$\begin{aligned}x_1(t_0) &= x_{1,0}, \\x_2(t_0) &= x_{2,0}, \\&\vdots \\x_m(t_0) &= x_{m,0}.\end{aligned}$$

Eksempel: *Et eksempel på Lotka-Volterra ligningene (Rovdyr/bytte-modell)*

$$\begin{aligned}x'_1 &= x_1 - x_1x_2, \\x'_2 &= x_1x_2 - 2x_2.\end{aligned}$$

1.1 Autonome system

En ODE kalles *autonom* hvis f ikke er en funksjon av t , men kun av $\mathbf{x}$. Lotka-Volterra ligningene er et eksempel på et autonomt system. Et ikke-autonomt system kan gjøres autonomt ved å legge til en ekstra ligning. Ved å sette $x_0 = t$ gir det

$$x'_0 = 1, \quad x_0(t_0) = t_0.$$

Man får så et system av $m + 1$ ligninger hvor t er erstattet med x_0 .

1.2 Høyere ordens ligninger

Et høyere ordens initialverdiproblem er gitt ved

$$\begin{aligned}x^{(m)} &= f(t, x, x', x'', \dots, x^{(m-1)}), \\x(t_0) &= x_0, \\x'(t_0) &= x'_0, \\&\vdots \\x^{(m-1)}(t_0) &= x_0^{(m-1)},\end{aligned}$$

hvor $x^{(m)} = d^m x / dt^m$. Dette kan skrives om til et system av 1.ordens ligninger ved å innføre

$$\begin{aligned}x_1 &= x, \\x_2 &= x', \\&\vdots \\x_m &= x^{(m-1)}.\end{aligned}$$

Dette gir systemet

$$\begin{aligned}\mathbf{x}' &= \mathbf{f}(t, \mathbf{x}), \\ \mathbf{x}(t_0) &= \mathbf{x}_0,\end{aligned}$$

hvor

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{m-1} \\ x_m \end{bmatrix}, \quad \mathbf{f}(t, \mathbf{x}) = \begin{bmatrix} x_2 \\ x_3 \\ \vdots \\ x_m \\ f(t, x_1, \dots, x_m) \end{bmatrix}, \quad \mathbf{x}_0 = \begin{bmatrix} x_0 \\ x'_0 \\ \vdots \\ x_0^{(m-2)} \\ x_0^{(m-1)} \end{bmatrix}.$$

Vi vil konkludere denne seksjonen med noen eksistens- og entydighetsresultater.

Definisjon: En funksjon $\mathbf{f} : \mathbb{R} \times \mathbb{R}^m \rightarrow \mathbb{R}^m$ oppfyller Lipschitz betingelsen med hensyn på $\mathbf{x}$ i et område $(a, b) \times D$, hvor $D \subset \mathbb{R}^m$, hvis det eksisterer en konstant L slik at

$$\|\mathbf{f}(t, \mathbf{x}) - \mathbf{f}(t, \tilde{\mathbf{x}})\| \leq L \|\mathbf{x} - \tilde{\mathbf{x}}\|, \quad \text{for alle } t \in (a, b), \quad \mathbf{x}, \tilde{\mathbf{x}} \in D.$$

L kalles Lipschitz konstanten.

Det kan vises at funksjonen $\mathbf{f}$ oppfyller Lipschitz betingelsen hvis $\partial f_i / \partial x_j$, $i, j = 1, \dots, m$, er kontinuerlig og begrenset i området.

Teorem: Vi har initialverdiproblemet

$$\mathbf{x}' = \mathbf{f}(t, \mathbf{x}), \quad \mathbf{x}(t_0) = \mathbf{x}_0. \tag{1}$$

Hvis

1. $\mathbf{f}(t, \mathbf{x})$ er kontinuerlig i $(a, b) \times D$,
2. $\mathbf{f}(t, \mathbf{x})$ oppfyller Lipschitz betingelsen med hensyn på $\mathbf{x}$ i $(a, b) \times D$,

med gitte initialverdier $t_0 \in (a, b)$ og $\mathbf{x}_0 \in D$, da har (1) en og kun en løsning i $(a, b) \times D$.

