Metode Numerik

Imam Fachruddin

Departemen Fisika, Universitas Indonesia

Untuk dipakai dalam kuliah Analisis Numerik

Dapat diunduh dari http://staff.fisika.ui.ac.id/imamf/

Metode Numerik

Imam Fachruddin

Departemen Fisika, Universitas Indonesia

Daftar Pustaka:

- P. L. DeVries, A First Course in Computational Physics (John Wiley & Sons, Inc., New York, 1994)
- W. H. Press, et. al., Numerical Recipes in Fortran 77, 2nd Ed. (Cambridge University Press, New York, 1992)

  (online / free download: http://www.nrbook.com/a/bookfpdf.php)
- R. H. Landau & M. J. Páez, Computational Physics: Problem Solving with Computers (John Wiley & Sons, Inc., New York, 1997)
- S. E. Koonin, Computational Physics (Addison-Wesley Publishing Co., Inc., Redwood City, 1986)

Isi

Akar Fungsi

$$f(x) = 0 \longrightarrow \boxed{x = ?}$$

akar fungsi f(x)

Contoh: $x - \frac{1}{x} = 0 \longrightarrow x^2 = 1 \longrightarrow x = 1 \text{ dan } -1$

$3x^2 = 6 - 7x \longrightarrow 3x^2 + 7x - 6 = (3x - 2)(x + 3) = 0$

$\downarrow$

$x = 2/3 \text{ dan } -3$

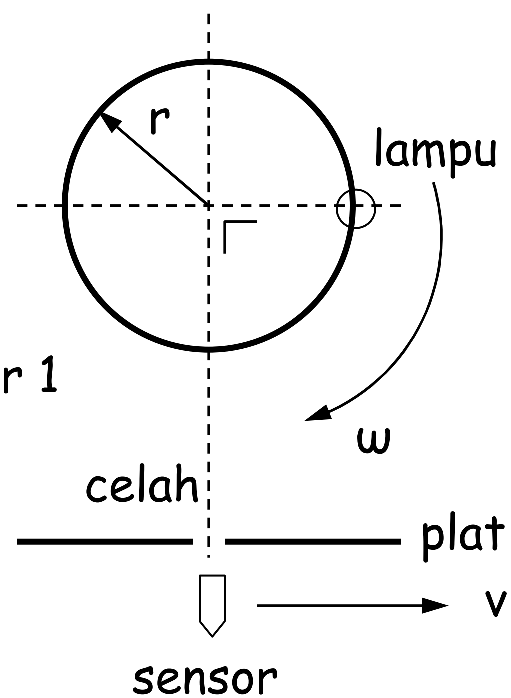
Pada dua contoh di atas akar fungsi dapat dicari secara analitik.

Secara umum, tidak selalu begitu keadaannya.

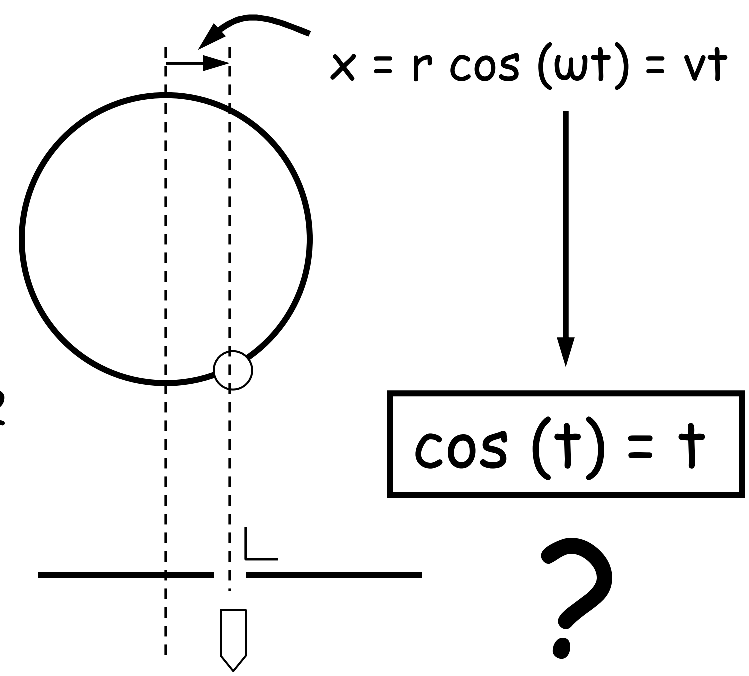
Problem:

Sebuah lampu dipasang di pinggir sebuah piringan berjari-jari 10 cm. Sebuah plat bercelah sempit diletakkan di dekat piringan itu. Tepat di belakang celah itu dipasang sebuah sensor cahaya yang menghadap tegak lurus ke celah. Piringan diputar konstan 1 rad/s dan plat beserta sensor digeser lurus konstan 10 cm/s. Saat ini posisi celah dan lampu seperti pada gambar 1. Kapan sensor cahaya menerima cahaya terbanyak?

Sensor menerima cahaya terbanyak pada saat posisi lampu dan celah membentuk garis tegak lurus terhadap plat, seperti pada gambar 2.

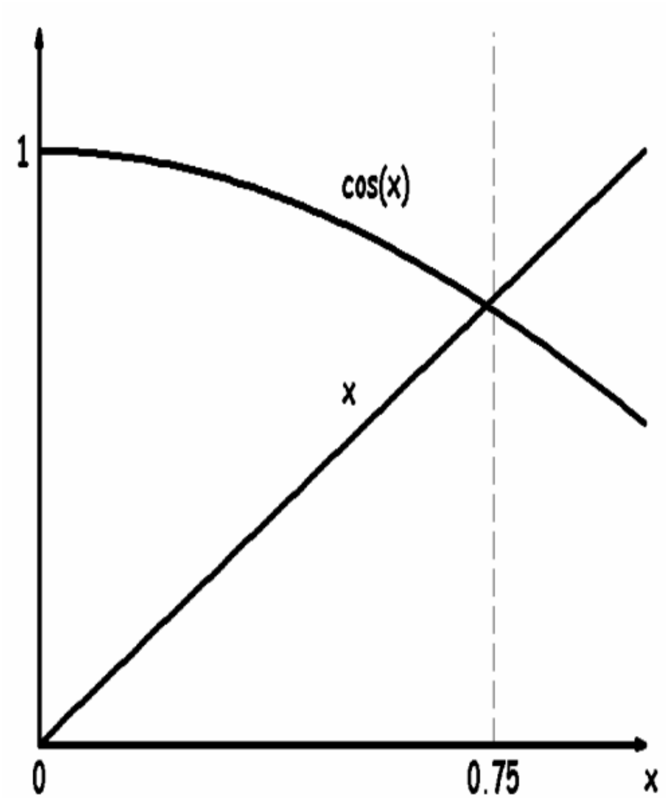


gambar 1



gambar 2

Plot cos(x) dan x:



Grafik ini menunjukkan bahwa cos(x) = x pada x sedikit kurang dari 0.75.

Bisakah lebih akurat lagi?

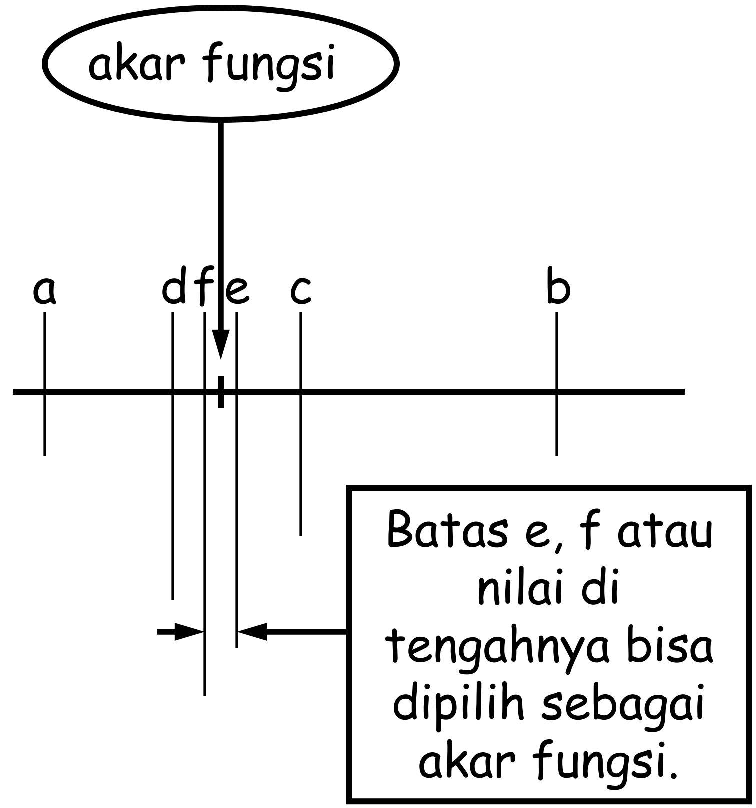
Cari secara numerik akar fungsi dari
$f(x) = \cos(x) - x$

Bisection

Prinsip: Kurung akar fungsi di antara dua batas, lalu paruh batas itu terus menerus sampai batas itu sedemikian sempit dan dengan demikian lokasi akar fungsi diketahui dengan keakuratan tertentu.

Langkah:

1. Perkirakan akar fungsi (bisa dengan cara memplot fungsi).
2. Tentukan batas awal yang mengurung akar fungsi.
3. Belah dua daerah berisi akar fungsi itu.
4. Tentukan daerah yang berisi akar fungsi.
5. Ulangi langkah 3 dan 4 sampai dianggap cukup.
6. Tentukan akar fungsi.

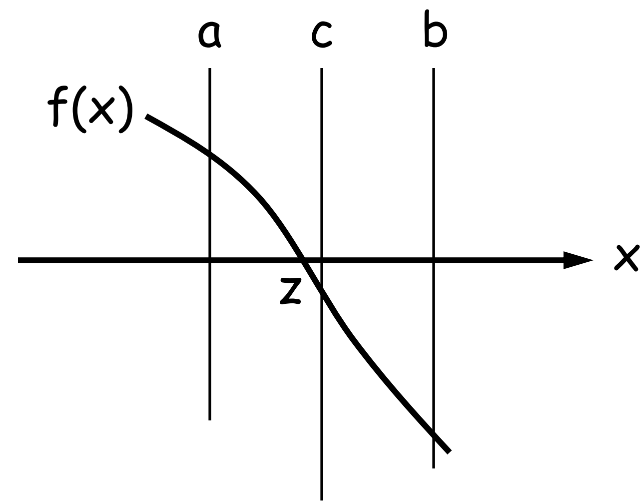


- Menentukan daerah yang berisi akar fungsi:

  Jika z merupakan akar fungsi, maka f(x < z) dan f(x > z) saling berbeda tanda.

  f(a)*f(c) negatif, berarti di antara a & c ada akar fungsi.

  f(b)*f(c) positif, berarti di antara b & c tidak ada akar fungsi



- Menentukan kapan proses pencarian akar fungsi berhenti:

  Proses pencarian akar fungsi dihentikan setelah keakuratan yang diinginkan dicapai, yang dapat diketahui dari kesalahan relatif semu.

$$\text{kesalahan relatif semu} = \left| \frac{\text{perkiraan sebelum - perkiraan berikut}}{\text{perkiraan berikut}} \right|$$

Kesalahan

$$\text{kesalahan mutlak} = |\text{perkiraan} - \text{nilai sebenarnya}|$$

$$\text{kesalahan relatif} = \left|\frac{\text{perkiraan} - \text{nilai sebenarnya}}{\text{nilai sebenarnya}}\right|$$

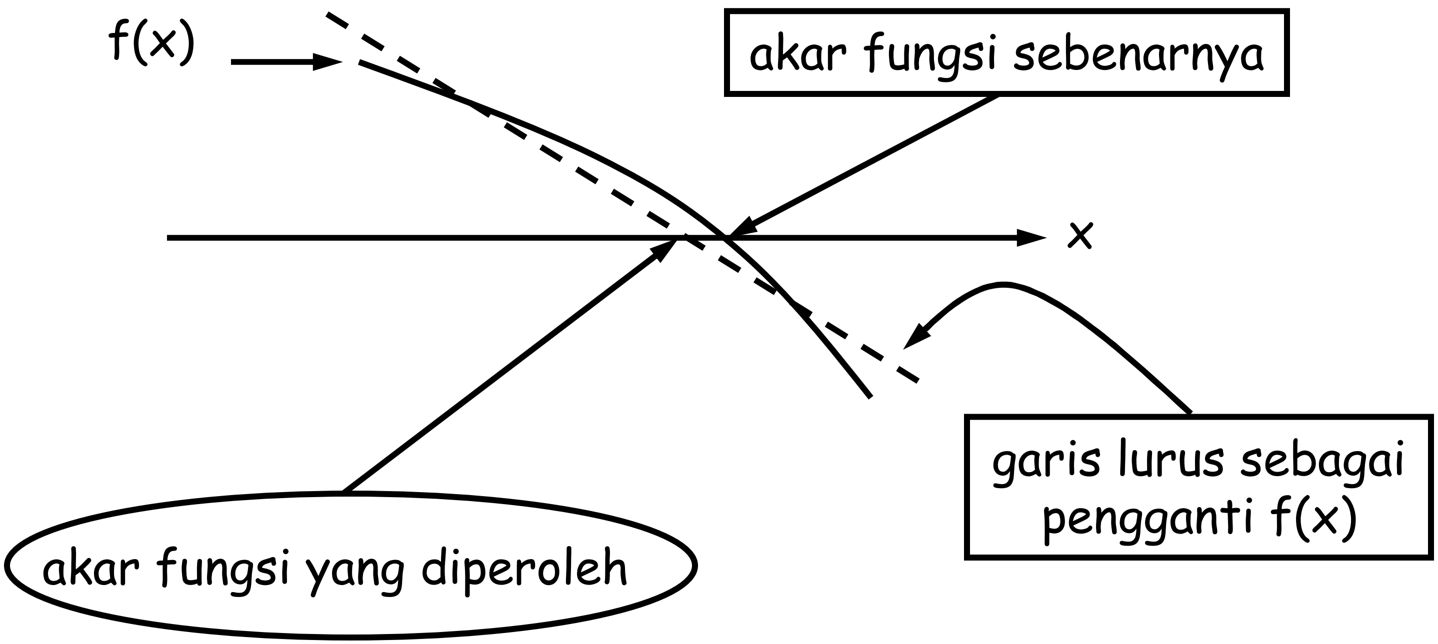
Dalam perhitungan numerik, nilai sebenarnya justru sering tidak diketahui, yang didapat hanya perkiraan terbaik. Karena perkiraan langkah berikut dianggap lebih akurat, yaitu lebih mendekati nilai sebenarnya, maka kesalahan yang dihitung yaitu:

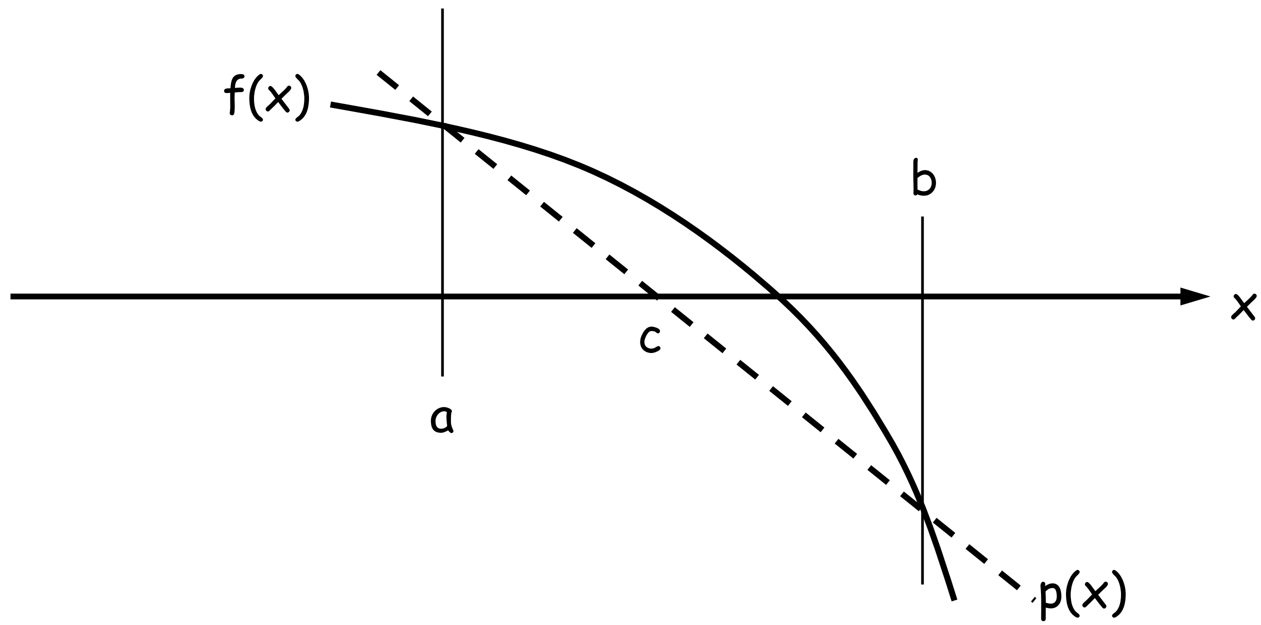
$$\text{kesalahan mutlak semu} = |\text{perkiraan sebelum} - \text{perkiraan berikut}|$$

$$\text{kesalahan relatif semu} = \left|\frac{\text{perkiraan sebelum} - \text{perkiraan berikut}}{\text{perkiraan berikut}}\right|$$

False Position

Prinsip: Di sekitar akar fungsi yang diperkirakan, anggap fungsi merupakan garis lurus. Titik tempat garis lurus itu memotong garis nol ditentukan sebagai akar fungsi.



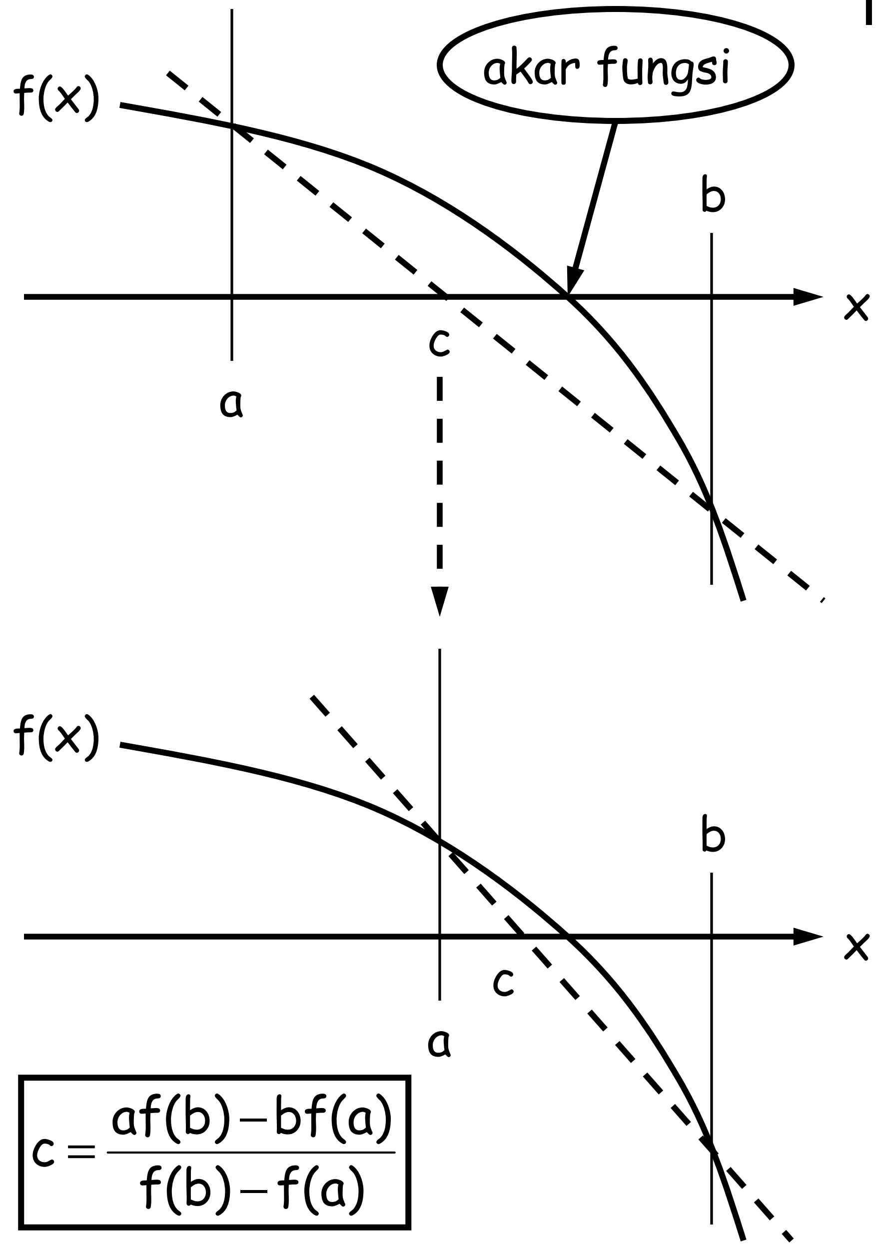


Diperoleh: $$p(x) = \left(\frac{x-b}{a-b}\right) f(a) + \left(\frac{x-a}{b-a}\right) f(b)$$

$$p(c) = 0 \longrightarrow \boxed{c = \frac{af(b) - bf(a)}{f(b) - f(a)}}$$

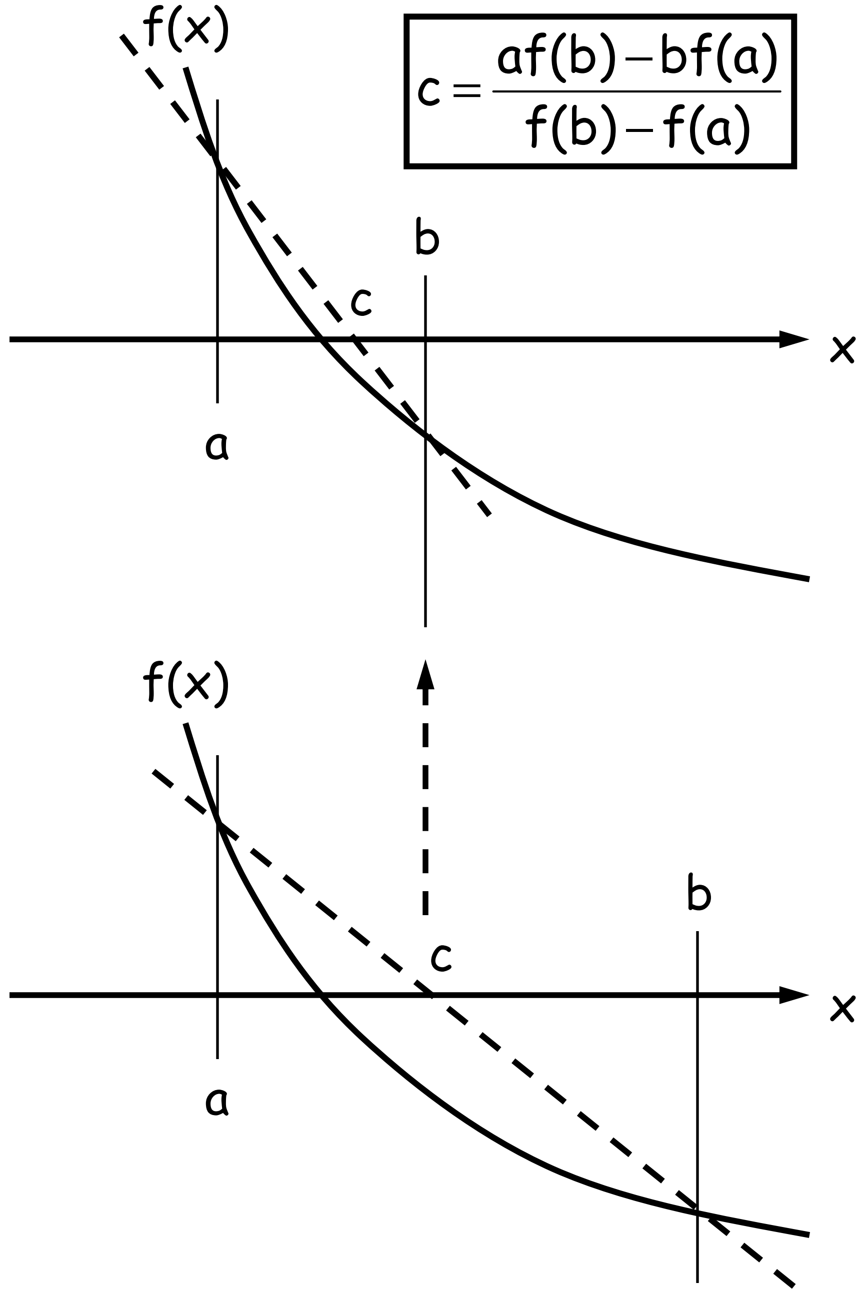
Langkah:

1. Perkirakan akar fungsi (bisa dengan cara memplot fungsi).
2. Tentukan batas awal yang mengurung akar fungsi.
3. Tarik garis lurus penghubung nilai fungsi pada kedua batas, lalu cari titik potongnya dengan garis nol.
4. Geser salah satu batas ke titik potong itu, sementara batas lain tidak berubah. Ulangi langkah 3.
5. Ulangi langkah 4 sampai dianggap cukup.
6. Titik potong garis nol dan garis lurus yang terakhir dinyatakan sebagai akar fungsi.



$$\boxed{c = \frac{af(b) - bf(a)}{f(b) - f(a)}}$$

$$c = \frac{af(b) - bf(a)}{f(b) - f(a)}$$



Metode false position juga menggunakan dua batas seperti metode bisection. Namun, berbeda dari metode bisection, pada metoda false position hanya satu batas yang berubah.