2 Numerisk løsning av ODE

I denne seksjonen vil vi se på noen enkle metoder for løsning av initialverdiproblemer. Vi ser fortsatt på problemet (1) og vil anta at vi har funnet løsninger $\mathbf{x}_l \simeq \mathbf{x}(t_l)$, $l = 0, 1, \dots, n$, og vi ønsker å finne en approksimasjon $\mathbf{x}_{n+1} \simeq \mathbf{x}(t_{n+1})$, hvor $t_{n+1} = t_n + h$, og h er steglengden. Vi vil se på to ulike klasser for løsning av slike problem:

1. *Enstegsmetoder*: Kun $\mathbf{x}_n$ blir brukt til å finne approksimasjonen $\mathbf{x}_{n+1}$. Slike metoder krever typisk flere enn en funksjonsevaluering pr. steg. De kan skrives på formen

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h\Phi(t_n, \mathbf{x}_n; h)$$

2. *Flerstegsmetoder*: $\mathbf{x}_{n+1}$ approksimeres ved å bruke $\mathbf{x}_{n-k+1}$, for ulike verdier av $k \geq 0$.

2.1 Noen eksempler på enstegsmetoder

Anta vi kjenner $t_n, \mathbf{x}_n$. Den eksakte løsningen $\mathbf{x}(t_{n+1})$ til (1) er gitt ved

$$\mathbf{x}(t_n + h) = \mathbf{x}_n + \int_{t_n}^{t_{n+1}} \mathbf{x}'(\tau) d\tau = \mathbf{x}_n + \int_{t_n}^{t_{n+1}} \mathbf{f}(\tau, \mathbf{x}(\tau)) d\tau.$$

Vi trenger altså en approksimasjon til det siste integralet. Det enkleste er å bruke $\mathbf{f}(\tau, \mathbf{x}(\tau)) \simeq \mathbf{f}(t_n, \mathbf{x}_n)$, som gir Euler's metode:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h\mathbf{f}(t_n, \mathbf{x}_n).$$

Et annet alternativ er å approksimere integralet med trapesmetoden:

$$\int_{t_n}^{t_{n+1}} \mathbf{f}(\tau, \mathbf{x}(\tau)) d\tau \simeq \frac{h}{2}(\mathbf{f}(t_n, \mathbf{x}_n) + \mathbf{f}(t_{n+1}, \mathbf{x}_{n+1})).$$

Dette leder til et numerisk skjema:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \frac{h}{2}(\mathbf{f}(t_n, \mathbf{x}_n) + \mathbf{f}(t_{n+1}, \mathbf{x}_{n+1})). \quad (2)$$

For å finne $\mathbf{x}_{n+1}$ må vi her løse et (generelt) ikke-lineært system. Slike metoder kalles *implisitte metoder*, og er ofte mer kostbare pr. tidssteg enn tilsvarende *eksplisitte metoder*. Hovedgrunnen til å velge en implisitt metode er relatert til numerisk stabilitet, noe som blir spesielt relevant for løsning av såkalte *stive problem*.

En mulig modifikasjon av (2) er å approksimere $\mathbf{x}_{n+1}$ på høyresiden ved bruk av Euler's metode

$$\begin{aligned} \tilde{\mathbf{x}}_{n+1} &= \mathbf{x}_n + h\mathbf{f}(t_n, \mathbf{x}_n), \\ \mathbf{x}_{n+1} &= \mathbf{x}_n + \frac{h}{2}(\mathbf{f}(t_n, \mathbf{x}_n) + \mathbf{f}(t_{n+1}, \tilde{\mathbf{x}}_{n+1})). \end{aligned}$$

Denne metoden er eksplisitt og kalles *Heun's metode*.