Pada contoh sebelum ini, batas a berubah sementara batas b tetap. Pada contoh berikut terjadi sebaliknya.

Menghitung akar fungsi dengan metode false position, menggunakan a dan b sebagai batas awal:

- jika batas a tetap, batas b berubah:

$$\boxed{x_{i+1} = \frac{af(x_i) - x_i f(a)}{f(x_i) - f(a)} \qquad (i = 0,1,2,...; \;\; x_0 = b)}$$

- jika batas b tetap, batas a berubah:

$$\boxed{x_{i+1} = \frac{bf(x_i) - x_i f(b)}{f(x_i) - f(b)} \qquad (i = 0,1,2,...; \;\; x_0 = a)}$$

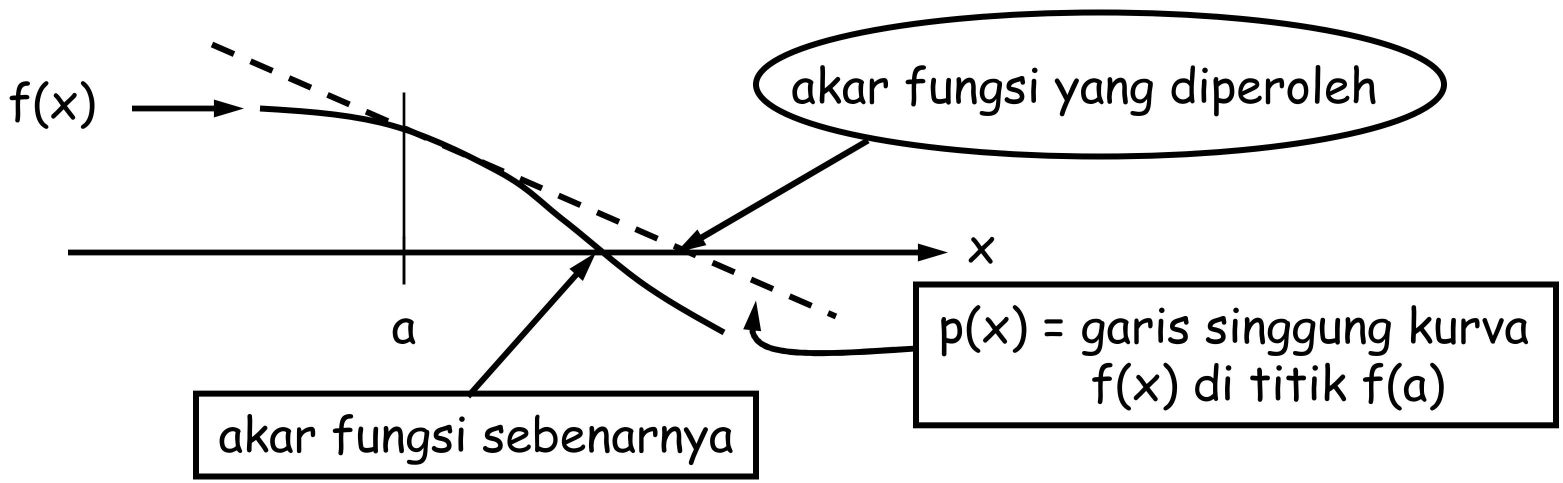
- kesalahan relatif semu:

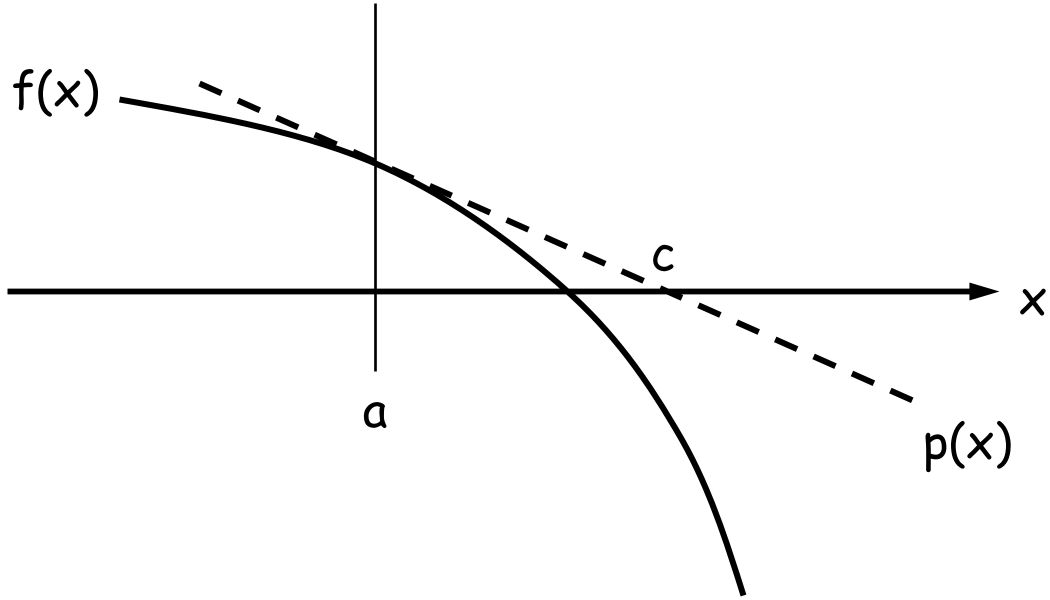
$$\boxed{\Delta_{rel} = \left| \frac{x_i - x_{i+1}}{x_{i+1}} \right|}$$

Penghitungan dihentikan jika kesalahan relatif semu sudah mencapai / melampaui batas yang diinginkan.

Newton-Raphson

Prinsip: Buat garis singgung kurva f(x) di titik di sekitar akar fungsi. Titik tempat garis singgung itu memotong garis nol ditentukan sebagai akar fungsi.





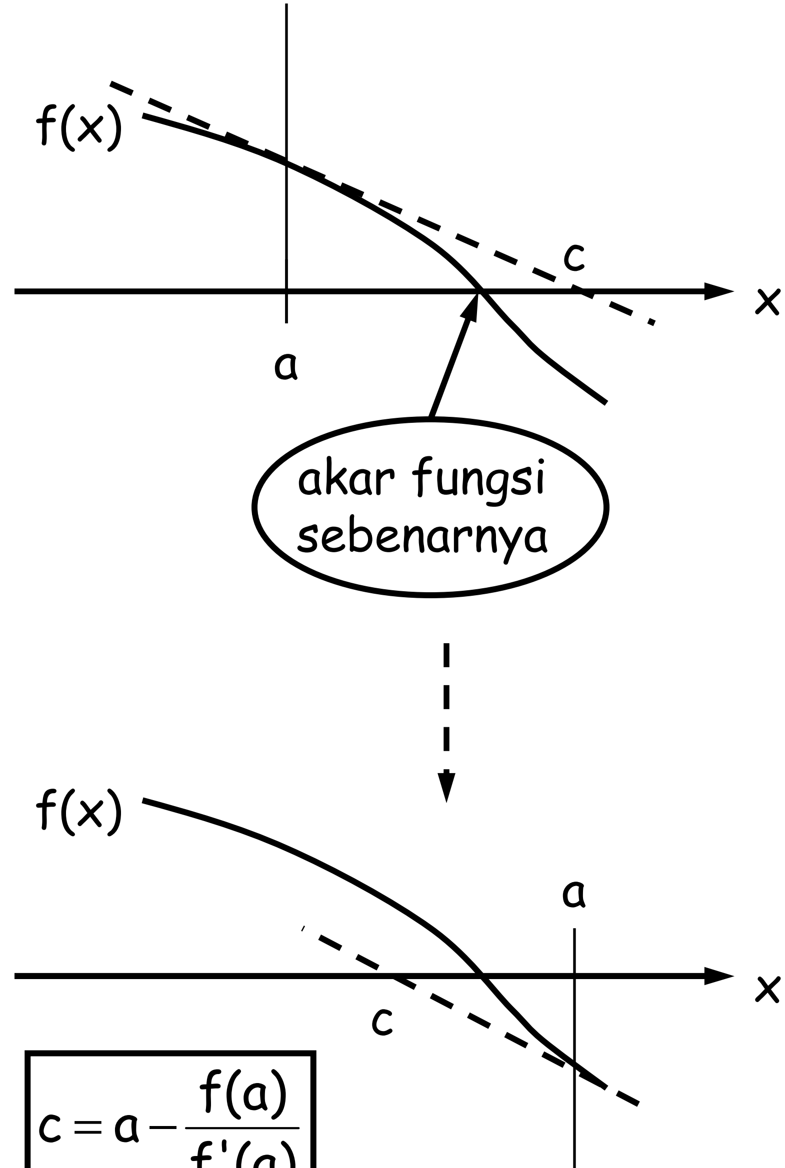
Diperoleh: $$p(x) = f(a) + (x-a)f'(a)$$

(f'(a) turunan pertama f(x) pada x = a)

$$p(c) = 0 \longrightarrow \boxed{c = a - \frac{f(a)}{f'(a)}}$$

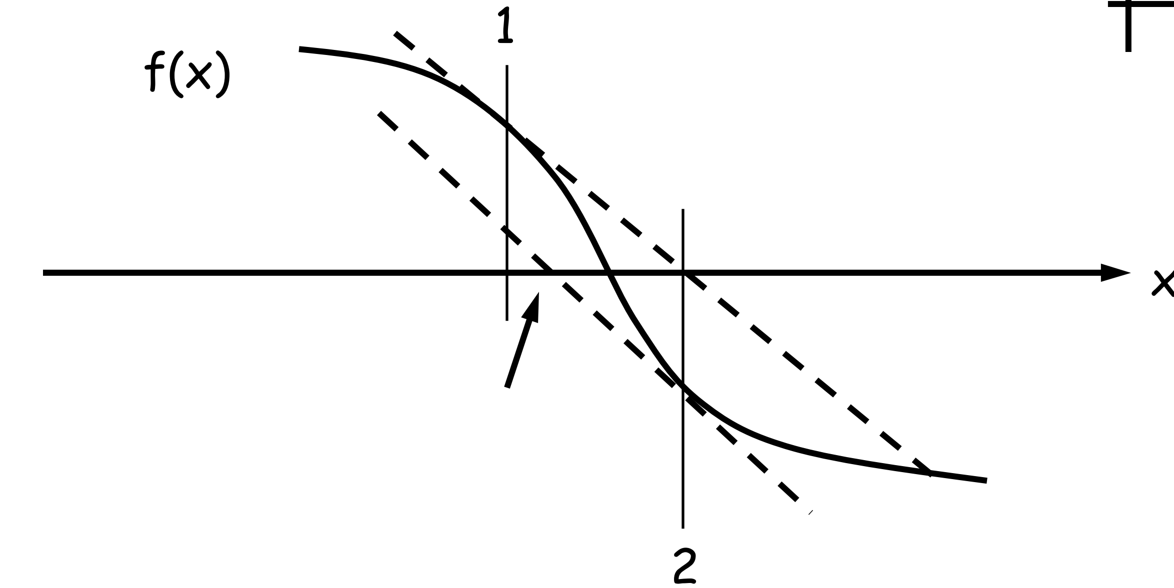
Langkah:

1. Perkirakan akar fungsi.
2. Buat garis singgung pada titik sesuai akar fungsi yang diperkirakan itu, lalu cari titik potongnya dengan garis nol.
3. Titik potong itu merupakan perkiraan akar fungsi baru.
4. Ulangi langkah 2 dan 3 sampai dianggap cukup.
5. Titik potong garis nol dan garis singgung kurva yang terakhir dinyatakan sebagai akar fungsi.

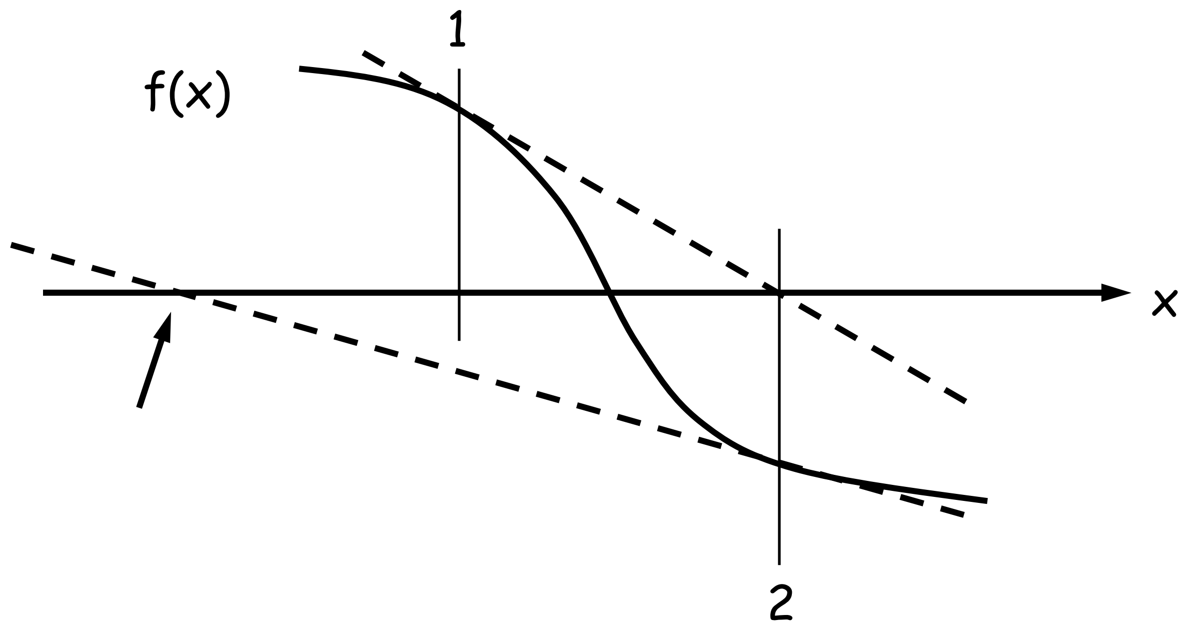


$$\boxed{c = a - \frac{f(a)}{f'(a)}}$$

Contoh perkiraan akar fungsi awal yang baik
→ perkiraan akar fungsi makin mendekati akar fungsi sebenarnya.



Contoh perkiraan akar fungsi awal yang buruk
→ perkiraan akar fungsi makin menjauhi akar fungsi sebenarnya.



Menghitung akar fungsi dengan metode Newton-Raphson:

$$\boxed{x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \qquad (i = 0,1,2,...;\ x_0 = a)}$$

kesalahan relatif semu:

$$\boxed{\Delta_{rel} = \left| \frac{x_i - x_{i+1}}{x_{i+1}} \right|}$$

Penghitungan dihentikan jika kesalahan relatif semu sudah mencapai / melampaui batas yang diinginkan.

Secant

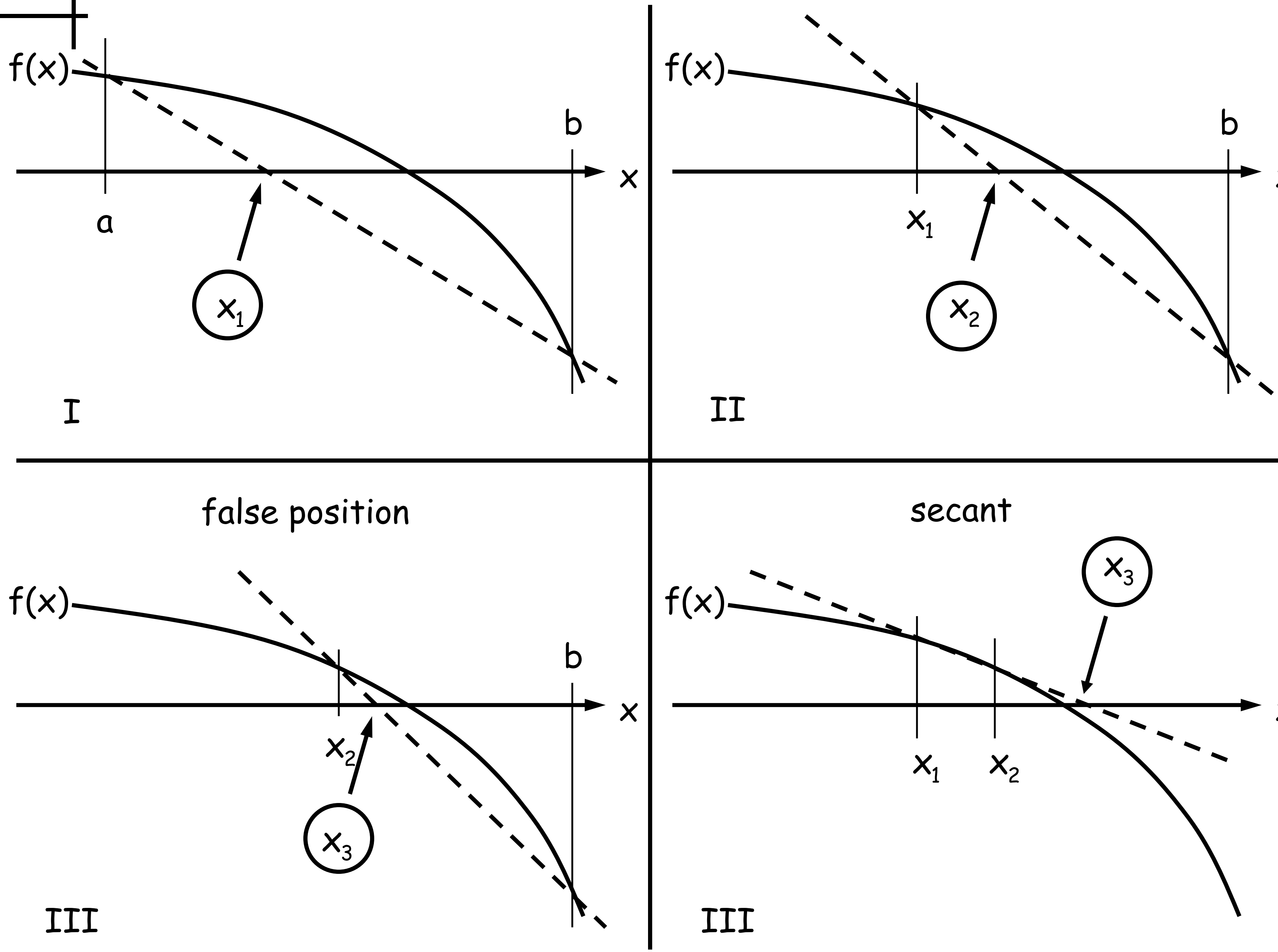
Kembali ke metode False Position, untuk contoh batas b tetap, akar fungsi dicari sebagai berikut:

$$x_1 = \frac{bf(x_0) - x_0 f(b)}{f(x_0) - f(b)} \longrightarrow x_2 = \frac{bf(x_1) - x_1 f(b)}{f(x_1) - f(b)} \longrightarrow x_3 = \frac{bf(x_2) - x_2 f(b)}{f(x_2) - f(b)} \quad \ldots\ldots$$

Pada metode Secant, batas tidak dijaga tetap, melainkan berubah. Akar fungsi dicari sebagai berikut:

$$x_1 = \frac{bf(x_0) - x_0 f(b)}{f(x_0) - f(b)} \longrightarrow x_2 = \frac{bf(x_1) - x_1 f(b)}{f(x_1) - f(b)} \longrightarrow x_3 = \frac{x_1 f(x_2) - x_2 f(x_1)}{f(x_2) - f(x_1)} \quad \ldots\ldots$$

Jadi, mulai dari i = 3, akar fungsi dihitung dengan: $\boxed{x_i = \frac{x_{i-2} f(x_{i-1}) - x_{i-1} f(x_{i-2})}{f(x_{i-1}) - f(x_{i-2})}}$


f(x)
b
x
a
x1
I
f(x)
b
x
x1
x2
II
false position
f(x)
b
x
x2
x3
III
secant
x3
f(x)
x
x1
x2
III

Akar fungsi pada metode Secant untuk i = 1, 2 bisa dihitung dengan metode yang lain atau ditebak. Mulai i = 3, akar fungsi dihitung dengan rumus:

$$x_i = \frac{x_{i-2} f(x_{i-1}) - x_{i-1} f(x_{i-2})}{f(x_{i-1}) - f(x_{i-2})} \longrightarrow \boxed{x_i = x_{i-1} - \left(\frac{f(x_{i-1}) - f(x_{i-2})}{x_{i-1} - x_{i-2}} \right)^{-1} f(x_{i-1})}$$

Yang menarik, jika i makin besar, maka beda antar dua akar fungsi yang berturutan semakin kecil, sehingga

$$\frac{f(x_{i-1}) - f(x_{i-2})}{x_{i-1} - x_{i-2}} \cong \frac{df(x_{i-1})}{dx_{i-1}} = f'(x_{i-1}) \longrightarrow x_i \cong x_{i-1} - \frac{f(x_{i-1})}{f'(x_{i-1})}$$

Dengan begitu, metode Secant menyerupai metode Newton-Raphson. Jika turunan fungsi f(x) sulit diperoleh / dihitung, maka metode Secant menjadi alternatif yang baik bagi metode Newton-Raphson.

Kesalahan relatif semu dihitung sama seperti pada metode False Position atau Newton-Raphson.

Kecepatan Konvergensi

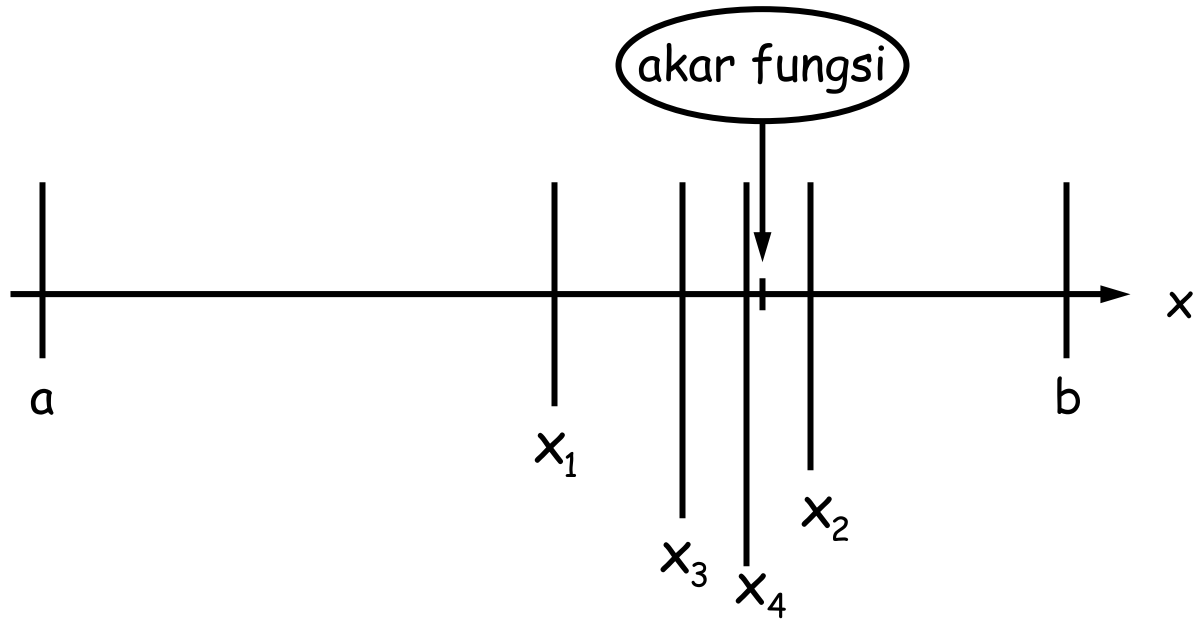
Pencarian akar fungsi dimulai dengan perkiraan akar fungsi yang pertama, lalu diikuti oleh perkiraan berikutnya dan seterusnya sampai perkiraan yang terakhir, yang kemudian dinyatakan sebagai akar fungsi hasil perhitungan tersebut. Proses itu harus bersifat konvergen yaitu, selisih perkiraan sebelum dari yang setelahnya makin lama makin kecil. Setelah dianggap cukup, proses pencarian akar fungsi berhenti.

$$|x_2 - x_1| > |x_3 - x_2| > |x_4 - x_3| \ldots |x_n - x_{n-1}| \leq \varepsilon$$

(ε = bilangan kecil)

Kecepatan konvergensi sebuah proses yaitu, kecepatan proses itu untuk sampai pada hasil akhir.

Contoh pencarian akar fungsi dengan metode Bisection:



Jika $\varepsilon_i \equiv |x_{i+1} - x_i|$, maka dari gambar diperoleh:

$$\varepsilon_1 = |x_2 - x_1|,\ \varepsilon_2 = |x_3 - x_2|,\ \varepsilon_3 = |x_4 - x_3|$$

$$\varepsilon_2 = \tfrac{1}{2}\varepsilon_1,\ \varepsilon_3 = \tfrac{1}{2}\varepsilon_2$$

Kecepatan konvergensi bersifat linear: $\boxed{\varepsilon_{i+1} = \tfrac{1}{2}\varepsilon_i}$

Pada metode False Position, Newton-Raphson dan Secant akar fungsi dicari dengan rumus yang bentuknya serupa:

False Position: $$x_{i+1} = x_i - \left(\frac{f(x_i) - f(a)}{x_i - a}\right)^{-1} f(x_i) \quad \text{(atau a diganti b)}$$

Newton-Raphson: $$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

Secant: $$x_{i+1} = x_i - \left(\frac{f(x_i) - f(x_{i-1})}{x_i - x_{i-1}}\right)^{-1} f(x_i)$$

Mengingat dengan berjalannya proses pencarian akar fungsi rumus pada metode False Position dan terlebih lagi Secant semakin mendekati rumus pada metode Newton-Raphson, maka akan dibahas kecepatan konvergen pada metode Newton-Raphson.

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \longrightarrow \varepsilon_i \equiv x_i - x_{i+1} = \frac{f(x_i)}{f'(x_i)} \longrightarrow \varepsilon_{i+1} = \frac{f(x_{i+1})}{f'(x_{i+1})} = ?$$

ekspansi deret Taylor:

$$f(x_{i+1}) = f(x_i - \varepsilon_i) = f(x_i) - \varepsilon_i f'(x_i) + \tfrac{1}{2}\varepsilon_i^2 f''(x_i) - \ldots$$

$$f'(x_{i+1}) = f'(x_i - \varepsilon_i) = f'(x_i) - \varepsilon_i f''(x_i) + \ldots$$

$$\longrightarrow \quad \varepsilon_{i+1} = \frac{f(x_i) - \varepsilon_i f'(x_i) + \tfrac{1}{2}\varepsilon_i^2 f''(x_i) - \ldots}{f'(x_i) - \varepsilon_i f''(x_i) + \ldots}$$

$$\cong \frac{f(x_i) - \varepsilon_i f'(x_i) + \tfrac{1}{2}\varepsilon_i^2 f''(x_i)}{f'(x_i)}$$

$$\cong \frac{f''(x_i)}{2f'(x_i)}\varepsilon_i^2$$

Kecepatan konvergensi pada metode Newton-Raphson (kira-kira demikian juga False Position dan Secant) bersifat kurang lebih kuadratik:

$$\boxed{\varepsilon_{i+1} \cong \frac{f''(x_i)}{2f'(x_i)}\varepsilon_i^2}$$

Dengan begitu, metode metode Newton-Raphson, False Position dan Secant lebih cepat dari metode Bisection.

Contoh hasil pencarian akar fungsi untuk soal cos(x) = x:

metode	akar	f(akar)	jumlah langkah
Bisection	0.7390795	9.3692161E-06	12
False Position	0.7390851	-7.7470244E-09	3
Newton-Raphson	0.7390851	-7.7470244E-09	4
Secant	0.7390851	-7.7470244E-09	3

Keterangan:

- Pencarian akar berhenti jika kesalahan relatif semu sama atau kurang dari 1.0E-05.
- Batas awal kiri dan kanan untuk metode Bisection, False Position dan Secant 0.72 dan 0.75.
- Perkiraan akar fungsi pertama untuk metode Newton-Raphson 0.72.

Solusi Sistem Persamaan Linear

Sistem persamaan linear:

$$
\begin{array}{cccccc}
a_{11}x_1 & +a_{12}x_2 & +a_{13}x_3 & \cdots & +a_{1n}x_n & =b_1 \\
a_{21}x_1 & +a_{22}x_2 & +a_{23}x_3 & \cdots & +a_{2n}x_n & =b_2 \\
a_{31}x_1 & +a_{32}x_2 & +a_{33}x_3 & \cdots & +a_{3n}x_n & =b_3 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
a_{n1}x_1 & +a_{n2}x_2 & +a_{n3}x_3 & \cdots & +a_{nn}x_n & =b_n
\end{array}
$$

n buah persamaan dengan n buah unknown x_j

a_{ij} dan b_i diketahui

i, j = 1, 2, …, n

$x_j = ?$

Soal:

$$\begin{aligned} 2x - 3y + 2z &= -6 \quad &(1)\\ -x + 2y - 3z &= 2 \quad &(2)\\ x + y - z &= 0 \quad &(3) \end{aligned}$$

3 persamaan dan
3 unknown

Jawab:

$$\begin{aligned} 2x - 3y + 2z &= -6 \quad &(1)\\ 0.5y - 2z &= -1 \quad &(2)\\ 2.5y - 2z &= 3 \quad &(3) \end{aligned}$$

eliminasi x:
pers. (2) + 0.5 pers. (1)
pers. (3) – 0.5 pers. (1)

$$\begin{aligned} 2x - 3y + 2z &= -6 \quad &(1)\\ 0.5y - 2z &= -1 \quad &(2)\\ 8z &= 8 \quad &(3) \end{aligned}$$

eliminasi y:
pers. (3) – 5 pers. (2)

$$z = 1$$

$$y = \frac{-1 + 2z}{0.5} = 2$$

$$x = \frac{-6 + 3y - 2z}{2} = -1$$

substitusi mundur:
pers. (3) → mencari z
pers. (2) → mencari y
pers. (1) → mencari x

Dalam bentuk matriks:

Soal:

$$\begin{pmatrix} 2 & -3 & 2 \\ -1 & 2 & -3 \\ 1 & 1 & -1 \end{pmatrix}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -6 \\ 2 \\ 0 \end{pmatrix}$$

Jawab:

$$\begin{pmatrix} 2 & -3 & 2 \\ 0 & 0.5 & -2 \\ 0 & 2.5 & -2 \end{pmatrix}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -6 \\ -1 \\ 3 \end{pmatrix}$$

$$\begin{pmatrix} 2 & -3 & 2 \\ 0 & 0.5 & -2 \\ 0 & 0 & 8 \end{pmatrix}\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} -6 \\ -1 \\ 8 \end{pmatrix}$$

$$z = 1$$

$$y = \frac{-1 + 2z}{0.5} = 2$$

$$x = \frac{-6 + 3y - 2z}{2} = -1$$

Eliminasi Gauss

Metode Eliminasi Gauss mencari solusi sebuah sistem persamaan linear dengan cara seperti ditunjukkan pada contoh sebelum ini:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

$$a_{ij}^{(0)} = a_{ij}, \quad b_i^{(0)} = b_i$$

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} a_{kj}^{(k-1)}$$

$$b_i^{(k)} = b_i^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} b_k^{(k-1)}$$

$$(k = 1, \ldots, n-1; \; i = k+1, \ldots, n; \; j = k, \ldots, n)$$

$a_{ij}^{(m)}, b_i^{(m)} \equiv a_{ij}, b_i$ pada langkah ke m

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & a_{13}^{(0)} & \cdots & a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn}^{(n-1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(2)} \\ \vdots \\ b_n^{(n-1)} \end{pmatrix}$$

→ halaman berikut

Substitusi mundur:

$$\begin{pmatrix} a_{11}^{(0)} & a_{12}^{(0)} & a_{13}^{(0)} & \cdots & a_{1n}^{(0)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} \\ 0 & 0 & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn}^{(n-1)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(0)} \\ b_2^{(1)} \\ b_3^{(2)} \\ \vdots \\ b_n^{(n-1)} \end{pmatrix}$$

$$\downarrow$$

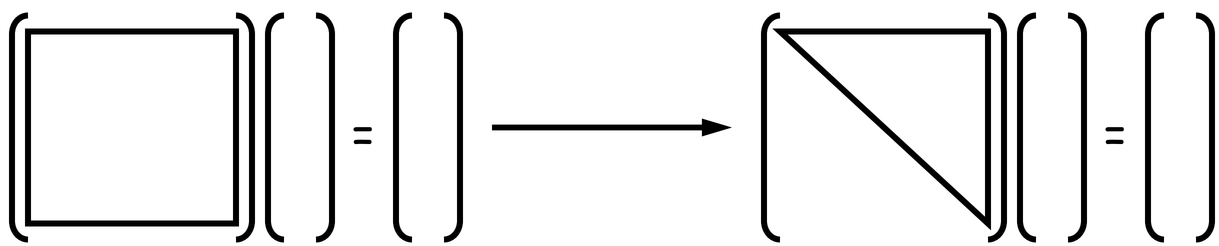
$$x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}$$

$$x_{n-j} = \frac{b_{n-j}^{(n-j-1)} - \sum_{k=n-j+1}^{n} a_{n-j,k}^{(n-j-1)} x_k}{a_{n-j,n-j}^{(n-j-1)}} \quad (j = 1, \ldots, n-1)$$

$$\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix} \begin{bmatrix} \\ X \\ \\ \end{bmatrix} = \begin{bmatrix} \\ B \\ \\ \end{bmatrix} \quad \text{atau} \quad AX = B$$

Jadi, metode Eliminasi Gauss terdiri dari dua tahap:

1. triangulasi: mengubah matriks A menjadi matriks segitiga (matriks B dengan begitu juga berubah)

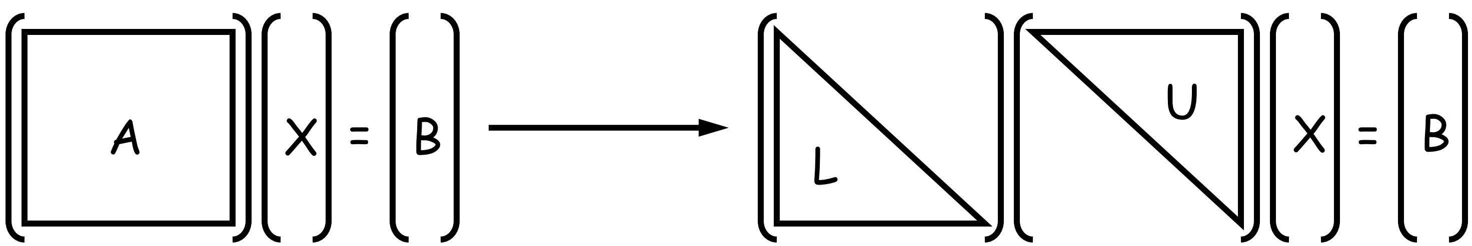


2. substitusi mundur: menghitung x mengikuti urutan terbalik, dari yang terakhir (x_n) sampai yang pertama (x_1)

LU Decomposition

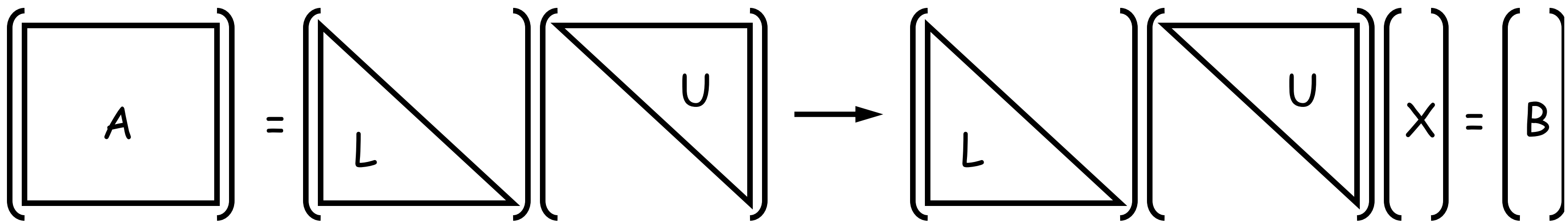
$$\begin{bmatrix} & A & \end{bmatrix}\begin{bmatrix} X \end{bmatrix} = \begin{bmatrix} B \end{bmatrix} \quad \text{atau} \quad AX = B$$

Pada metode LU Decomposition, matriks A ditulis ulang sebagai perkalian matriks L dan U (matriks A diurai menjadi matriks L dan U). Matriks L dan U merupakan matriks segitiga. Matriks B tidak berubah, karena matriks A tidak berubah, melainkan hanya ditulis ulang.

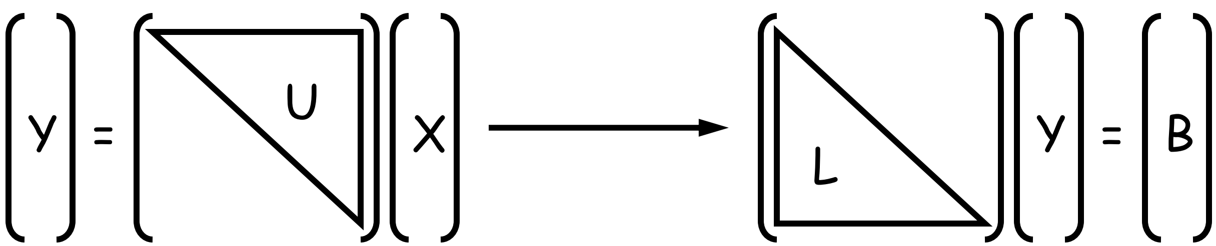


Langkah:

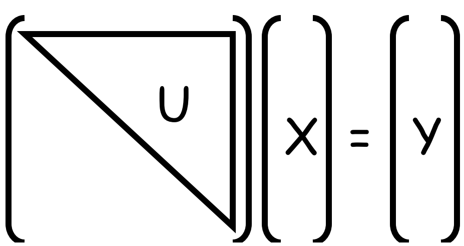
1. Cari matriks L dan U sehingga A = LU. Matriks B tetap.



2. Definisikan sebuah matriks kolom baru, misalnya Y, yaitu Y = UX, sehingga LY = B. Lalu hitung y dengan substitusi maju (mulai dari y_1 sampai y_n).



3. Hitung x dengan substitusi mundur (mulai dari x_n sampai x_1).



Jelas bahwa metode LU Decomposition pada prinsipnya sama dengan metode Eliminasi Gauss: matriks U merupakan hasil triangulasi matriks A, yang juga mengakibatkan B berubah menjadi Y.

Mencari matriks L dan U:

$$\begin{pmatrix} l_{11} & 0 & 0 & \cdots & 0 \\ l_{21} & l_{22} & 0 & \cdots & 0 \\ l_{31} & l_{32} & l_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{pmatrix} \begin{pmatrix} 1 & u_{12} & u_{13} & \cdots & u_{1n} \\ 0 & 1 & u_{23} & \cdots & u_{2n} \\ 0 & 0 & 1 & \cdots & u_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix}$$

Diperoleh:

$$a_{i1} = l_{i1} \quad \rightarrow \quad l_{i1} = a_{i1} \quad (i = 1, \ldots, n)$$

$$a_{1j} = l_{11} u_{1j} \quad \rightarrow \quad u_{1j} = \frac{a_{1j}}{l_{11}} \quad (j = 2, \ldots, n)$$

$$a_{i2} = l_{i1} u_{12} + l_{i2} \quad \rightarrow \quad l_{i2} = a_{i2} - l_{i1} u_{12} \quad (i = 2, \ldots, n)$$

$$a_{2j} = l_{21} u_{1j} + l_{22} u_{2j} \quad \rightarrow \quad u_{2j} = \frac{a_{2j} - l_{21} u_{1j}}{l_{22}} \quad (j = 3, \ldots, n)$$

$$a_{i3} = l_{i1} u_{13} + l_{i2} u_{23} + l_{i3} \quad \rightarrow \quad l_{i3} = a_{i3} - l_{i1} u_{13} - l_{i2} u_{23} \quad (i = 3, \ldots, n)$$

$$a_{3j} = l_{31} u_{1j} + l_{32} u_{2j} + l_{33} u_{3j} \quad \rightarrow \quad u_{3j} = \frac{a_{3j} - l_{31} u_{1j} - l_{32} u_{2j}}{l_{33}} \quad (j = 4, \ldots, n)$$

$$\ldots$$

Jadi, elemen matriks L dan U dicari menurut:

$$l_{i1} = a_{i1} \qquad (i = 1, \ldots, n)$$

$$u_{1j} = \frac{a_{1j}}{l_{11}} \qquad (j = 2, \ldots, n)$$

$$l_{ij} = a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj} \qquad (i = j, \ldots, n;\ j = 2, \ldots, n)$$

$$u_{ij} = \frac{a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj}}{l_{ii}} \qquad (j = i+1, \ldots, n;\ i = 2, \ldots, n-1)$$

secara bergantian:

1. matriks L kolom 1, matriks U baris 1
2. matriks L kolom 2, matriks U baris 2
3. …
4. matriks L kolom (n-1), matriks U baris (n-1)
5. matriks L kolom n

Substitusi maju untuk menghitung y:

$$\begin{pmatrix} l_{11} & 0 & 0 & \cdots & 0 \\ l_{21} & l_{22} & 0 & \cdots & 0 \\ l_{31} & l_{32} & l_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{pmatrix}$$

$$\boxed{\begin{aligned} y_1 &= \frac{b_1}{l_{11}} \\ y_i &= \frac{b_i - \sum_{j=1}^{i-1} l_{ij} y_j}{l_{ii}} \quad (i = 2, \ldots, n) \end{aligned}}$$

Substitusi mundur untuk menghitung x:

$$\begin{pmatrix} 1 & u_{12} & u_{13} & \cdots & u_{1n} \\ 0 & 1 & u_{23} & \cdots & u_{2n} \\ 0 & 0 & 1 & \cdots & u_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix}$$

$$\boxed{\begin{aligned} x_n &= y_n \\ x_{n-i} &= y_{n-i} - \sum_{j=n-i+1}^{n} u_{n-i,j} x_j \quad (i = 1, \ldots, n-1) \end{aligned}}$$

Kembali ke soal $AX = B$, dengan $A = \begin{pmatrix} 2 & -3 & 2 \\ -1 & 2 & -3 \\ 1 & 1 & -1 \end{pmatrix}$, $B = \begin{pmatrix} -6 \\ 2 \\ 0 \end{pmatrix}$.

Jawab:

$$A = LU \longrightarrow L = \begin{pmatrix} 2 & 0 & 0 \\ -1 & 0.5 & 0 \\ 1 & 2.5 & 8 \end{pmatrix}, \quad U = \begin{pmatrix} 1 & -1.5 & 1 \\ 0 & 1 & -4 \\ 0 & 0 & 1 \end{pmatrix}$$

$$Y = UX, \quad LY = B \longrightarrow Y = \begin{pmatrix} -3 \\ -2 \\ 1 \end{pmatrix}$$

$$UX = Y \longrightarrow X = \begin{pmatrix} -1 \\ 2 \\ 1 \end{pmatrix}$$

Kasus Beberapa Sistem Persamaan Linear

Pada kasus yang lebih umum bisa saja terdapat beberapa sistem persamaan linear dengan nilai B yang berlainan, namun memiliki nilai A yang sama.