3 Runge-Kutta metoder

Euler's metode og Heun's metode er begge eksempler på *eksplisitte Runge-Kutta metoder* (ERK). Slike skjema er gitt ved

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(t_n, \mathbf{x}_n), \\ \mathbf{k}_2 &= \mathbf{f}(t_n + c_2 h, \mathbf{x}_n + h a_{21} \mathbf{k}_1), \\ \mathbf{k}_3 &= \mathbf{f}(t_n + c_3 h, \mathbf{x}_n + h(a_{31} \mathbf{k}_1 + a_{32} \mathbf{k}_2)), \\ &\vdots \\ \mathbf{k}_s &= \mathbf{f}(t_n + c_s h, \mathbf{x}_n + h \sum_{j=1}^{s-1} a_{sj} \mathbf{k}_j), \\ \mathbf{x}_{n+1} &= \mathbf{x}_n + h \sum_{i=1}^s b_i \mathbf{k}_i, \end{aligned}$$

hvor c_i, a_{ij} og b_i er koeffisienter som definerer metoden. Vi krever alltid $c_i = \sum_{j=1}^s a_{ij}$. s kalles her antall *nivåer*.

Heun's metode er en to-nivå RK-metode, gitt ved

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(t_n, \mathbf{x}_n), \\ \mathbf{k}_2 &= \mathbf{f}(t_n + h, \mathbf{x}_n + h \mathbf{k}_1), \\ \mathbf{x}_{n+1} &= \mathbf{x}_n + \frac{h}{2}(\mathbf{k}_1 + \mathbf{k}_2). \end{aligned}$$

Også implisitte metoder, som trapesmetoden

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \frac{h}{2}(\mathbf{f}(t_n, \mathbf{x}_n) + \mathbf{f}(t_{n+1}, \mathbf{x}_{n+1})).$$

kan skrives på tilsvarende form

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(t_n, \mathbf{x}_n), \\ \mathbf{k}_2 &= \mathbf{f}(t_n + h, \mathbf{x}_n + \frac{h}{2}(\mathbf{k}_1 + \mathbf{k}_2)), \\ \mathbf{x}_{n+1} &= \mathbf{x}_n + \frac{h}{2}(\mathbf{k}_1 + \mathbf{k}_2). \end{aligned}$$

Her må vi altså løse et (ikke-lineært) system for å finne $\mathbf{k}_2$.

Definisjon: En s -nivå Runge-Kutta metode er gitt ved

$$\begin{aligned} \mathbf{k}_i &= \mathbf{f}(t_n + c_i h, \mathbf{x}_n + h \sum_{j=1}^s a_{ij} \mathbf{k}_j), \quad i = 1, 2, \dots, s, \\ \mathbf{x}_{n+1} &= \mathbf{x}_n + h \sum_{i=1}^s b_i \mathbf{k}_i. \end{aligned}$$

Denne metoden defineres av koeffisientene, som ofte blir presentert i en Butcher-tabell

c_1	a_{11}	a_{12}	$\dots$	a_{1s}
c_2	a_{21}	a_{22}	$\dots$	a_{2s}
$\vdots$	$\vdots$			$\vdots$
c_s	a_{s1}	a_{s2}	$\dots$	a_{ss}
	b_1	b_2	$\dots$	b_s

hvor $c_i = \sum_{j=1}^s a_{ij}$, $i = 1, \dots, s$. Denne metoden er eksplisitt hvis $a_{ij} = 0$ når $j \geq i$

Eksempel: Butcher-tabeller for the tre metodene vi hittil har sett på

0	0
1	1

Euler

0	0	0
1	1	0
	$\frac{1}{2}$	$\frac{1}{2}$

Heun

0	0	0
1	$\frac{1}{2}$	$\frac{1}{2}$
	$\frac{1}{2}$	$\frac{1}{2}$

Trapez

Når metoden er eksplisitt er vanligvis nullene på og over diagonalen ikke inkludert. Vi konkluderer denne seksjonen med å presentere den mest kjente RK-metoden; 4.ordens eksplisitt Runge-Kutta metode (Kutta - 1901):