Dalam bentuk matriks sistem seperti ini dituliskan sebagai:

$$\begin{bmatrix} & & \\ & A & \\ & & \end{bmatrix}\begin{bmatrix} \\ X \\ \\ \end{bmatrix} = \begin{bmatrix} \\ B \\ \\ \end{bmatrix} \quad \text{atau} \quad AX = B$$

Keterangan:

- A matriks n x n, X dan B matriks n x m, dengan m = jumlah sistem persamaan linear, n = jumlah persamaan / unknown dalam tiap sistem persamaan tersebut
- Tiap kolom matriks X merupakan solusi untuk kolom yang sama pada matriks B.

Langkah dan rumus pada metode Eliminasi Gauss dan LU Decomposition berlaku sama untuk kasus ini. Hanya saja, di sini matriks X dan B terdiri dari beberapa kolom, bukan hanya satu.

Contoh dua sistem persamaan linear yang memiliki nilai A sama tapi B berbeda.

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_{11} \\ x_{21} \\ x_{31} \\ \vdots \\ x_{n1} \end{pmatrix} = \begin{pmatrix} b_{11} \\ b_{21} \\ b_{31} \\ \vdots \\ b_{n1} \end{pmatrix} \qquad \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_{12} \\ x_{22} \\ x_{32} \\ \vdots \\ x_{n2} \end{pmatrix} = \begin{pmatrix} b_{12} \\ b_{22} \\ b_{32} \\ \vdots \\ b_{n2} \end{pmatrix}$$

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ x_{31} & x_{32} \\ \vdots & \vdots \\ x_{n1} & x_{n2} \end{pmatrix} = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \\ \vdots & \vdots \\ b_{n1} & b_{n2} \end{pmatrix}$$

Metode Eliminasi Gauss:

- rumus triangulasi:

$$a_{ij}^{(0)} = a_{ij}, \quad b_{ir}^{(0)} = b_{ir} \qquad (i, j = 1, \ldots, n; r = 1, \ldots, m)$$

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} a_{kj}^{(k-1)} \qquad (k = 1, \ldots, n-1; i = k+1, \ldots, n;$$

$$b_{ir}^{(k)} = b_{ir}^{(k-1)} - \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}} b_{kr}^{(k-1)} \qquad j = k, \ldots, n; r = 1, \ldots, m)$$

$a_{ij}^{(m)}, b_{ir}^{(m)} \equiv a_{ij}, b_{ir}$
pada langkah ke m

- rumus substitusi mundur:

$$x_{nr} = \frac{b_{nr}^{(n-1)}}{a_{nn}^{(n-1)}} \qquad (r = 1, \ldots, m)$$

$$x_{n-j,r} = \frac{b_{n-j,r}^{(n-j-1)} - \sum_{k=n-j+1}^{n} a_{n-j,k}^{(n-j-1)} x_{kr}}{a_{n-j,n-j}^{(n-j-1)}} \qquad (j = 1, \ldots, n-1; r = 1, \ldots, m)$$

Metode LU Decomposition:

- rumus substitusi maju untuk menghitung y (kini Y matriks n x m):

$$y_{1r} = \frac{b_{1r}}{l_{11}} \qquad (r = 1, \dots, m)$$

$$y_{ir} = \frac{b_{ir} - \sum_{j=1}^{i-1} l_{ij} y_{jr}}{l_{ii}} \qquad (i = 2, \dots, n; r = 1, \dots, m)$$

- rumus substitusi mundur untuk menghitung x:

$$x_{nr} = y_{nr} \qquad (r = 1, \dots, m)$$

$$x_{n-i,r} = y_{n-i,r} - \sum_{j=n-i+1}^{n} u_{n-i,j} x_{jr} \qquad (i = 1, \dots, n-1; r = 1, \dots, m)$$

Perhatikan metode LU Decomposition, anggap matriks L dan U telah diperoleh. Jika kemudian terdapat lagi sistem persamaan linear dengan A sama dan B berbeda, maka matriks L dan U yang telah diperoleh itu bisa langsung dipakai untuk sistem persamaan yang baru tersebut.

Kini perhatikan metode Eliminasi Gauss, anggap triangulasi matriks A sudah dikerjakan. Jika kemudian terdapat lagi sistem persamaan linear dengan A sama dan B berbeda, maka hasil triangulasi matriks A yang sudah diperoleh tidak dapat dipakai untuk sistem persamaan yang baru. Untuk sistem yang baru tersebut proses triangulasi matriks A harus dilakukan lagi dari awal.

Hal ini disebabkan, matriks B harus berubah mengikuti proses triangulasi matriks A, sementara proses penguraian matriks A menjadi matriks L dan U tidak melibatkan matriks B.

Catatan:

Dalam rumus-rumus baik pada metode Eliminasi Gauss maupun LU Decomposition terdapat pembagian oleh elemen diagonal matriks yaitu, oleh elemen diagonal matriks A pada metode Eliminasi Gauss, dan elemen diagonal matriks L pada metode LU Decomposition.

Jika secara kebetulan elemen diagonal itu nol, maka akan timbul error.

Karena itu, pada setiap langkah dalam proses triangulasi matriks A (metode Eliminasi Gauss) atau pencarian matriks L dan U (metode LU Decomposition) perlu dilakukan pemeriksaan, apakah elemen matriks A atau L yang bersangkutan sama dengan nol.

Jika bernilai nol, maka baris berisi elemen diagonal nol itu harus ditukar dengan salah satu baris setelahnya, sehingga elemen diagonal menjadi bukan nol. Perubahan baris pada matriks A (metode Eliminasi Gauss) harus disertai perubahan baris yang sama pada matriks B. Perubahan baris pada matriks L (metode LU Decomposition) harus disertai perubahan baris yang sama pada matriks A dan B.

Soal:

$$\begin{pmatrix} 2 & -4 & 1 & 3 \\ -1 & 2 & 3 & -2 \\ 3 & -4 & 1 & 2 \\ 1 & -3 & -1 & 5 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}$$

Jawab:

$$\begin{pmatrix} 2 & -4 & 1 & 3 \\ 0 & \boxed{0} & 3.5 & -0.5 \\ 0 & 2 & -0.5 & -2.5 \\ 0 & -1 & -1.5 & 3.5 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ -1 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 2 & -4 & 1 & 3 \\ 0 & 2 & -0.5 & -2.5 \\ 0 & 0 & 3.5 & -0.5 \\ 0 & -1 & -1.5 & 3.5 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \\ 3 \\ 1 \end{pmatrix}$$

baris 2 ditukar dengan baris 3

$$\begin{pmatrix} 2 & -4 & 1 & 3 \\ 0 & 2 & -0.5 & -2.5 \\ 0 & 0 & 3.5 & -0.5 \\ 0 & 0 & -1.75 & 2.25 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \\ 3 \\ 0.5 \end{pmatrix}$$

$$\begin{pmatrix} 2 & -4 & 1 & 3 \\ 0 & 2 & -0.5 & -2.5 \\ 0 & 0 & 3.5 & -0.5 \\ 0 & 0 & 0 & 2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \\ 3 \\ 2 \end{pmatrix}$$

$$x_4 = 1$$

$$x_3 = \frac{3 + 0.5x_4}{3.5} = 1$$

$$x_2 = \frac{-1 + 0.5x_3 + 2.5x_4}{2} = 1$$

$$x_1 = \frac{2 + 4x_2 - x_3 - 3x_4}{2} = 1$$

$$A=\begin{pmatrix}2 & -4 & 1 & 3\\ -1 & 2 & 3 & -2\\ 3 & -4 & 1 & 2\\ 1 & -3 & -1 & 5\end{pmatrix} B=\begin{pmatrix}2\\2\\2\\2\end{pmatrix} \rightarrow L=\begin{pmatrix}2 & 0 & 0 & 0\\ -1 & l_{22} & 0 & 0\\ 3 & l_{32} & l_{33} & 0\\ 1 & l_{42} & l_{43} & l_{44}\end{pmatrix} \rightarrow U=\begin{pmatrix}1 & -2 & 0.5 & 1.5\\ 0 & 1 & u_{23} & u_{24}\\ 0 & 0 & 1 & u_{34}\\ 0 & 0 & 0 & 1\end{pmatrix}$$

$$L=\begin{pmatrix}2 & 0 & 0 & 0\\ -1 & \boxed{0} & 0 & 0\\ 3 & 2 & l_{33} & 0\\ 1 & -1 & l_{43} & l_{44}\end{pmatrix}$$

$$A=\begin{pmatrix}2 & -4 & 1 & 3\\ 3 & -4 & 1 & 2\\ -1 & 2 & 3 & -2\\ 1 & -3 & -1 & 5\end{pmatrix} L=\begin{pmatrix}2 & 0 & 0 & 0\\ 3 & 2 & 0 & 0\\ -1 & 0 & l_{33} & 0\\ 1 & -1 & l_{43} & l_{44}\end{pmatrix} \leftarrow L$$

$$B=\begin{pmatrix}2\\2\\2\\2\end{pmatrix} \rightarrow U=\begin{pmatrix}1 & -2 & 0.5 & 1.5\\ 0 & 1 & -0.25 & -1.25\\ 0 & 0 & 1 & u_{34}\\ 0 & 0 & 0 & 1\end{pmatrix} \rightarrow L=\begin{pmatrix}2 & 0 & 0 & 0\\ 3 & 2 & 0 & 0\\ -1 & 0 & 3.5 & 0\\ 1 & -1 & -1.75 & l_{44}\end{pmatrix}$$

baris 2 $\leftrightarrow$ baris 3

$$L=\begin{pmatrix}2 & 0 & 0 & 0\\ 3 & 2 & 0 & 0\\ -1 & 0 & 3.5 & 0\\ 1 & -1 & -1.75 & 2\end{pmatrix} \leftarrow U=\begin{pmatrix}1 & -2 & 0.5 & 1.5\\ 0 & 1 & -0.25 & -1.25\\ 0 & 0 & 1 & -1/7\\ 0 & 0 & 0 & 1\end{pmatrix}$$

Iterasi Jacobi

sistem persamaan linear: $\sum_{j=1}^{n} a_{ij}x_j = b_i \quad (i=1,\ldots,n)$ solusi: $x_i = \frac{1}{a_{ii}}\left(b_i - \sum_{j\neq i}^{n} a_{ij}x_j\right)$

Pencarian solusi dimulai dengan nilai awal $x_i^{(0)}$ (i = 1, …, n) hasil perkiraan / tebakan. Dengan nilai tebak awal ini diperoleh nilai perkiraan berikut $x_i^{(1)}$ melalui:

$$x_i^{(1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j\neq i}^{n} a_{ij}x_j^{(0)}\right) \quad (i=1,\ldots,n)$$

Demikian seterusnya berulang-ulang, nilai perkiraan pada langkah ke k diperoleh dari nilai perkiraan pada langkah ke k-1:

$$\boxed{x_i^{(k)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j\neq i}^{n} a_{ij}x_j^{(k-1)}\right) \quad (i=1,\ldots,n)}$$

Pencarian dihentikan setelah didapat nilai x_i yang konvergen yaitu, yang tidak atau sedikit berubah dari nilai yang diperoleh pada langkah sebelumnya:

$$\left|1 - \frac{x_i^{(k-1)}}{x_i^{(k)}}\right| < \varepsilon, \qquad \varepsilon = \text{bilangan kecil}$$

Iterasi Gauss-Siedel

Rumus iterasi Jacobi dapat ditulis: $x_i^{(k)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j<i} a_{ij}x_j^{(k-1)} - \sum_{j>i} a_{ij}x_j^{(k-1)}\right)$

Jika pada tiap langkah pencarian dilakukan dengan urutan i yang makin besar, maka semua $x_{j<i}^{(k)}$ sudah diperoleh ketika mencari $x_i^{(k)}$.

Sebaliknya, jika dilakukan dengan urutan i yang makin kecil, maka semua $x_{j>i}^{(k)}$ sudah diperoleh ketika mencari $x_i^{(k)}$.

Karena itu, nilai $x_{j<i}^{(k)}$ atau $x_{j>i}^{(k)}$ itu bisa langsung dipakai untuk mencari $x_i^{(k)}$, sehingga iterasi mencapai nilai konvergen menjadi lebih cepat:

$$\boxed{\begin{aligned} x_i^{(k)} &= \frac{1}{a_{ii}}\left(b_i - \sum_{j<i} a_{ij}x_j^{(k)} - \sum_{j>i} a_{ij}x_j^{(k-1)}\right) \qquad (i=1,2,\ldots,n) \\ x_i^{(k)} &= \frac{1}{a_{ii}}\left(b_i - \sum_{j<i} a_{ij}x_j^{(k-1)} - \sum_{j>i} a_{ij}x_j^{(k)}\right) \qquad (i=n,\ldots,2,1) \end{aligned}}$$

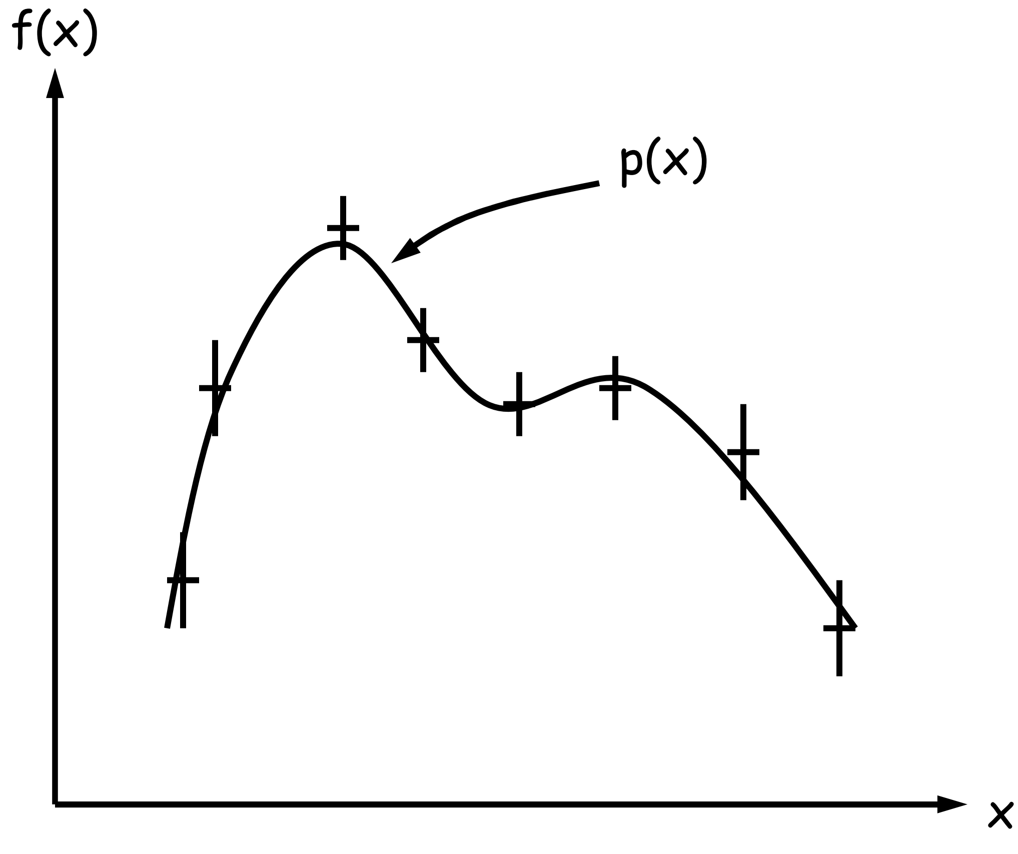
Iterasi seperti ini disebut iterasi Gauss-Siedel.

Kita lihat kembali metode Eliminasi Gauss dan LU Decomposition untuk mencari solusi sebuah sistem persamaan linear. Pada metode ini terdapat substitusi mundur dan maju. Pada substitusi mundur (maju), nilai x_i dihitung dari nilai $x_{j>i}$ ($x_{j<i}$), sehingga kesalahan (ketidakakuratan) pada $x_{j>i}$ ($x_{j<i}$) terakumulasi pada x_i . Dengan kata lain, terjadi perambatan kesalahan.

Pada metode iterasi tidak terdapat perambatan kesalahan seperti itu. Semua elemen x dilihat secara sama. Pada tiap langkah dilakukan pemeriksaan konvergensi untuk semua elemen x. Jadi, untuk tiap elemen x terdapat kesempatan yang sama untuk mencapai keakuratan yang diinginkan.

Namun, pada metode iterasi ada keharusan menentukan nilai awal, yang bisa saja sulit dilakukan atau menimbulkan masalah, misalnya membuat iterasi terlalu lama mencapai konvergensi.

Data Fitting dengan Metode Least Square



Keterangan:

- $f(x_i)$ mewakili data;
  $i = 1, \ldots, N$;
  N = jumlah data
- p(x) merupakan fungsi yang dicocokkan (fitted) terhadap data $f(x_i)$

Sifat fitting:
tidak selalu $p(x_i) = f(x_i)$ untuk semua x_i .

Prinsip penentuan fungsi p(x):

- p(x) merupakan polinomial orde m:

$$p(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \ldots + a_m x^m = \sum_{j=0}^{m} a_j x^j$$

(Secara umum, p(x) juga bisa merupakan polinomial bentuk yang lain seperti, polinomial Legendre.)

- Selisih antara p(x) dan f(x) untuk titik data tertentu:

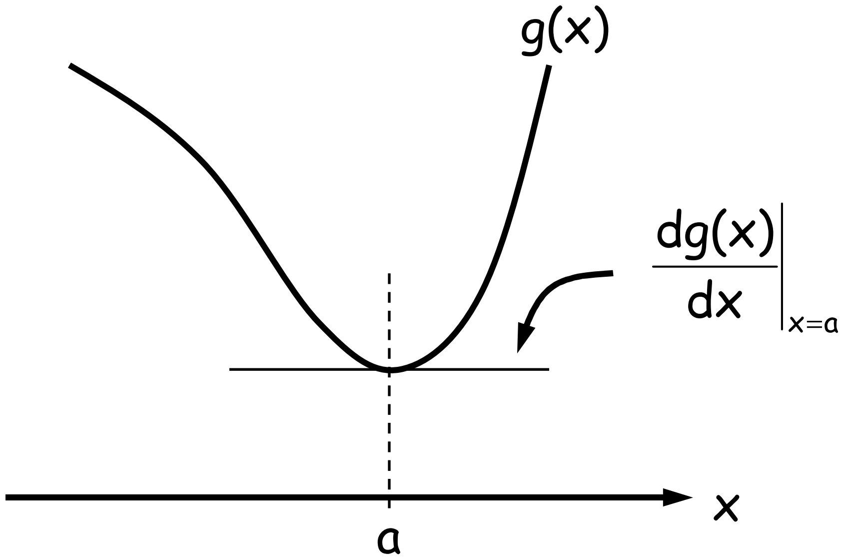
$$\Delta_i = f(x_i) - p(x_i) = f(x_i) - \sum_{j=0}^{m} a_j x_i^j \qquad (i = 1, \ldots, N)$$

- Jumlah kuadrat selisih antara p(x) dan f(x) untuk semua titik data:

$$S = \sum_{i=1}^{N} \Delta_i^2 = \sum_{i=1}^{N} \left(f(x_i) - p(x_i)\right)^2 = \sum_{i=1}^{N} \left(f(x_i) - \sum_{j=0}^{m} a_j x_i^j \right)^2$$

Fungsi p(x) ditentukan dengan mencari nilai a_j (j = 0, …, m) yang membuat S bernilai minimum.

Titik Minimum



g(a) merupakan titik minimum jika:

$$\boxed{\left.\frac{dg(x)}{dx}\right|_{x=a} = 0 \quad \text{dan} \quad \left.\frac{d^2g(x)}{dx^2}\right|_{x=a} > 0}$$

Spesial: fungsi kuadratik $g(x) = ax^2 + bx + c$

$$\left.\begin{aligned} \frac{dg(x)}{dx} &= 2ax + b \\ \frac{d^2g(x)}{dx^2} &= 2a \end{aligned}\right\}$$

g(x) memiliki satu titik minimun jika a > 0 atau sebaliknya satu titik maksimum jika a < 0.

S merupakan fungsi kuadratik dalam a_j (j = 0, …, m):

$$S(a_0,\dots,a_m)=\sum_{i=1}^{N}\left(f(x_i)-\sum_{j=0}^{m}a_j x_i^j\right)^2=\sum_{i=1}^{N}\left(\sum_{j=0}^{m}\left(a_j^2 x_i^{2j}+\dots\right)+f^2(x_i)\right)$$

$$\frac{\partial S(a_0,\dots,a_m)}{\partial a_k}=-2\sum_{i=1}^{N}\left(f(x_i)-\sum_{j=0}^{m}a_j x_i^j\right)x_i^k \qquad (k=0,\dots,m)$$

$$\frac{\partial^2 S(a_0,\dots,a_m)}{\partial a_k{}^2}=2\sum_{i=1}^{N}x_i^{2k}>0 \qquad (k=0,\dots,m)$$

S memiliki satu titik minimum pada nilai a_j (j = 0, …, m) tertentu.

Mencari a_j $(j = 0, \ldots, m)$:

$$\frac{\partial S(a_0,\ldots,a_m)}{\partial a_k} = -2\sum_{i=1}^{N}\left(f(x_i) - \sum_{j=0}^{m} a_j x_i^j\right)x_i^k = 0 \qquad (k = 0,\ldots,m)$$

$$\longrightarrow \sum_{j=0}^{m}\left(\sum_{i=1}^{N} x_i^{j+k}\right)a_j = \sum_{i=1}^{N} f(x_i)x_i^k \qquad (k = 0,\ldots,m)$$

Definisikan: $$\boxed{c_{kj} \equiv \sum_{i=1}^{N} x_i^{j+k} \qquad b_k \equiv \sum_{i=1}^{N} f(x_i)x_i^k}$$

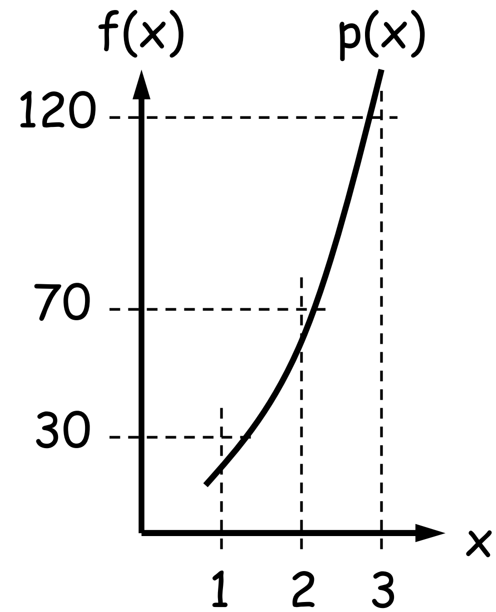
maka diperoleh sebuah sistem persamaan linear: $\sum_{j=0}^{m} c_{kj}a_j = b_k \qquad (k = 0,\ldots,m)$

dalam bentuk matrik: $\begin{pmatrix} & C & \end{pmatrix}\begin{pmatrix} A \end{pmatrix} = \begin{pmatrix} B \end{pmatrix}$ atau $CA = B$

Jadi, a_j $(j = 0, \ldots, m)$ diperoleh sebagai solusi persamaan linear $CA = B$.

Contoh: Terdapat tiga data f(x) yaitu, f(1) = 30, f(2) = 70 dan f(3) = 120.
Cari fungsi p(x) yang dapat melukiskan data itu.

Dari data itu jelas p(x) bukan fungsi linear.
Jadi, dicoba fungsi kuadratik:



$$p(x) = a_0 + a_1 x + a_2 x^2$$

Sistem persamaan linier untuk mencari a_j :

$$\begin{pmatrix} 3 & 6 & 14 \\ 6 & 14 & 36 \\ 14 & 36 & 98 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 220 \\ 530 \\ 1390 \end{pmatrix}$$

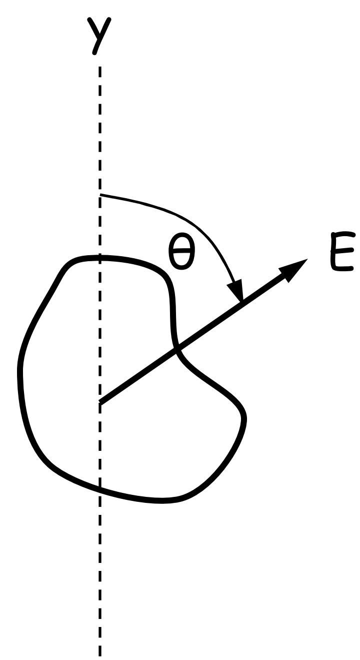
$$\begin{pmatrix} 3 & 6 & 14 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 220 \\ 45 \\ 5 \end{pmatrix} \longrightarrow \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 25 \\ 5 \end{pmatrix}$$

Jadi, $\boxed{p(x) = 5x(x+5)}$ $\longrightarrow$ Cek: p(1) = 30, p(2) = 70,
p(3) = 120

OK!

Contoh: Kuat medan listrik E di sekitar sebuah benda berbentuk lempeng diukur pada jarak 10 cm dari pusat massanya dan arah yang bervariasi. Arah dinyatakan dalam sudut θ terhadap sumbu y yang ditetapkan sebelum pengukuran. Diperoleh data sebagai berikut:

θ [derajat]	E [V/cm]
10	0.01794775
15	0.03808997
20	0.05516225
25	0.05598281
30	0.04795629
35	0.04807485
40	0.06273566
45	0.07853982
50	0.07395442
55	0.04201338



Cari fungsi p(x) yang dapat melukiskan data itu.

Dicoba beberapa polinomial dengan orde berbeda, diperoleh:

m = 3:

$a_0 = 8.983713484853211E-03$
$a_1 = 1.324478388111303E-03$
$a_2 = 3.487808787880805E-05$
$a_3 = -8.085809790211842E-07$

$S = 1.0339E-03$

m = 5:

$a_0 = -3.557800654975570E-02$
$a_1 = 1.061996221844471E-03$
$a_2 = 8.802185976358352E-04$
$a_3 = -5.862332690401015E-05$
$a_4 = 1.362046192596346E-06$
$a_5 = -1.063951754163944E-08$

$S = 8.1573E-05$

m = 9:

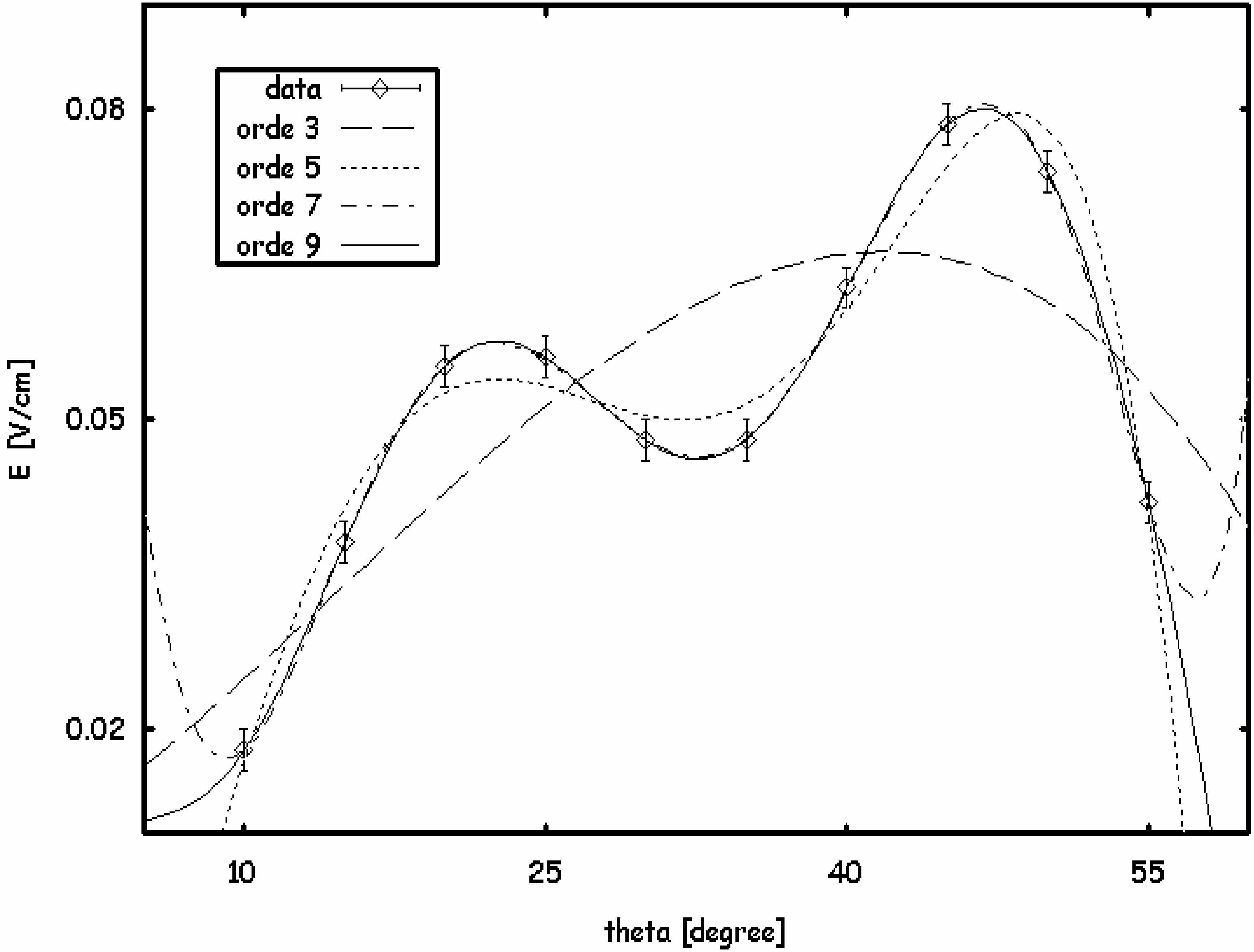
$a_0 = -1.757260839248139E-02$
$a_1 = 1.596300085173997E-02$
$a_2 = -3.402768734407800E-03$
$a_3 = 3.358961098305538E-04$
$a_4 = -1.368895999268855E-05$
$a_5 = 1.132254508386570E-07$
$a_6 = 8.262829873458547E-09$
$a_7 = -2.741786330789355E-10$
$a_8 = 3.317446724324134E-12$
$a_9 = -1.459511835946927E-14$

$S = 1.7528E-11$

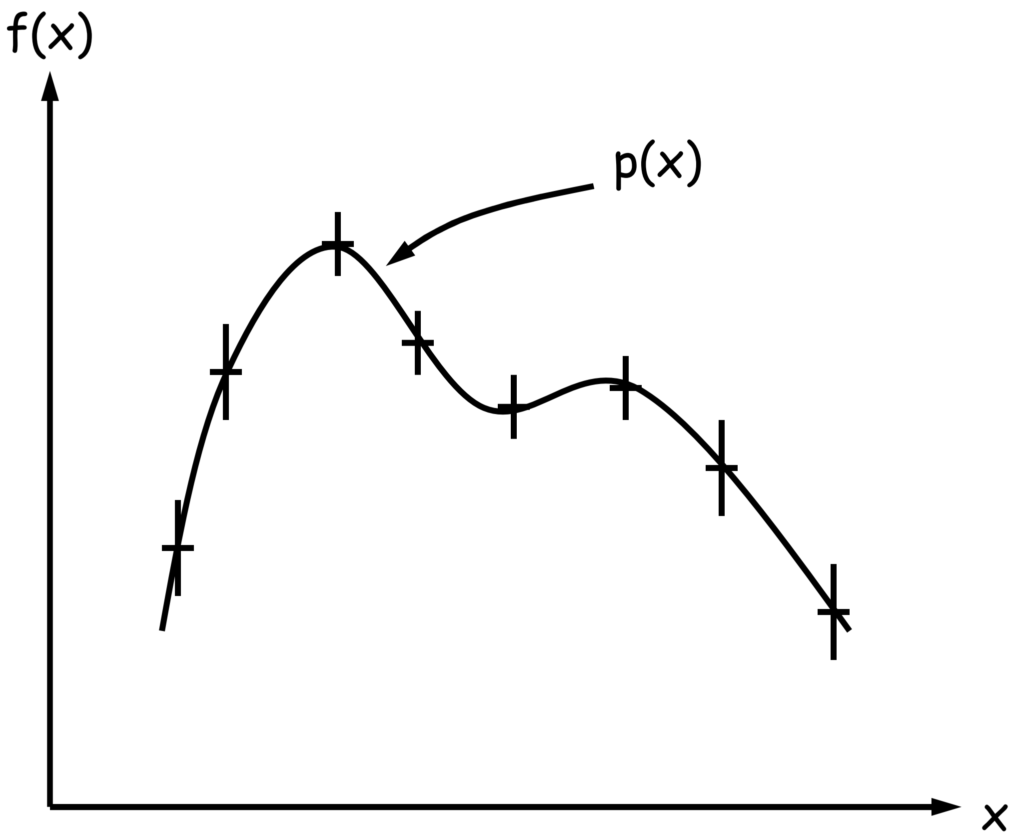
m = 7:

$a_0 = 1.864754537649403E-01$
$a_1 = -4.631839872868015E-02$
$a_2 = 4.007658091692495E-03$
$a_3 = -8.985715636865594E-05$
$a_4 = -3.230489224228010E-06$
$a_5 = 1.912806006890119E-07$
$a_6 = -3.252863805243949E-09$
$a_7 = 1.876184315740421E-11$

$S = 3.1629E-07$


data
orde 3
orde 5
orde 7
orde 9
0.08
0.05
0.02
E [V/cm]
10
25
40
55
theta [degree]

Interpolasi



Keterangan:

- $f(x_i)$ mewakili data;
  $i = 1, \ldots, N$;
  N = jumlah data
- p(x) merupakan fungsi interpolasi berdasarkan data $f(x_i)$

Sifat interpolasi:
$p(x_i) = f(x_i)$
untuk semua x_i .

Interpolasi Lagrange

Digunakan p(x), suatu polinomial berorde m = N – 1, dengan N = jumlah data:

$$p(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{N-1} x^{N-1} \cong f(x)$$

Nilai a_i (i = 0, …, N-1) ditetukan dengan menetapkan bahwa untuk semua titik data:

$$p(x_i) = f(x_i) \qquad (i = 1, \ldots, N)$$

Jadi, diperoleh persamaan linear:

$$\begin{aligned}
p(x_1) &= a_0 + a_1 x_1 + a_2 x_1^2 + \ldots + a_{N-1} x_1^{N-1} = f(x_1) \\
p(x_2) &= a_0 + a_1 x_2 + a_2 x_2^2 + \ldots + a_{N-1} x_2^{N-1} = f(x_2) \\
p(x_3) &= a_0 + a_1 x_3 + a_2 x_3^2 + \ldots + a_{N-1} x_3^{N-1} = f(x_3) \\
&\ldots \\
p(x_N) &= a_0 + a_1 x_N + a_2 x_N^2 + \ldots + a_{N-1} x_N^{N-1} = f(x_N)
\end{aligned}$$

dan a_i (i = 0, …, N-1) diperoleh sebagai solusi dari persamaan linear itu.

N = 2:

$$p(x_1) = a_0 + a_1 x_1 = f(x_1)$$
$$p(x_2) = a_0 + a_1 x_2 = f(x_2)$$

$$a_0 = -\frac{x_2 f(x_1) - x_1 f(x_2)}{x_1 - x_2} \qquad a_1 = \frac{f(x_1) - f(x_2)}{x_1 - x_2}$$

$$p(x) = \left(\frac{x - x_2}{x_1 - x_2}\right) f(x_1) + \left(\frac{x - x_1}{x_2 - x_1}\right) f(x_2)$$

N = 3:

$$p(x_1) = a_0 + a_1 x_1 + a_2 x_1^2 = f(x_1)$$
$$p(x_2) = a_0 + a_1 x_2 + a_2 x_2^2 = f(x_2)$$
$$p(x_3) = a_0 + a_1 x_3 + a_2 x_3^2 = f(x_3)$$

$$a_0 = \frac{(x_2 - x_3) x_2 x_3 f(x_1) + (x_3 - x_1) x_3 x_1 f(x_2) + (x_1 - x_2) x_1 x_2 f(x_3)}{(x_2 - x_3) x_1^2 + (x_3 - x_1) x_2^2 + (x_1 - x_2) x_3^2}$$

$$a_1 = -\frac{(x_2^2 - x_3^2) f(x_1) + (x_3^2 - x_1^2) f(x_2) + (x_1^2 - x_2^2) f(x_3)}{(x_2 - x_3) x_1^2 + (x_3 - x_1) x_2^2 + (x_1 - x_2) x_3^2}$$

$$a_2 = \frac{(x_2 - x_3) f(x_1) + (x_3 - x_1) f(x_2) + (x_1 - x_2) f(x_3)}{(x_2 - x_3) x_1^2 + (x_3 - x_1) x_2^2 + (x_1 - x_2) x_3^2}$$

$$p(x) = \left(\frac{x - x_2}{x_1 - x_2}\right)\left(\frac{x - x_3}{x_1 - x_3}\right) f(x_1) + \left(\frac{x - x_1}{x_2 - x_1}\right)\left(\frac{x - x_3}{x_2 - x_3}\right) f(x_2) + \left(\frac{x - x_1}{x_3 - x_1}\right)\left(\frac{x - x_2}{x_3 - x_2}\right) f(x_3)$$

Secara umum, untuk N data rumus interpolasi Lagrange:

$$\boxed{\begin{aligned} p(x) &= \sum_{i=1}^{N} l(x,x_i)f(x_i) \\ l(x,x_i) &= \prod_{j\neq i}\left(\frac{x-x_j}{x_i-x_j}\right) \end{aligned}}$$

Untuk $x = x_k$ (k = 1, …, N):

$$l(x_k,x_i) = \prod_{j\neq i}\left(\frac{x_k-x_j}{x_i-x_j}\right) = \begin{cases} \displaystyle\prod_{j\neq i}\left(\frac{x_i-x_j}{x_i-x_j}\right) & = 1, \quad (i = k) \\ \displaystyle\ldots\left(\frac{x_k-x_k}{x_i-x_j}\right)\ldots & = 0, \quad (i \neq k) \end{cases}$$

$$\longrightarrow \quad l(x_k,x_i) = \delta_{ik} \qquad p(x_k) = f(x_k)$$

Perlukah memakai semua N data yang ada?

Pada bagian sebelum ini interpolasi menggunakan seluruh N data $f(x_i)$ yang tersedia, yang berarti menggunakan polinomial p(x) berorde N-1.

Kini, misal N = 4 dan x berada di sekitar x_4 , maka diperoleh:

$$l(x,x_1)=\left(\frac{x-x_2}{x_1-x_2}\right)\left(\frac{x-x_3}{x_1-x_3}\right)\left(\frac{x-x_4}{x_1-x_4}\right) \qquad l(x,x_2)=\left(\frac{x-x_1}{x_2-x_1}\right)\left(\frac{x-x_3}{x_2-x_3}\right)\left(\frac{x-x_4}{x_2-x_4}\right)$$

$$l(x,x_3)=\left(\frac{x-x_1}{x_3-x_1}\right)\left(\frac{x-x_2}{x_3-x_2}\right)\left(\frac{x-x_4}{x_3-x_4}\right) \qquad l(x,x_4)=\left(\frac{x-x_1}{x_4-x_1}\right)\left(\frac{x-x_2}{x_4-x_2}\right)\left(\frac{x-x_3}{x_4-x_3}\right)$$

Dapat dilihat bahwa, $l(x,x_1)<l(x,x_2)<l(x,x_3)<l(x,x_4)$.

Ini berarti, semakin jauh dari x pengaruh data $f(x_i)$ semakin kecil dalam menentukan nilai p(x). Data yang penting yaitu yang berada di sekitar titik x. Karena itu, cukup data-data di sekitar titik x yang digunakan.

Dengan kata lain, untuk interpolasi cukup digunakan polinomial p(x) berorde rendah, contoh berorde 3 (fungsi kubik).

Interpolasi Lagrange Kubik

Interpolasi Lagrange Kubik menggunakan polinomial p(x) berorde 3 sebagai fungsi interpolasi:

$$p(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 \cong f(x)$$

Untuk mencari nilai a_j (j = 0, 1, 2, 3) diperlukan 4 data f(x_i) di sekitar x:

$$f(x_0),\ f(x_1),\ f(x_2),\ f(x_3) \qquad \left(x_i \le x \le x_{i+1};\ x_0 = x_{i-1}, x_1 = x_i, x_2 = x_{i+1}, x_3 = x_{i+2}\right)$$

untuk membentuk sistem persamaan linear:

$$a_0 + a_1 x_j + a_2 x_j^2 + a_3 x_j^3 = f(x_j) \qquad (j = 0,1,2,3)$$

Langkah pertama dengan begitu, menentukan x_j (j = 0, 1, 2, 3) dengan melihat posisi x di antara titik data x_i (i = 1, …, N).

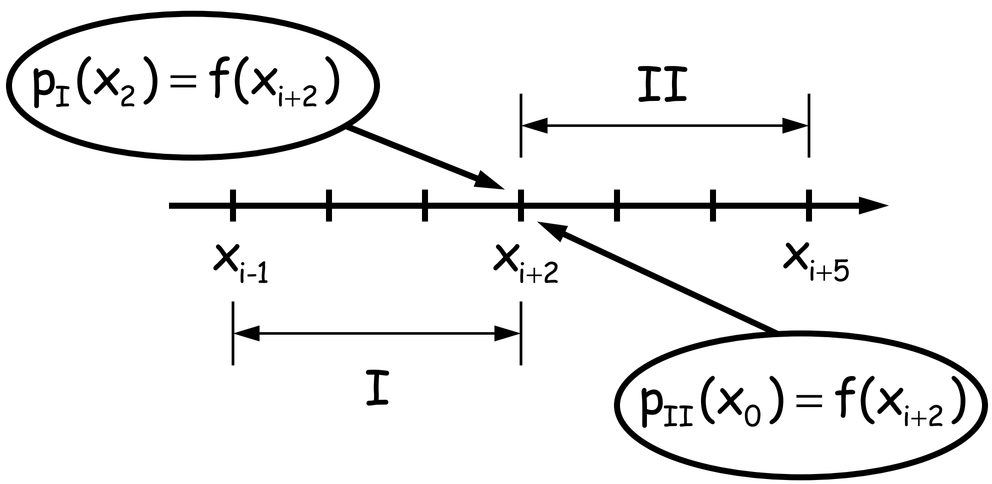
Diperoleh

$$\boxed{p(x) = \sum_{j=0}^{3} l(x, x_j) f(x_j) \qquad l(x, x_j) = \prod_{k \ne j} \left(\frac{x - x_k}{x_j - x_k} \right)}$$

Catatan:

Karena fungsi interpolasi p(x) dicocokkan dengan data $f(x_0 = x_{i-1})$, ..., $f(x_3 = x_{i+2})$ maka p(x) berlaku hanya untuk daerah $x_{i-1} \le x \le x_{i+2}$. Untuk daerah x yang lain berlaku fungsi interpolasi p(x) yang lain.

Pada batas antara dua daerah yang bersebelahan, masing-masing fungsi interpolasi p(x) dari kedua daerah berbeda itu menunjukan nilai yang sama, karena dalam menentukan p(x) selalu dibuat agar p(x) cocok dengan setiap titik data dalam daerah itu.



Dengan kata lain, p(x) bersifat kontinyu. Tetapi, tidak begitu dengan turunannya: p'(x) bersifat diskontinyu pada batas dua daerah yang bersebelahan.

Interpolasi Hermite Kubik

Dengan menggunakan polinomial p(x) berorde 3 (kubik), interpolasi dilakukan di antara dua titik data yang berurutan, yaitu dalam interval $x_i \le x \le x_{i+1}$:

$$p(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 \cong f(x)$$

Jadi, yang pertama dilakukan yaitu, menentukan posisi x di antara titik data x_i (i = 1, …, N).

Untuk mencari a_j (j = 0, 1, 2, 3) diperlukan 4 persamaan, yang ditetapkan sebagai berikut:

$$\begin{aligned}
p(x_1) &= a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3 = f(x_1) \\
p(x_2) &= a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^3 = f(x_2) \\
p'(x_1) &= a_1 + 2a_2 x_1 + 3a_3 x_1^2 = f'(x_1) \\
p'(x_2) &= a_1 + 2a_2 x_2 + 3a_3 x_2^2 = f'(x_2)
\end{aligned}
\qquad (x_i \le x \le x_{i+1};\ \ x_1 = x_i,\ \ x_2 = x_{i+1})$$

Jadi, pada interpolasi Hermite diperlukan sebagai data bukan saja f(x) namun juga turunannya f′(x).