$$\begin{aligned}
\mathbf{k}_1 &= \mathbf{f}(t_n, \mathbf{x}_n), \\
\mathbf{k}_2 &= \mathbf{f}(t_n + \frac{h}{2}, \mathbf{x}_n + \frac{h}{2}\mathbf{k}_1), \\
\mathbf{k}_3 &= \mathbf{f}(t_n + \frac{h}{2}, \mathbf{x}_n + \frac{h}{2}\mathbf{k}_2), \\
\mathbf{k}_4 &= \mathbf{f}(t_n + h, \mathbf{x}_n + h\mathbf{k}_3), \\
\mathbf{x}_{n+1} &= \mathbf{x}_n + \frac{h}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4),
\end{aligned}$$

0	$\frac{1}{2}$	0	0
$\frac{1}{2}$	0	$\frac{1}{2}$	0
$\frac{1}{2}$	0	0	1
1	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$

3.1 Ordensbetingelser for Runge-Kutta metoder

Vi har følgende teorem:

Teorem: La problemet

$$\mathbf{x}' = \mathbf{f}(t, \mathbf{x}), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad t_0 \leq t \leq t_{\text{end}}$$

bli løst av en enstegsmetode

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h\Phi(t_n, \mathbf{x}_n; h)$$

med steglengde $h = (t_{\text{end}} - t_0)/N$, hvor N er antall steg. Hvis

1. $\Phi(t_n, \mathbf{x}_n; h)$ er Lipschitz i $\mathbf{x}$ og
2. lokal trunkeringsfeil er $\mathbf{d}_{n+1} = \mathcal{O}(h^{p+1})$,

har metoden orden p , hvilket betyr at den globale feilen er gitt ved

$$\mathbf{e} = \mathbf{x}(t_{\text{end}}) - \mathbf{x}_N = \mathcal{O}(h^p).$$

En RK-metode er en enstegsmetode med inkrementfunksjon $\Phi(t_n, \mathbf{x}_n; h) = \sum_{i=1}^s b_i \mathbf{k}_i$. Det kan vises at Φ er Lipschitz i $\mathbf{x}$ hvis $\mathbf{f}$ er Lipschitz og $h \leq h_{\max}$, hvor $h_{\max}$ er maksimal steglengde. For å finne lokal trunkeringsfeil sammenligner vi Taylor-utviklingen til eksakt og numerisk løsning. Den lokale trunkeringsfeilen er $\mathcal{O}(h^{p+1})$ hvis de er like for alle ledd h^q , $q \leq p$. Prinsipielt er dette enkelt, men i praksis blir dette fort komplisert. Ordensbetingelsene opp til 4 er gitt ved

Orden	Betingelse
1	$\sum b_i = 1$
2	$\sum b_i c_i = 1/2$
3	$\sum b_i c_i^2 = 1/3$ $\sum b_i a_{ij} c_j = 1/6$
4	$\sum b_i c_i^3 = 1/4$ $\sum b_i c_i a_{ij} c_j = 1/8$ $\sum b_i a_{ij} c_j^2 = 1/12$ $\sum b_i a_{ij} a_{jk} c_k = 1/24$

4 Flerstegsmetoder

En alternativ strategi for løsning av problemer på formen (1) er ved bruk av flerstegsmetoder.

En lineær flerstegsmetode med s steg er på formen

$$\sum_{j=0}^s \alpha_j \mathbf{x}_{k+j} = h \sum_{j=0}^s \beta_j \mathbf{f}(t_{k+j}, \mathbf{x}_{k+j}), \quad (3)$$

hvor $\alpha_s \neq 0$. Hvis $\beta_s \neq 0$ inkluderer høyresiden $\mathbf{x}_{k+s}$ og metoden er *implisitt*; hvis $\beta_s = 0$ er metoden *eksplisitt*.

En fordel med slike metoder er at funksjonsevalueringer fra tidligere tidssteg gjenbrukes, noe som ikke var tilfelle for enstegsmetodene fra forrige avsnitt. En ulempe er at det blir mer komplisert å endre steglengden for ulike tidssteg i forhold til enstegsmetoder.