Diperoleh a_j (j = 0, 1, 2, 3) sebagai berikut:

$$
\begin{aligned}
a_0 &= \left(\frac{x_1^2(3x_2 - x_1)f(x_2) + x_2^2(x_2 - 3x_1)f(x_1)}{(x_2 - x_1)^3}\right) \\
&\quad - x_2 x_1 \left(\frac{x_1 f'(x_2) + x_2 f'(x_1)}{(x_2 - x_1)^2}\right) \\
a_1 &= -6x_2 x_1 \left(\frac{f(x_2) - f(x_1)}{(x_2 - x_1)^3}\right) \\
&\quad + \left(\frac{x_1(2x_2 + x_1)f'(x_2) + x_2(x_2 + 2x_1)f'(x_1)}{(x_2 - x_1)^2}\right) \\
a_2 &= 3(x_2 + x_1)\left(\frac{f(x_2) - f(x_1)}{(x_2 - x_1)^3}\right) \\
&\quad - \left(\frac{(x_2 + 2x_1)f'(x_2) + (2x_2 + x_1)f'(x_1)}{(x_2 - x_1)^2}\right) \\
a_3 &= -2\left(\frac{f(x_2) - f(x_1)}{(x_2 - x_1)^3}\right) + \left(\frac{f'(x_2) + f'(x_1)}{(x_2 - x_1)^2}\right)
\end{aligned}
$$

Dengan a_j (j = 0, 1, 2, 3) yang sudah diperoleh, didapat fungsi interpolasi p(x) sebagai berikut:

$$p(x) = \sum_{j=1}^{2} \left(h_1(x, x_j) f(x_j) + h_2(x, x_j) f'(x_j) \right)$$

$$h_1(x, x_1) = \left(1 - 2\frac{(x - x_1)}{(x_1 - x_2)}\right)\left(\frac{x - x_2}{x_1 - x_2}\right)^2$$

$$h_1(x, x_2) = \left(1 - 2\frac{(x - x_2)}{(x_2 - x_1)}\right)\left(\frac{x - x_1}{x_2 - x_1}\right)^2$$

$$h_2(x, x_1) = (x - x_1)\left(\frac{x - x_2}{x_1 - x_2}\right)^2$$

$$h_2(x, x_2) = (x - x_2)\left(\frac{x - x_1}{x_2 - x_1}\right)^2$$

Pada interpolasi Hermite bukan saja p(x) yang dicocokkan dengan data f(x) namun juga turunannya p'(x) dicocokkan dengan data f'(x). Karena itu, baik p(x) maupun p'(x) bersifat kontinyu. Ini berbeda dari yang ditemui pada interpolasi Lagrange.

Interpolasi Hermite Orde Lebih Tinggi

Interpolasi Hermite tidak terbatas hanya menggunakan polinomial p(x) berorde 3 (kubik), namun dapat juga yang berorde lebih tinggi. Untuk itu diperlukan lebih banyak data, bukan hanya f(x) dan f'(x) pada titik x_i dan x_{i+1} .

Secara umum fungsi interpolasi Hermite p(x) berupa polinomial berorde (2n - 1) memerlukan n data f(x) dan n data f'(x):

$$\boxed{p(x) = \sum_{j=1}^{n} \left(h_1(x, x_j) f(x_j) + h_2(x, x_j) f'(x_j) \right)}$$

dengan:

$$h_1(x, x_j) = \left(1 - 2(x - x_j) l'(x_j)\right) l^2(x, x_j)$$

$$h_2(x, x_j) = (x - x_j) l^2(x, x_j)$$

$$l(x, x_j) = \prod_{k \neq j} \left(\frac{x - x_k}{x_j - x_k} \right)$$

$$l'(x_j) = \sum_{k \neq j} \frac{1}{(x_j - x_k)}$$

Interpolasi Hermite Kubik tanpa Data f'(x)

Interpolasi Hermite memerlukan sebagai data selain f(x) juga f'(x). Pada beberapa kasus bisa saja data f'(x) tidak tersedia, melainkan hanya data f(x). Pada kasus ini sebenarnya interpolasi Hermite tidak bisa dipakai. Tetapi, jika f'(x) bisa diperoleh melalui pendekatan (approximation) maka, interpolasi Hermite bisa dipakai.

$f'(x_i)$ dapat dihitung sebagai turunan sebuah fungsi kuadratik $g(x)$, yang dicocokkan dengan data f(x) pada titik-titik x_{i-1}, x_i, x_{i+1} :

$$g(x) = ax^2 + bx + c \longrightarrow \left.\begin{aligned} g(x_{i-1}) &= f(x_{i-1}) \\ g(x_i) &= f(x_i) \\ g(x_{i+1}) &= f(x_{i+1}) \end{aligned}\right\} (a, b, c) \longrightarrow \boxed{f'(x_i) \cong g'(x_i) = 2ax_i + b}$$

Dapat dilihat bahwa, proses pencarian f'(x) ini berdiri sendiri, berada di luar atau bukan bagian dari proses interpolasi Hermite. Dengan begitu, sifat kontinyu fungsi interpolasi Hermite p(x) dan turunannya p'(x) tidak berubah.

Dari sistem persamaan linear:

$$
\begin{aligned}
ax_{i-1}^2 + bx_{i-1} + c &= f(x_{i-1}) \\
ax_i^2 \;\, + bx_i \;\, + c &= f(x_i) \\
ax_{i+1}^2 + bx_{i+1} + c &= f(x_{i+1})
\end{aligned}
$$

diperoleh:

$$
a = \frac{f(x_{i-1})}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})} + \frac{f(x_i)}{(x_i-x_{i-1})(x_i-x_{i+1})} + \frac{f(x_{i+1})}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)}
$$

$$
b = -\frac{(x_i+x_{i+1})f(x_{i-1})}{(x_{i-1}-x_i)(x_{i-1}-x_{i+1})} - \frac{(x_{i-1}+x_{i+1})f(x_i)}{(x_i-x_{i-1})(x_i-x_{i+1})} - \frac{(x_{i-1}+x_i)f(x_{i+1})}{(x_{i+1}-x_{i-1})(x_{i+1}-x_i)}
$$

sehingga:

$$
\boxed{f'(x_i) \cong \left(\frac{x_i-x_{i+1}}{x_{i-1}-x_{i+1}}\right)\frac{f(x_{i-1})}{(x_{i-1}-x_i)} + \left(\frac{1}{x_i-x_{i+1}} + \frac{1}{x_i-x_{i-1}}\right)f(x_i) + \left(\frac{x_i-x_{i-1}}{x_{i+1}-x_{i-1}}\right)\frac{f(x_{i+1})}{(x_{i+1}-x_i)}}
$$

Jika diaplikasikan pada interpolasi Hermite kubik:

$$p(x) = \sum_{j=1}^{2} \left(h_1(x, x_j) f(x_j) + h_2(x, x_j) f'(x_j) \right)$$

maka diperoleh fungsi interpolasi Hermite kubik p(x) sebagai berikut:

$$p(x) = \sum_{j=0}^{3} h(x, x_j) f(x_j) \qquad (x_i \le x \le x_{i+1};\ \ x_0 = x_{i-1},\ \ x_1 = x_i,\ \ x_2 = x_{i+1},\ \ x_3 = x_{i+2})$$

$$h(x, x_0) = h_2(x, x_1)\left(\frac{x_1 - x_2}{x_0 - x_2}\right)\frac{1}{(x_0 - x_1)}$$

$$h(x, x_1) = h_1(x, x_1) + h_2(x, x_1)\left(\frac{1}{x_1 - x_2} + \frac{1}{x_1 - x_0}\right) + h_2(x, x_2)\left(\frac{x_2 - x_3}{x_1 - x_3}\right)\frac{1}{(x_1 - x_2)}$$

$$h(x, x_2) = h_1(x, x_2) + h_2(x, x_2)\left(\frac{1}{x_2 - x_1} + \frac{1}{x_2 - x_3}\right) + h_2(x, x_1)\left(\frac{x_1 - x_0}{x_2 - x_0}\right)\frac{1}{(x_2 - x_1)}$$

$$h(x, x_3) = h_2(x, x_2)\left(\frac{x_2 - x_1}{x_3 - x_1}\right)\frac{1}{(x_3 - x_2)}$$

Interpolasi Spline Kubik

Seperti interpolasi Lagrange, interpolasi Spline kubik juga memerlukan hanya f(x) sebagai data. Namun, turunan fungsi interpolasi Spline kubik p'(x) dibuat bersifat kontinyu.

Interpolasi Spline kubik menggunakan polinomial p(x) orde 3, untuk $x_i \le x \le x_{i+1}$:

$$p(x) = d_i + c_i(x - x_i) + b_i(x - x_i)^2 + a_i(x - x_i)^3 \cong f(x)$$

Turunan pertama dan kedua p(x) yaitu:

$$p'(x) = c_i + 2b_i(x - x_i) + 3a_i(x - x_i)^2$$
$$p''(x) = 2b_i + 6a_i(x - x_i)$$

Evaluasi pada titik $x = x_i$ menghasilkan:

$$p_i \equiv p(x_i) = d_i = f(x_i) \qquad p''_i \equiv p''(x_i) = 2b_i$$

dan pada titik $x = x_{i+1}$:

$$p''_{i+1} \equiv p''(x_{i+1}) = 2b_i + 6a_i h_i \quad p_{i+1} \equiv p(x_{i+1}) = d_i + c_i h_i + b_i h_i^2 + a_i h_i^3 = f(x_{i+1}) \quad h_i \equiv x_{i+1} - x_i$$

Jadi,

$$d_i = p_i \qquad b_i = \frac{p''_i}{2} \qquad a_i = \frac{p''_{i+1} - p''_i}{6h_i} \qquad c_i = \frac{p_{i+1} - p_i}{h_i} - \frac{h_i p''_{i+1} + 2h_i p''_i}{6}$$

sehingga diperoleh:

$$\boxed{p(x) = p_i + \left(\frac{p_{i+1} - p_i}{h_i} - \frac{h_i p''_{i+1}}{6} - \frac{h_i p''_i}{3}\right)(x - x_i) + \frac{p''_i}{2}(x - x_i)^2 + \left(\frac{p''_{i+1} - p''_i}{6h_i}\right)(x - x_i)^3}$$

$$p'(x) = \frac{p_{i+1} - p_i}{h_i} - \frac{h_i p''_{i+1}}{6} - \frac{h_i p''_i}{3} + p''_i(x - x_i) + \left(\frac{p''_{i+1} - p''_i}{2h_i}\right)(x - x_i)^2$$

p(x) telah dicocokkan dengan data f(x) di titik-titik batas interval, sehingga bersifat kontinyu. Untuk membuat p'(x) kontinyu maka dicari ekspresi p'(x) untuk daerah sebelumnya $x_{i-1} \le x \le x_i$:

$$p'(x) = \frac{p_i - p_{i-1}}{h_{i-1}} - \frac{h_{i-1} p''_i}{6} - \frac{h_{i-1} p''_{i-1}}{3} + p''_{i-1}(x - x_{i-1}) + \left(\frac{p''_i - p''_{i-1}}{2h_{i-1}}\right)(x - x_{i-1})^2$$

dan disamakan dengan p'(x) untuk daerah $x_i \le x \le x_{i+1}$ di titik $x = x_i$.

Untuk N = jumlah data, diperoleh:

$$h_{i-1}p''_{i-1}+2(h_{i-1}+h_i)p''_i+h_i p''_{i+1}=6\left(\frac{p_{i+1}-p_i}{h_i}-\frac{p_i-p_{i-1}}{h_{i-1}}\right) \qquad (i=2,\ldots,N-1)$$

Untuk menghitung p(x) diperlukan p''(x) di semua N titik data. (N-2) buah persamaan di atas tidak cukup untuk mendapatkan p''(x) di semua titik data. Masih diperlukan 2 persamaan lagi, yang diperoleh dengan mengevaluasi p'(x) di titik awal $x=x_1$ (memakai ekspresi p'(x) untuk $x_1 \le x \le x_2$) dan akhir $x=x_N$ (memakai ekspresi p'(x) untuk $x_{N-1} \le x \le x_N$). Didapat:

$$(i=1) \qquad 2h_1 p''_1+h_1 p''_2=6\left(\frac{p_2-p_1}{h_1}-p'_1\right)$$

$$(i=N) \qquad h_{N-1}p''_{N-1}+2h_{N-1}p''_N=6\left(p'_N-\frac{p_N-p_{N-1}}{h_{N-1}}\right)$$

Masalah: p'(x) di titik awal $x=x_1$ dan akhir $x=x_N$ tidak diketahui, $\longrightarrow$??

Ada dua cara. Pertama yang disebut spline alamiah yaitu, menetapkan p''(x) di titik awal $x=x_1$ dan akhir $x=x_N$ sama dengan nol. Kedua, menebak nilai p'(x) di titik awal $x=x_1$ dan akhir $x=x_N$.

Interpolasi Multidimensi

Jika data bergantung pada lebih dari satu variabel, maka dilakukan interpolasi multidimensi. Metode interpolasi yang telah disampaikan bisa dipakai untuk melakukan interpolasi multidimensi. Sebagai contoh di sini ditunjukkan interpolasi 2 dimensi. Untuk dimensi lebih tinggi berlaku cara yang sama.

$$\boxed{p(x,y) = \sum_{i=1}^{n} S(x,x_i) \sum_{j=1}^{m} S(y,y_j) f(x_i,y_j)}$$

Pada contoh di atas, interpolasi menggunakan (n x m) data f(x,y). Interpolasi dilakukan per dimensi: Untuk satu titik data x tertentu dilakukan interpolasi di sepanjang sumbu y, hal yang sama dilakukan untuk semua titik data x yang lain. Prinsip yang sama berlaku untuk interpolasi berdimensi lebih tinggi.

Contoh, interpolasi Lagrange kubik:

$$p(x,y) = \sum_{i=0}^{3} l(x,x_i) \sum_{j=0}^{3} l(y,y_j) f(x_i,y_j)$$

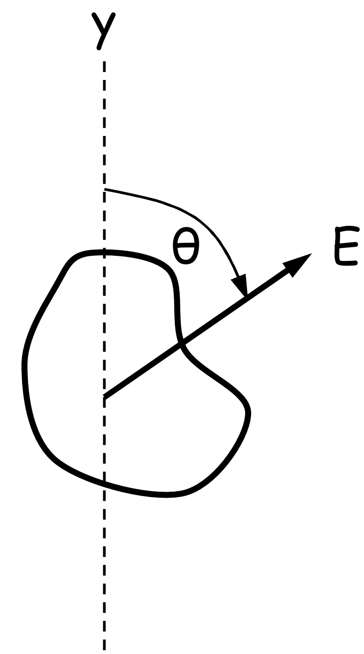
$$l(x,x_i) = \prod_{k \neq i} \left(\frac{x - x_k}{x_i - x_k} \right)$$

$$l(y,y_j) = \prod_{s \neq j} \left(\frac{y - y_s}{y_j - y_s} \right)$$

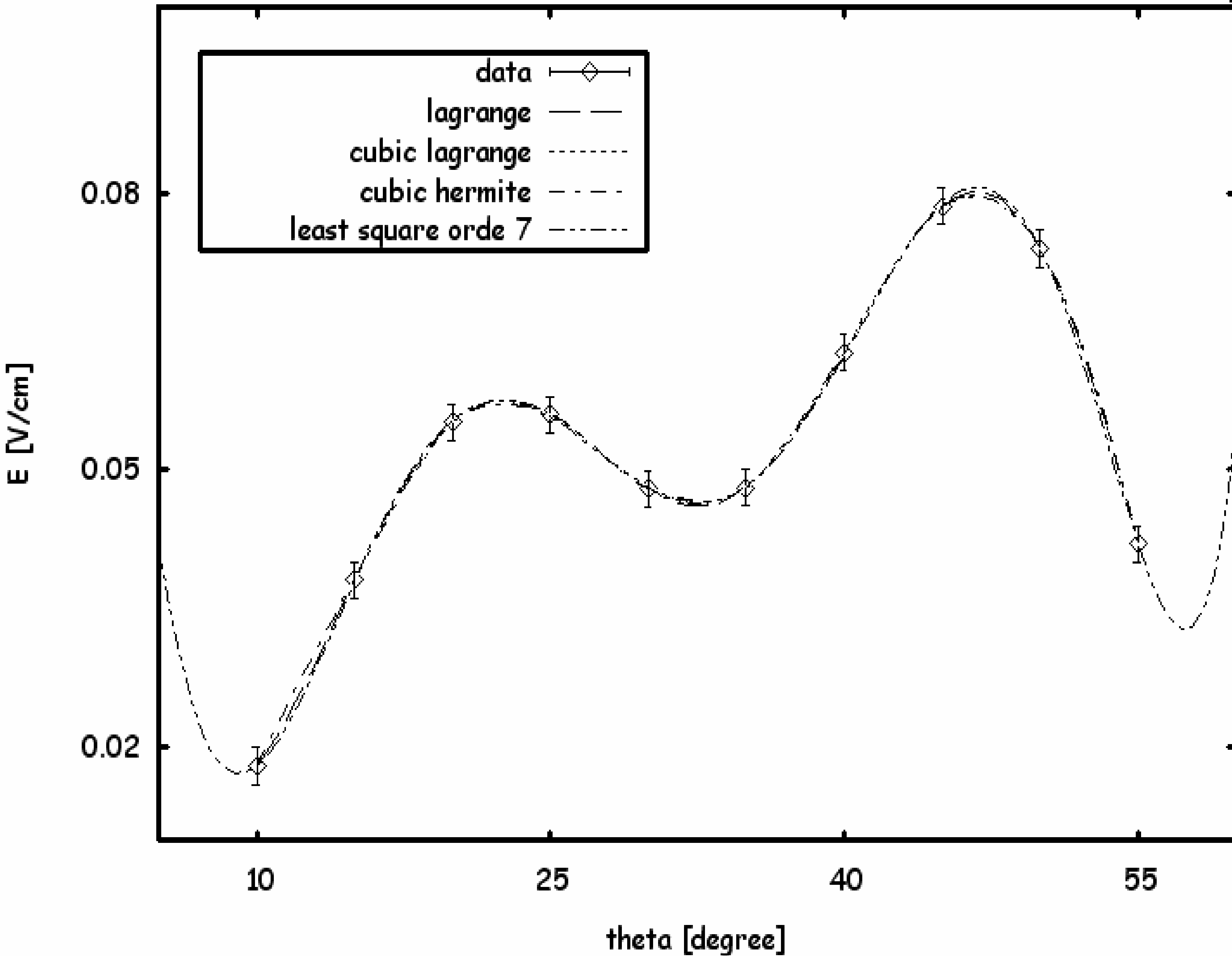
Kembali ke contoh problem least square:

Kuat medan listrik E di sekitar sebuah benda berbentuk lempeng diukur pada jarak 10 cm dari pusat massanya dan arah yang bervariasi. Arah dinyatakan dalam sudut θ terhadap sumbu y yang ditetapkan sebelum pengukuran. Diperoleh data sebagai berikut:

θ [derajat]	E [V/cm]
10	0.01794775
15	0.03808997
20	0.05516225
25	0.05598281
30	0.04795629
35	0.04807485
40	0.06273566
45	0.07853982
50	0.07395442
55	0.04201338

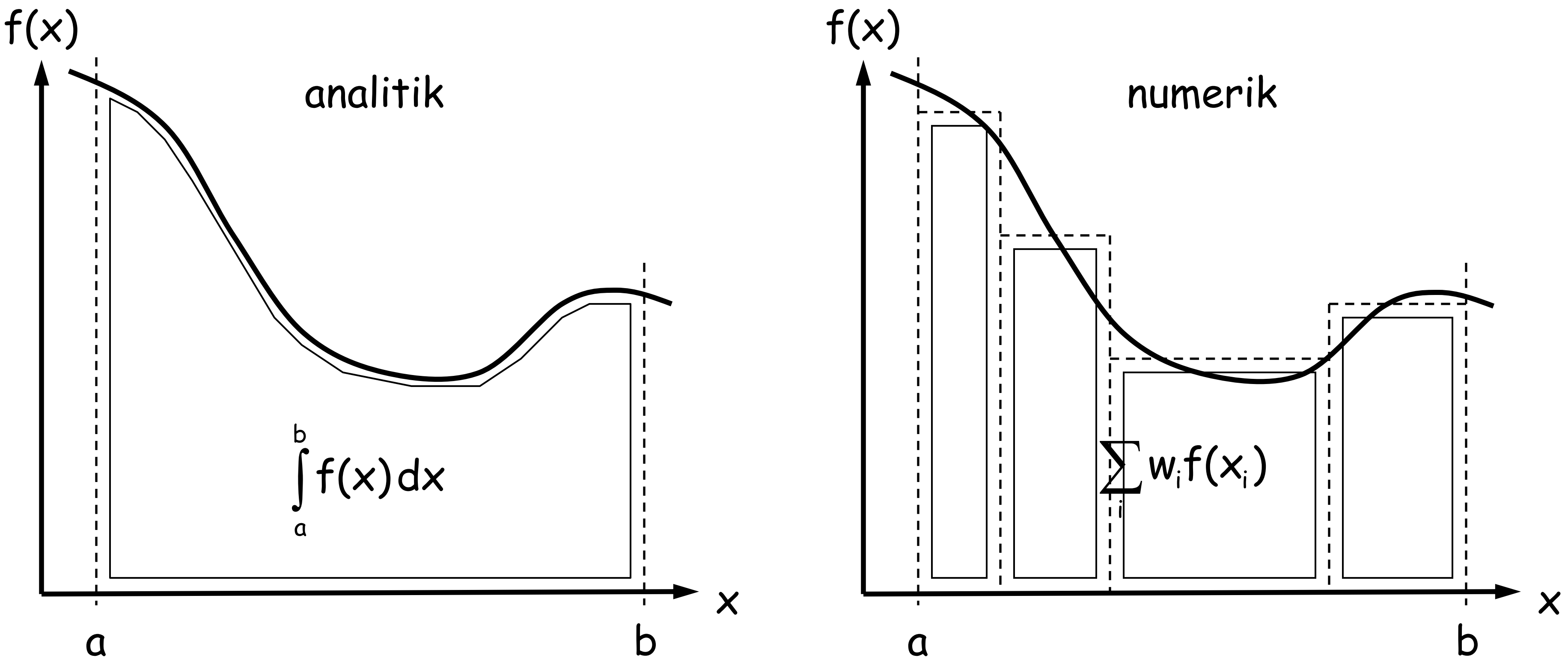


Dengan interpolasi, cari nilai p(x) di sepanjang titik data.


data
lagrange
cubic lagrange
cubic hermite
least square orde 7
0.08
0.05
0.02
E [V/cm]
10
25
40
55
theta [degree]

Integrasi

Menghitung luas daerah di bawah kurva:



$$\boxed{I = \int_a^b f(x)\,dx \cong \sum_{i=1}^{N} w_i f(x_i)} \longrightarrow$$

Integral numerik sering disebut juga sebagai quadrature; integrasi numerik disebut sebagai integrasi dgn menjumlah quadrature.

Meski tidak terlihat pada rumus akhir, pada integrasi numerik integrand f(x) diinterpolasi dengan suatu polinomial:

$$I = \int_a^b f(x)\,dx \cong \sum_{i=1}^{N} w_i f(x_i)$$

$$f(x) \cong p(x) \longleftarrow \text{polinomial}$$

Dilihat dari titik-titik x_i tempat integrand f(x) dihitung, ada teknik integrasi numerik yang menggunakan x_i berjarak tetap dan ada yang memakai x_i berjarak tidak tetap.

Contoh (akan dibahas):

- quadrature trapezoid, Simpson menggunakan x_i berjarak sama,
- quadrature Gaussian menggunakan x_i berjarak tidak sama.