4.1 Adams-Bashforth metoder

Adams-Bashforth metoder er en klasse eksplisitte flerstegsmetoder (3) med $\alpha_0 = \alpha_1 = \dots = \alpha_{s-2} = 0$, $\alpha_{s-1} = -1$, $\alpha_s = 1$ og $\beta_s = 0$. Vi fant tidligere

$$\mathbf{x}(t_n + h) = \mathbf{x}_n + \int_{t_n}^{t_{n+1}} \mathbf{f}(\tau, \mathbf{x}(\tau)) d\tau. \quad (4)$$

Ved å approksimere integranden i det siste integralet som en konstant over (t_n, t_{n+1}) finner vi (se Figur 1):

AB1:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h \mathbf{f}(t_n, \mathbf{x}_n).$$

Dette tilsvarer Euler's metode. Hvis vi derimot bruker $\mathbf{f}(t_{n-1}, \mathbf{x}_{n-1})$ og $\mathbf{f}(t_n, \mathbf{x}_n)$ til å approksimere $\mathbf{f}(\tau, \mathbf{x}(\tau))$ som en lineær funksjon over (t_n, t_{n+1}) finner vi (se Figur 1):

AB2:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h \left(\frac{3}{2} \mathbf{f}(t_n, \mathbf{x}_n) - \frac{1}{2} \mathbf{f}(t_{n-1}, \mathbf{x}_{n-1}) \right).$$

Tilsvarende, ved å bruke $\mathbf{f}(t_{n-2}, \mathbf{x}_{n-2})$, $\mathbf{f}(t_{n-1}, \mathbf{x}_{n-1})$ og $\mathbf{f}(t_n, \mathbf{x}_n)$ til å approksimere $\mathbf{f}(\tau, \mathbf{x}(\tau))$ som en kvadratisk funksjon over (t_n, t_{n+1}) finner vi (se Figur 1):

AB3:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h \left(\frac{23}{12} \mathbf{f}(t_n, \mathbf{x}_n) - \frac{4}{3} \mathbf{f}(t_{n-1}, \mathbf{x}_{n-1}) + \frac{5}{12} \mathbf{f}(t_{n-2}, \mathbf{x}_{n-2}) \right).$$

Det kan vises at global feil for en Adams-Bashforth metode med s steg er $\mathcal{O}(h^s)$.

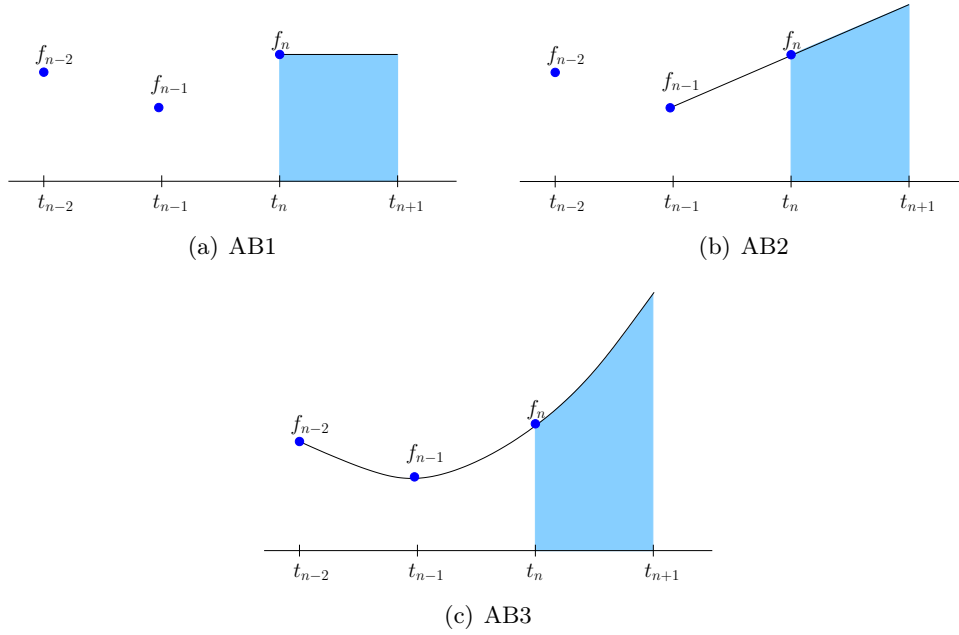


Figure 1: Approximasjon av $\mathbf{f}(\tau, \mathbf{x}(\tau))$ for Adams-Bashforth metoder.