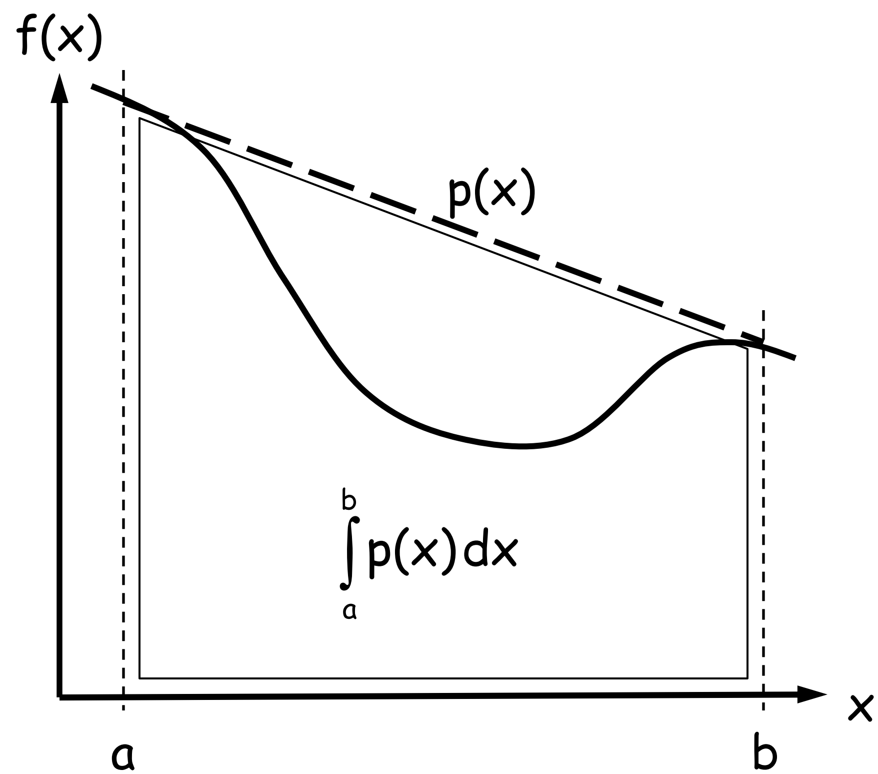
Quadrature Trapezoid

Kurva integrand f(x) diinterpolasi dengan sebuah garis lurus (f(x) diinterpolasi dengan fungsi linier / polinomial orde 1):

$$I = \int_a^b f(x)\,dx \cong \int_a^b p(x)\,dx = \sum_{i=1}^{N} w_i p(x_i), \qquad p(x) = r + sx$$

Untuk menarik garis lurus diperlukan minimal 2 titik, dipilih titik f(a) dan f(b):

$$p(a) = f(a), \;\; p(b) = f(b)$$



Dengan diketahui hanya p(a) dan p(b) (r dan s tidak dicari), maka integrasi numerik dikerjakan untuk N = 2:

$$\int_a^b p(x)\,dx = \sum_{i=1}^{2} w_i p(x_i) = w_1 p(x_1) + w_2 p(x_2) = w_1 p(a) + w_2 p(b) \longrightarrow w_1, w_2 = ?$$

Mencari w_1 dan w_2 :

$$p(x) = r + sx \longrightarrow \int_a^b (r+sx)\,dx = w_1(r+sa) + w_2(r+sb)$$

$$r(b-a) + \frac{1}{2}s(b^2 - a^2) = r(w_1 + w_2) + s(aw_1 + bw_2)$$

$$\downarrow$$

$$\begin{aligned} w_1 + \ \ w_2 &= b - a \\ aw_1 + bw_2 &= \frac{1}{2}(b^2 - a^2) \end{aligned} \longrightarrow w_1 = w_2 = \frac{1}{2}(b-a)$$

Rumus quadrature trapezoid: $\boxed{I = \int_a^b f(x)\,dx \cong \underbrace{\frac{h}{2}\big(f(a) + f(b)\big)}_{\text{luas trapezoid (lihat gambar)}}}$ $\quad (h = b - a)$

Quadrature Simpson & Boole

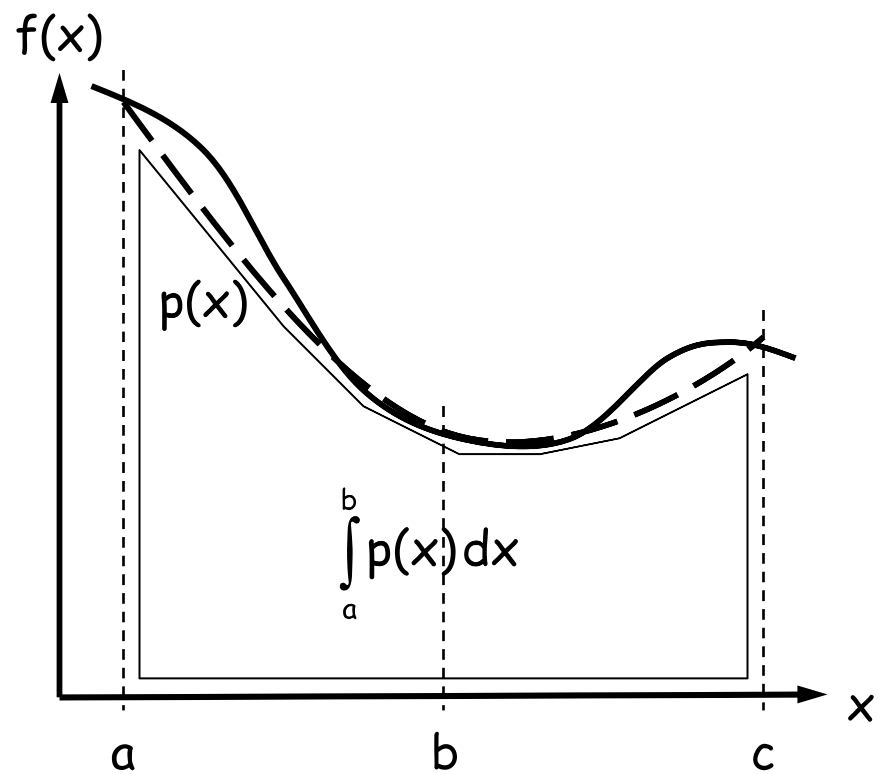
Cara yang sama seperti pada quadrature trapezoid bisa dipakai untuk polinomial p(x) orde lebih tinggi. Contoh, quadrature Simpson memakai p(x) fungsi kuadratik / polinomial orde 2 untuk menginterpolasi integrand f(x):

$$I=\int_a^c f(x)\,dx \cong \int_a^c p(x)\,dx=\sum_{i=1}^{N} w_i p(x_i), \qquad p(x)=r+sx+tx^2$$

Untuk membuat kurva kuadratik diperlukan minimal 3 titik, dipilih titik f(a), f(b) dan f(c):

$$p(a)=f(a),\ \ p(b)=f(b),$$
$$p(c)=f(c)$$

dengan $\quad b=\dfrac{a+c}{2}$



Integrasi numerik dikerjakan untuk N = 3:

$$\int_a^c p(x)\,dx = \sum_{i=1}^{3} w_i p(x_i) = w_1 p(a) + w_2 p(b) + w_3 p(c) \longrightarrow w_1, w_2, w_3 = ?$$

Mencari w_1, w_2, w_3:

$$p(x) = r + sx + tx^2$$

$$\int_a^c (r + sx + tx^2)\,dx = w_1(r + sa + ta^2) + w_2(r + sb + tb^2) + w_3(r + sc + tc^2)$$

$$r(c-a) + \frac{1}{2}s(c^2 - a^2) + \frac{1}{3}t(c^3 - a^3) = r(w_1 + w_2 + w_3) + s(aw_1 + bw_2 + cw_3) + t(a^2w_1 + b^2w_2 + c^2w_3)$$

$$\begin{aligned} w_1 + w_2 + w_3 &= c - a \\ aw_1 + bw_2 + cw_3 &= \frac{1}{2}(c^2 - a^2) \\ a^2w_1 + b^2w_2 + c^2w_3 &= \frac{1}{3}(c^3 - a^3) \end{aligned} \longrightarrow \begin{aligned} w_1 = w_3 &= \frac{1}{6}(c - a) \\ w_2 &= \frac{2}{3}(c - a) \end{aligned}$$

Diperoleh Rumus quadrature Simpson: $\boxed{I = \int_a^c f(x)\,dx \cong \frac{h}{3}\left(f(a) + 4f(b) + f(c)\right)}$

dengan $h = \frac{c-a}{2}$ yaitu jarak antar titik x_i tempat f(x) dihitung: $h = b - a = c - b$

Dengan cara yang sama, menggunakan p(x) polinomial orde 3 diperoleh rumus quadrature Simpson ⅜ :

$$\boxed{I = \int_a^d f(x)\,dx \cong \frac{3h}{8}\left(f(a) + 3f(b) + 3f(c) + f(d)\right)} \qquad \left(h = \frac{d-a}{3} = b - a = c - b = d - c\right)$$

dan dengan p(x) polinomial orde 4 rumus quadrature Boole:

$$\boxed{I = \int_a^e f(x)\,dx \cong \frac{2h}{45}\left(7f(a) + 32f(b) + 12f(c) + 32f(d) + 7f(e)\right)} \qquad \left(\begin{array}{lcl} h = \frac{e-a}{4} & = & b - a \\ & = & c - b \\ & = & d - c \\ & = & e - d \end{array}\right)$$

Integrasi Komposit

Polinomial orde rendah memadai untuk menginterpolasi sebuah fungsi dalam daerah yang sempit. Untuk daerah yang lebar diperlukan orde yang lebih tinggi. Alternatif lain yaitu, membagi daerah fungsi yang lebar itu dalam beberapa daerah yang sempit, lalu di tiap daerah yang sempit itu digunakan polinomial orde rendah untuk interpolasi.

Quadrature trapezoid dan Simpson pada dasarnya memadai untuk daerah integrasi yang sempit, namun dengan membagi daerah integrasi dalam beberapa daerah yang sempit, maka quadrature trapezoid dan Simpson bisa dipakai juga untuk daerah integrasi yang lebar. Integral total merupakan jumlah semua integral untuk daerah yang sempit. Integrasi seperti ini disebut integrasi komposit.

Bergantung pada integrand $f(x)$, daerah integrasi yang lebar bisa dibagi dalam beberapa daerah sempit yang sama atau berbeda panjang. Juga, semua integral untuk daerah yang sempit bisa dihitung menurut rumus quadrature yang sama, misal semuanya trapezoid, atau berbeda-beda, sesuai kurva di tiap daerah sempit itu. Kasus sederhana yaitu, bila daerah integrasi dibagi sama panjang dan untuk tiap daerah digunakan rumus quadrature yang sama.

Contoh, daerah integrasi [a,b] dibagi dalam N bagian sama panjang.

$$I = \int_a^b f(x)\,dx = \int_a^{a+d} f(x)\,dx + \int_{a+d}^{a+2d} f(x)\,dx + \ldots + \int_{b-2d}^{b-d} f(x)\,dx + \int_{b-d}^{b} f(x)\,dx \qquad \left(d = \frac{b-a}{N}\right)$$

- integrasi komposit menggunakan quadrature trapezoid

$$I = \int_a^b f(x)\,dx \cong h\left[\tfrac{1}{2}\left(f_0 + f_N\right) + f_1 + f_2 + \ldots + f_{N-1}\right]$$

$$h = \frac{b-a}{N}, \quad f_i = f(a+ih), \quad i = 0,\ldots,N$$

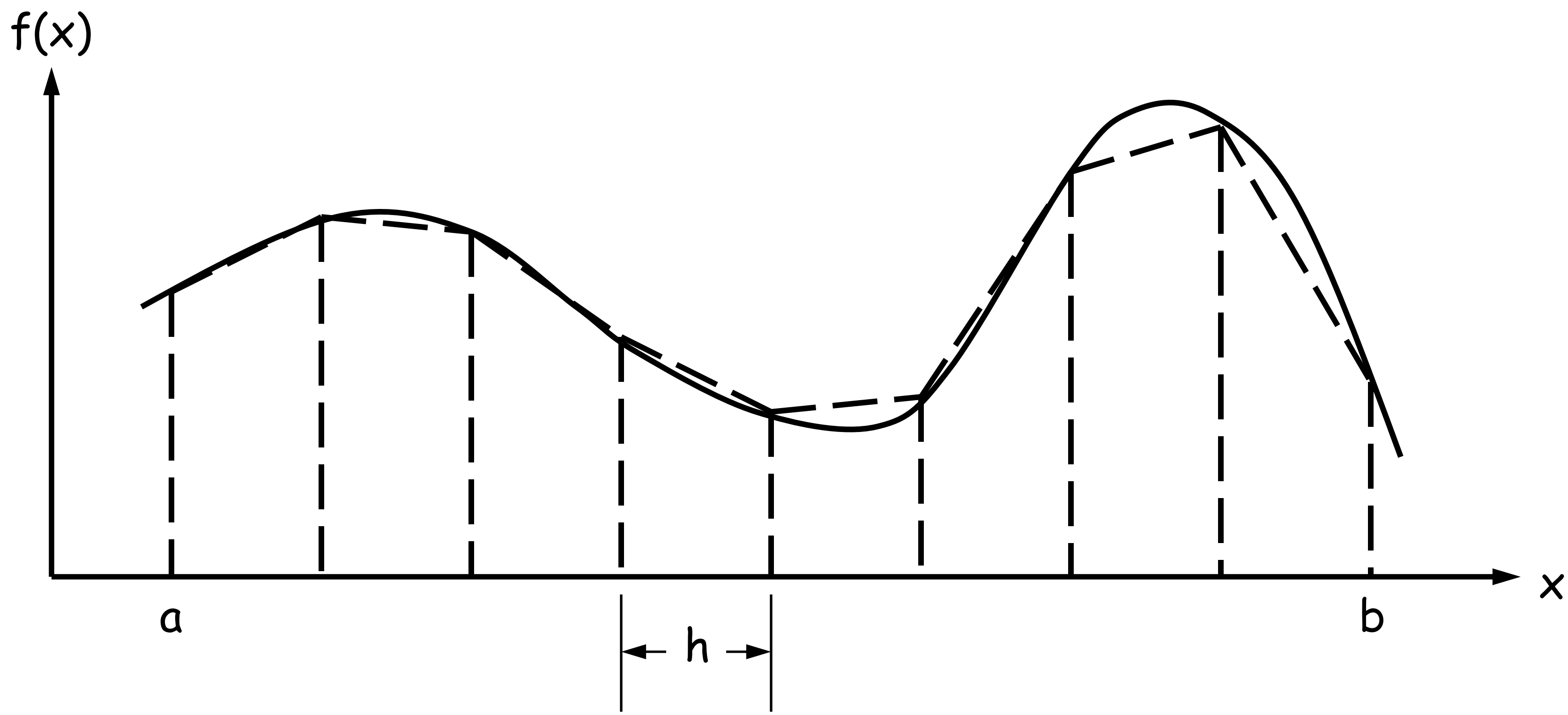
- integrasi komposit menggunakan quadrature Simpson

$$I = \int_a^b f(x)\,dx \cong \frac{2h}{3}\left[\tfrac{1}{2}\left(f_0 + f_{2N}\right) + 2\left(f_1 + f_3 + \ldots + f_{2N-1}\right) + f_2 + f_4 + \ldots + f_{2N-2}\right]$$

$$h = \frac{b-a}{2N}, \quad f_i = f(a+ih), \quad i = 0,\ldots,2N$$

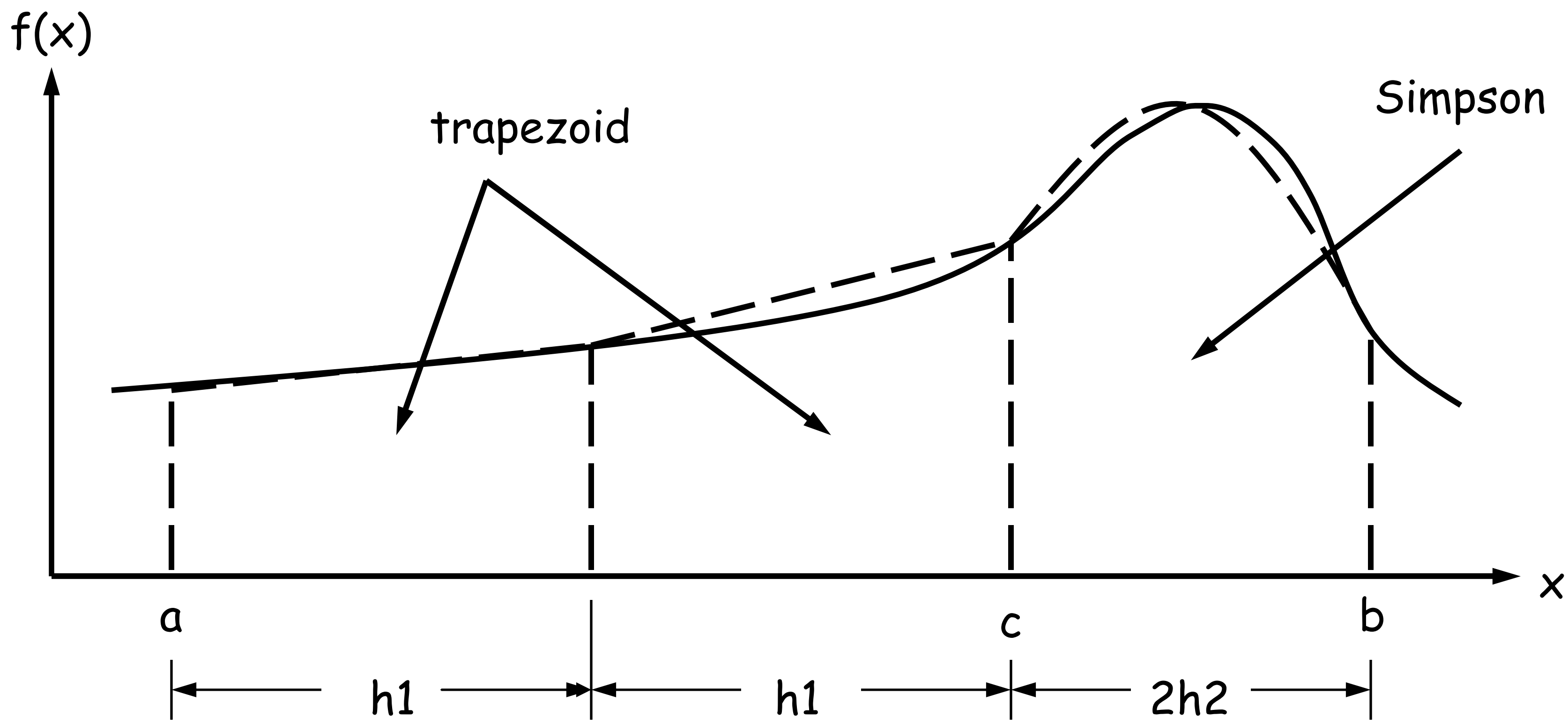
Integrasi komposit trapezoid untuk daerah integrasi [a,b] yang dibagi 8 sama panjang:

$$I = \int_a^b f(x)\,dx \cong h\left[\tfrac{1}{2}\left(f_0 + f_8\right) + f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7\right]$$



Integrasi komposit yang menggunakan quadrature trapezoid dan Simpson; daerah integrasi [a,b] yang dibagi 3:

$$I = \int_a^b f(x)\,dx \cong \frac{h1}{2}\left(f_a + 2f_{a+h1} + f_c\right) + \frac{h2}{3}\left(f_c + 4f_{c+h2} + f_b\right)$$



Quadrature Gaussian

Quadrature Gaussian memanfaatkan polinomial yang memiliki sifat orthogonal dan ternormalisasi sebagai berikut:

$$\int_a^b v(x) O_n(x) O_m(x)\, dx = \delta_{nm}, \qquad O_n(x) = \sum_{i=0}^{n} b_i x^i$$

Contoh:

$$O_n = \sqrt{\tfrac{2n+1}{2}} P_n, \quad P_n = \text{ polinomial Legendre} \longrightarrow \int_{-1}^{1} O_n(x) O_m(x)\, dx = \delta_{nm}$$

$$O_n = \tfrac{1}{n!} L_n, \quad L_n = \text{ polinomial Laguerre} \longrightarrow \int_0^\infty e^{-x} O_n(x) O_m(x)\, dx = \delta_{nm}$$

Dengan quadrature Gaussian, dievaluasi integral berbentuk:

$$\int_a^b v(x) f(x) dx = \sum_{i=1}^{N} w_i f(x_i) \longrightarrow w_i, x_i = ?$$

Mencari x_i :

Anggap integrand f(x) merupakan polinomial orde 2N-1 (atau katakan saja f(x) diinterpolasi dengan polinomial p(x) orde 2N-1):

$$f(x) \cong p(x) = \sum_{i=0}^{2N-1} a_i x^i = r(x) + s(x) \qquad \text{dengan} \qquad r(x) = \sum_{i=0}^{N-1} a_i x^i, \quad s(x) = \sum_{i=N}^{2N-1} a_i x^i$$

s(x) bisa ditulis sebagai $s(x) = q(x)O_N(x)$ dengan q(x) polinomial orde N-1:

$$q(x) = \sum_{i=0}^{N-1} d_i x^i = \sum_{i=0}^{N-1} c_i O_i(x)$$

Maka: $$\int_a^b v(x)s(x)dx = \sum_{i=0}^{N-1} c_i \int_a^b v(x)O_i(x)O_N(x)dx = \sum_{i=0}^{N-1} c_i \delta_{iN} = 0$$

Secara numerik: $$\int_a^b v(x)s(x)dx = \sum_{i=1}^{N} w_i s(x_i) = \sum_{i=1}^{N} w_i q(x_i) O_N(x_i) = 0$$

Mengingat q(x) fungsi sembarang, persamaan terakhir dipenuhi hanya jika $O_N(x_i) = 0 \;\; (i = 1, ..., N)$.

$$\longrightarrow \quad \boxed{x_i = \text{akar polinomial } O_N(x) \quad (i = 1, ..., N)}$$

Mencari w_i:

Untuk integrand f(x) dan s(x), yang merupakan polinomial orde 2N-1 berlaku integrasi numerik:

$$\int_a^b v(x)f(x)dx = \sum_{i=1}^{N} w_i f(x_i) \qquad \int_a^b v(x)s(x)dx = \sum_{i=1}^{N} w_i s(x_i)$$

Integrasi numerik yang sama tentu berlaku juga untuk integrand polinomial orde lebih rendah, contohnya r(x), yang berorde N-1:

$$\int_a^b v(x)r(x)dx = \sum_{i=1}^{N} w_i r(x_i), \qquad r(x) = \sum_{i=0}^{N-1} a_i x^i$$

Dari penurunan rumus quadrature trapezoid, Simpson dll sebelum ini diketahui bahwa untuk mencari w_i bisa digunakan r(x) sembarang polinomial orde N-1 (koefisien a_i tidak diperlukan). Karena itu, dipilih r(x) yang memudahkan:

$$r(x) = l(x, x_i) = \prod_{j \neq i} \left(\frac{x - x_j}{x_i - x_j} \right), \quad (i, j = 1, \ldots, N) \longrightarrow l(x_k, x_i) = \delta_{ik}$$

Diperoleh: $\int_a^b v(x) l(x, x_j) dx = \sum_{i=1}^{N} w_i l(x_i, x_j) = w_j \longrightarrow \boxed{w_j = \int_a^b v(x) l(x, x_j) dx \quad (j = 1, \ldots, N)}$

Pada integrasi numerik Gaussian, diperlukan N buah titik evaluasi x_i untuk integrand $f(x) \cong p(x)$ polinomial orde 2N-1.