4.2 Adams-Moulton metoder

Adams-Moulton metoder er en klasse implisitte flerstegsmetoder. Disse utledes også ved å approksimere integranden i (4), men nå brukes også $\mathbf{f}(t_{n+1}, \mathbf{x}_{n+1})$ i denne approksimasjonen, noe som gjør at $\mathbf{f}(\tau, \mathbf{x}(\tau))$ approksimeres ved interpolasjon og ikke ekstrapolasjon som AB-skjemaene. Ved å bruke $\mathbf{f}(t_n, \mathbf{x}_n)$ og $\mathbf{f}(t_{n+1}, \mathbf{x}_{n+1})$ til å approksimere $\mathbf{f}(\tau, \mathbf{x}(\tau))$ som en lineær funksjon over (t_n, t_{n+1}) finner vi (se Figur 2):

AM1:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h \left(\frac{1}{2} \mathbf{f}(t_{n+1}, \mathbf{x}_{n+1}) + \frac{1}{2} \mathbf{f}(t_n, \mathbf{x}_n) \right).$$

Tilsvarende, ved å bruke $\mathbf{f}(t_{n-1}, \mathbf{x}_{n-1})$, $\mathbf{f}(t_n, \mathbf{x}_n)$ og $\mathbf{f}(t_{n+1}, \mathbf{x}_{n+1})$ til å approksimere $\mathbf{f}(\tau, \mathbf{x}(\tau))$ som en kvadratisk funksjon over (t_n, t_{n+1}) finner vi (se Figur 2):

AM2:

$$\mathbf{x}_{n+1} = \mathbf{x}_n + h \left(\frac{5}{12} \mathbf{f}(t_{n+1}, \mathbf{x}_{n+1}) + \frac{2}{3} \mathbf{f}(t_n, \mathbf{x}_n) - \frac{1}{12} \mathbf{f}(t_{n-1}, \mathbf{x}_{n-1}) \right).$$

Global feil for en Adams-Moulton metode med s steg er $\mathcal{O}(h^{s+1})$.

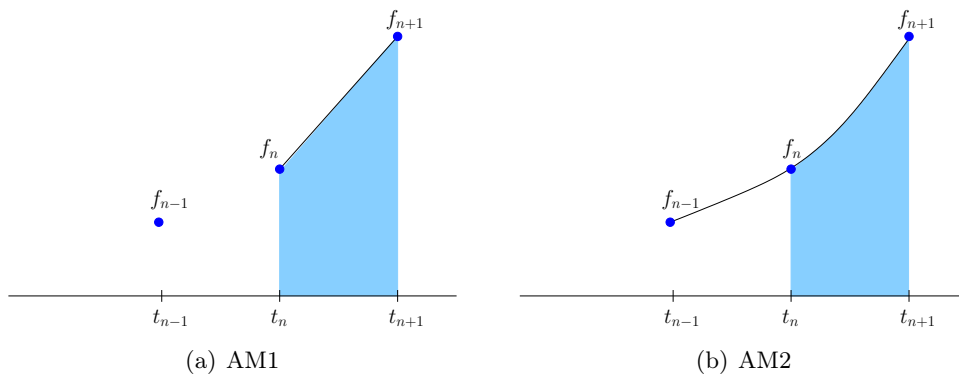


Figure 2: Apprøksimasjon av $\mathbf{f}(\tau, \mathbf{x}(\tau))$ for Adams-Moulton metoder.