Pada integrasi numerik seperti quadrature trapezoid dan Simpson, diperlukan 2N buah titik x_i untuk integrand $f(x) \cong p(x)$ polinomial orde 2N-1:

trapezoid	: 2N = 2
Simpson	: 2N = 3
Simpson $\frac{3}{8}$	: 2N = 4
Boole	: 2N = 5
dst	

Secara umum, dengan begitu, quadrature Gaussian memerlukan titik evaluasi lebih sedikit (separuh) dari yang diperlukan integrasi numerik yang mengikuti cara seperti quadrature trapezoid dan Simpson.

Bergantung pada keperluan, integrasi komposit juga bisa diterapkan menggunakan quadrature Gaussian atau campuran quadrature Gaussian dan yang lain.

Quadrature Gauss-Legendre

Quadrature Gauss-Legendre menggunakan polinomial Legendre P_n :

$$O_n = \sqrt{\tfrac{2n+1}{2}} P_n : \qquad \int_{-1}^{1} O_n(x) O_m(x)\, dx = \delta_{nm}$$

Asalnya, quadrature Gauss-Legendre dipakai untuk integral berbatas [-1,1]:

$$\int_{-1}^{1} f(x)dx = \sum_{i=1}^{N} w_i f(x_i)$$

Namun dengan mengganti variabel integrasi, quadrature Gauss-Legendre dapat juga dipakai untuk mengevaluasi integral dengan batas bukan [-1,1].

Contoh:

$$\int_{-1}^{1} f(x)dx = \sum_{i=1}^{N} w_i f(x_i) = \frac{2}{b-a} \int_{a}^{b} f(y)dy$$

$$\frac{y-a}{b-a} = \frac{x-(-1)}{1-(-1)} = \frac{x+1}{2}$$

(transformasi linier)

$$\boxed{\begin{aligned} \int_{a}^{b} f(y)dy &= \sum_{i=1}^{N} u_i f(y_i) \\ y_i &= \tfrac{1}{2}(x_i+1)(b-a)+a \\ u_i &= \left(\frac{b-a}{2}\right) w_i \end{aligned}}$$

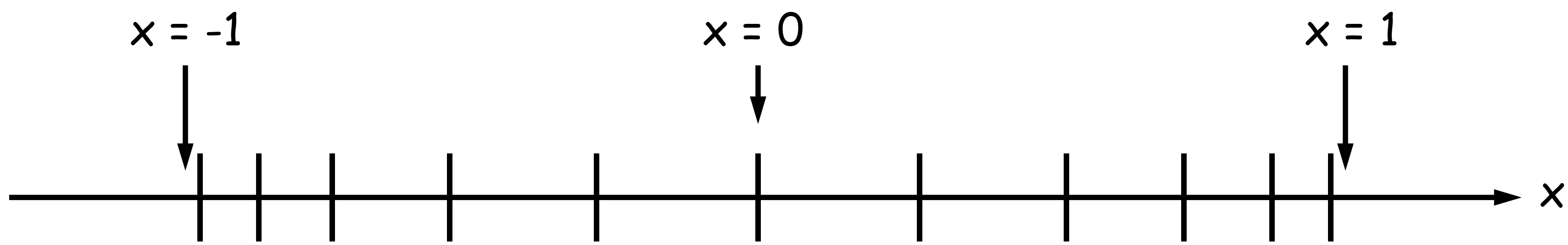
Contoh x_i dan w_i quadrature Gauss-Legendre untuk beberapa N terkecil:

$$\int_{-1}^{1} f(x)dx = \sum_{i=1}^{N} w_i f(x_i)$$

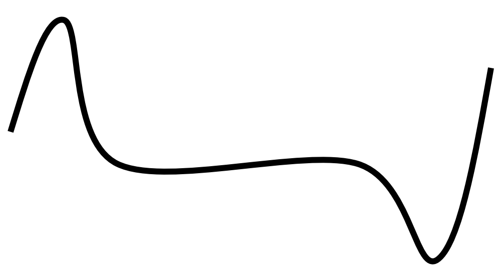
N	x	w
2	$\pm$ 0.577350269189626	1.000000000000000
3	$\pm$ 0.774596669241483	0.555555555555556
	0.000000000000000	0.888888888888889
4	$\pm$ 0.861136311594053	0.347854845137454
	$\pm$ 0.339981043584856	0.652145154862546
5	$\pm$ 0.906179845938664	0.236926885056189
	$\pm$ 0.538469310105683	0.478628670499367
	0.000000000000000	0.568888888888889

Distribusi x_i pada quadrature Gauss-Legendre tidak merata seperti distribusi pada quadrature trapezoid dan Simpson. Makin dekat ke batas-batas integral distribusi makin rapat. Distribusi itu simetris terhadap garis x = 0.

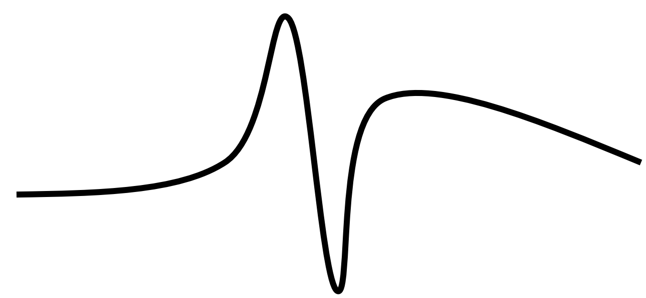
Ilustrasi untuk N = 11:



Distribusi ini lebih cocok untuk integrand f(x) yang bentuk kurvanya lebih tajam di sekitar batas integral, sementara kurang tajam di bagian tengah.



Untuk f(x) yang berkurva tajam di bagian tengah dan kurang tajam di sekitar batas integral diperlukan beberapa penanganan (mis. membagi daerah integrasi, redistribusi x dll).



Quadrature Gauss-Laguerre

Quadrature Gauss-Laguerre menggunakan polinomial Laguerre L_n :

$$O_n = \tfrac{1}{n!} L_n : \qquad \int_0^\infty e^{-x} O_n(x) O_m(x)\, dx = \delta_{nm}$$

dipakai untuk integral berbentuk: $\int_0^\infty e^{-x} f(x) dx = \sum_{i=1}^{N} w_i f(x_i)$

Contoh x_i dan w_i quadrature Gauss-Laguerre untuk beberapa N:

N	x	w
2	0.585786437626905	0.853553390593274
	3.414213562373095	0.146446609406726
4	0.322547689619392	0.603154104341634
	1.745761101158347	0.357418692437800
	4.536620296921128	0.038887908515005
	9.395070912301133	0.000539294705561

Lain-Lain

Mengganti Variabel Integrasi

Pada topik quadrature Gauss-Legendre terdapat contoh penggantian variabel integrasi. Penggantian variabel integrasi bisa juga diperlukan pada kasus lain. Tujuannya, agar evaluasi integral menjadi lebih mudah dan hasilnya baik.

Contoh: $I = \int_0^\infty \frac{dx}{1+x^2} \longleftarrow$ batas integral sampai tak behingga, jika dievaluasi langsung memerlukan sangat banyak titik, tidak praktis dan hasilnya bisa saja buruk

transformasi: $x = \frac{1+y}{1-y}, \quad dx = \frac{2}{(1-y)^2} dy$

$$\longrightarrow \quad I = \int_{-1}^{1} \frac{2dy}{(1-y)^2 \left(1 + \left(\frac{1+y}{1-y}\right)^2\right)}$$

$$= 2\int_{-1}^{1} \frac{dy}{(1-y)^2 + (1+y)^2} \longrightarrow$$ quadrature Gauss-Legendre

Meringkas Daerah Integrasi

Beberapa fungsi bersifat genap, ini memungkinkan daerah integrasi diringkas menjadi separuhnya (mengurangi jumlah titik evaluasi 2N menjadi N).

fungsi genap: $f(-x) = f(x)$ fungsi ganjil: $f(-x) = -f(x)$

Contoh:

- $$I = \int_{-a}^{a} \frac{dx}{1+x^2} = \sum_{i=1}^{2N} \frac{w_i}{1+x_i^2}$$
$$= 2\int_{0}^{a} \frac{dx}{1+x^2} = 2\sum_{i=N+1}^{2N} \frac{w_i}{1+x_i^2}$$

- $$I = \int_{-1}^{1} \frac{dy}{(1-y)^2+(1+y)^2} = \sum_{i=1}^{2N} \frac{w_i}{(1-y_i)^2+(1+y_i)^2}$$
$$= 2\int_{0}^{1} \frac{dy}{(1-y)^2+(1+y)^2} = 2\sum_{i=N+1}^{2N} \frac{w_i}{(1-y_i)^2+(1+y_i)^2}$$

Beberapa fungsi memiliki simetri, contoh fungsi trigonometri:

$$\sin(-x) = -\sin(x) \qquad \sin(\pi \pm x) = \mp \sin(x)$$
$$\cos(-x) = \cos(x) \qquad \cos(\pi \pm x) = -\cos(x)$$

Dengan memanfaatkan relasi simetri di atas batas integrasi sebuah integral tertutup (loop) seperti contoh di bawah dapat diringkas menjadi seperempatnya, sehingga jumlah titik evaluasi berkurang banyak:

$$I = \int_0^{2\pi} [f(\sin(x-a)) + f(\cos(x-a))] e^{im(x-a)} dx$$

integral tertutup bisa dimulai dari titik mana saja

$$\left.\begin{aligned}
&= \int_0^{2\pi} [f(\sin(x)) + f(\cos(x))] e^{imx} dx \\
&= \int_0^{\pi} \left[\{f(\sin(x)) + f(\cos(x))\} e^{imx} + \{f(-\sin(x)) + f(-\cos(x))\} e^{im(x+\pi)} \right] dx \\
&= \int_0^{\pi/2} \Big[f(\sin(x)) \left(e^{imx} + e^{im(\pi-x)}\right) + f(\cos(x)) \left(e^{imx} + e^{im(2\pi-x)}\right) \\
&\quad + f(-\sin(x)) \left(e^{im(\pi+x)} + e^{im(2\pi-x)}\right) + f(-\cos(x)) \left(e^{im(\pi+x)} + e^{im(\pi-x)}\right) \Big] dx
\end{aligned}\right\}$$

telah dipakai

$x = x' + \pi$

$x = -x'$

Menangani Singularitas

Kadang ditemui integrand f(x) yang memiliki singularitas dalam daerah integrasi. Salah satu cara menangani singularitas yaitu subtraksi, yang dimulai dengan menambahkan integral bernilai nol pada integral yang dihitung.

Contoh:

- $$I = \int_0^a \frac{dx}{(1+x)\sqrt{x}}$$ ← singular pada x = 0

$$= \int_0^a \frac{dx}{(1+x)\sqrt{x}} - \int_0^a \frac{dx}{\sqrt{x}} + \int_0^a \frac{dx}{\sqrt{x}}$$ ← ditambah nol

$$= \int_0^a \left(\frac{1}{(1+x)\sqrt{x}} - \frac{1}{\sqrt{x}} \right) dx + \int_0^a \frac{dx}{\sqrt{x}}$$ ← subtraksi pada integral asal

$$= -\int_0^a \frac{\sqrt{x}}{1+x} dx + 2\sqrt{a}$$

- $I=\int_0^a \frac{x^2 f(x)dx}{(b^2-x^2)} \qquad (0<b^2\le a^2)$ ← singular pada x = b

$= \int_0^a \frac{x^2 f(x)dx}{(b^2-x^2)} - \int_0^\infty \frac{b^2 f(b)dx}{(b^2-x^2)}$ ← ditambah nol (lihat *)

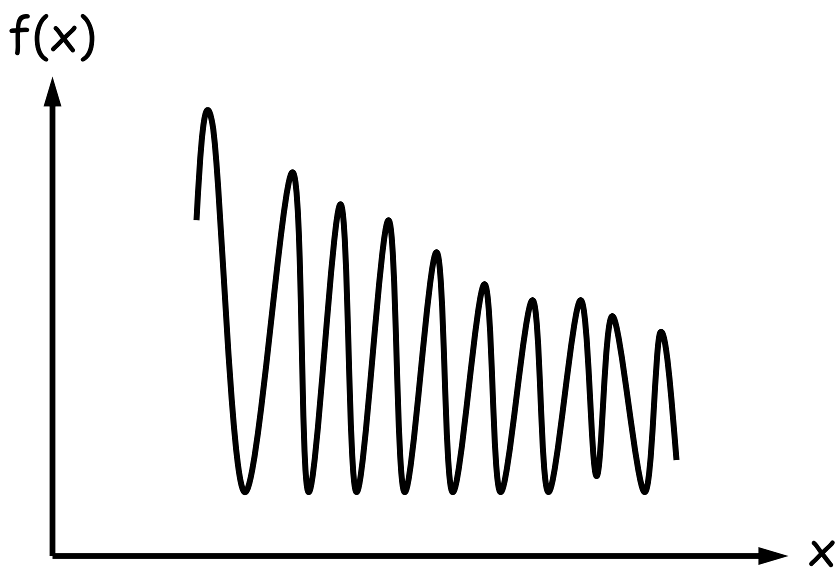
$= \int_0^a \frac{\left(x^2 f(x)-b^2 f(b)\right)dx}{(b^2-x^2)} - \int_a^\infty \frac{b^2 f(b)dx}{(b^2-x^2)}$ ← subtraksi pada integral asal

$= \int_0^a \frac{\left(x^2 f(x)-b^2 f(b)\right)dx}{(b^2-x^2)} - \frac{1}{2} b f(b) \ln\left(\frac{a-b}{a+b}\right)$

(*) $\int_0^\infty \frac{dx}{(b^2-x^2)} = \frac{1}{2b}\int_0^\infty \left(\frac{1}{b-x}+\frac{1}{b+x}\right)dx = \frac{1}{2b}\int_{-\infty}^\infty \frac{dx}{b+x} = \frac{1}{2b}\int_{-\infty}^\infty \frac{dx}{x} = 0$

Quadrature Filon

Bisa saja ditemui integrand f(x) yang sangat berosilasi; dalam jarak yang pendek f(x) berubah-ubah naik turun. Dengan macam-macam quadrature yang sudah disampaikan, integrasi menjadi sulit karena dibutuhkan banyak sekali titik evaluasi. Integral seperti ini dapat dihitung dengan menggunakan rumus quadrature Filon (M. Abramowitz & I. A. Stegun, Handbook of Mathematical Function, Dover Publications, Inc., NY, 1972).



Quadrature Filon (tanpa suku kesalahan, yang bisa diabaikan):

$$\int_a^b f(x)\cos(tx)dx = h\left[\alpha(th)(f_{2n}\sin(tb) - f_0\sin(ta)) + \beta(th)C_{genap} + \gamma(th)C_{ganjil}\right]$$

$$C_{genap} = \sum_{i=0}^{n} f_{2i}\cos(tx_{2i}) - \tfrac{1}{2}\left(f_{2n}\cos(tb) + f_0\cos(ta)\right)$$

$$C_{ganjil} = \sum_{i=1}^{n} f_{2i-1}\cos(tx_{2i-1})$$

$$\left(\begin{array}{l} h = \dfrac{b-a}{2n} = x_{i+1} - x_i \\ x_0 = a \\ f_j = f(x_j) \end{array}\right)$$

$$\int_a^b f(x)\sin(tx)dx = h\left[\alpha(th)(f_0\cos(ta) - f_{2n}\cos(tb)) + \beta(th)S_{genap} + \gamma(th)S_{ganjil}\right]$$

$$S_{genap} = \sum_{i=0}^{n} f_{2i}\sin(tx_{2i}) - \tfrac{1}{2}\left(f_{2n}\sin(tb) + f_0\sin(ta)\right)$$

$$S_{ganjil} = \sum_{i=1}^{n} f_{2i-1}\sin(tx_{2i-1})$$

$$\alpha(x) = \frac{1}{x} + \frac{\sin(2x)}{2x^2} - \frac{2\sin^2 x}{x^3}$$

$$\beta(x) = 2\left(\frac{1+\cos^2 x}{x^2} - \frac{\sin(2x)}{x^3}\right)$$

$$\gamma(x) = 4\left(\frac{\sin x}{x^3} - \frac{\cos x}{x^2}\right)$$

untuk nilai x kecil:

$$\alpha(x) = \frac{2x^3}{45} - \frac{2x^5}{315} + \frac{2x^7}{4725} - \ldots$$

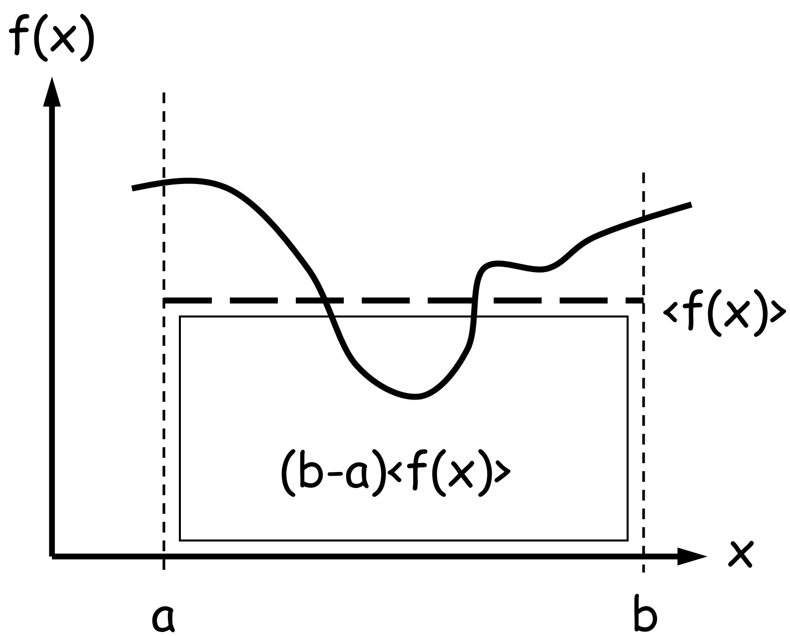
$$\beta(x) = \frac{2}{3} + \frac{2x^2}{15} - \frac{4x^4}{105} + \frac{2x^6}{567} - \ldots$$

$$\gamma(x) = \frac{4}{3} - \frac{2x^2}{15} + \frac{x^4}{210} - \frac{x^6}{11340} + \ldots$$

Integrasi Monte Carlo

Mungkin saja cara-cara integrasi numerik yang sudah disampaikan sulit atau tidak bisa diterapkan untuk mengevaluasi suatu integral. Pada keadaan ini, integrasi Monte Carlo dapat dipilih.

Integrasi Monte Carlo tidak menggunakan interpolasi seperti pada cara-cara integrasi numerik sebelum ini. Integral dianggap sebagai satu persegi panjang, dengan lebar daerah integrasi dan tinggi nilai rata-rata integrand f(x), yang diperoleh melalui statistik dengan memanfaatkan bilangan acak:



$$\langle f(x) \rangle = \frac{1}{n} \sum_{i=1}^{n} f(x_i)$$

$$x_i = \text{bilangan acak}: \quad a \le x_i \le b$$

$$\boxed{I = \int_a^b f(x)dx \cong (b-a)\frac{1}{n}\sum_{i=1}^{n} f(x_i)}$$

Persamaan Differensial

Persamaan differensial (PD) yang dibahas meliputi persamaan differensial biasa dan persamaan differensial parsial.

Beberapa persamaan differensial merupakan juga persamaan eigenvalue, contoh persamaan untuk senar gitar (gelombang berdiri). Karena itu, akan dibahas juga persamaan eigenvalue.

Persamaan Differensial Biasa

Pada bagian ini disampaikan metode numerik untuk menyelesaikan persamaan differensial biasa orde 1 dan 2. Dua masalah yang akan dibahas yaitu:

- PD dengan syarat awal
- PD dengan syarat batas

PD dengan Syarat Awal

PD Orde 1

Bentuk umum PD orde 1: $y' = \frac{dy}{dx} = f(x,y)$

Diketahui: $y(x_0) = y_0 \longrightarrow y(x) = ?$

Integrasi: $\int_{y_0}^{y} dy = \int_{x_0}^{x} f(x,y)dx$

$$\downarrow$$

$$y(x) = y_0 + \int_{x_0}^{x} f(x,y)dx$$

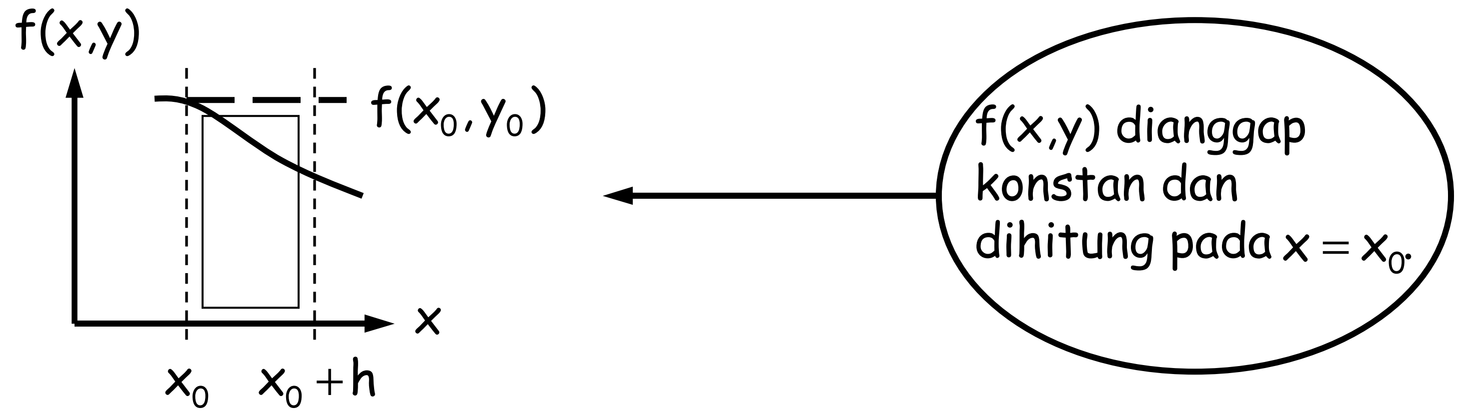
Masalah persamaan differensial berubah menjadi masalah persamaan integral.

Dicari y(x) pada titik $x = x_0 + h$: $\boxed{y(x_0 + h) = y_0 + \int_{x_0}^{x_0+h} f(x,y)dx}$

Setelah $y(x_0 + h)$ didapat, selanjutnya dicari $y(x_0 + 2h)$. Demikian seterusnya.

Metode Euler

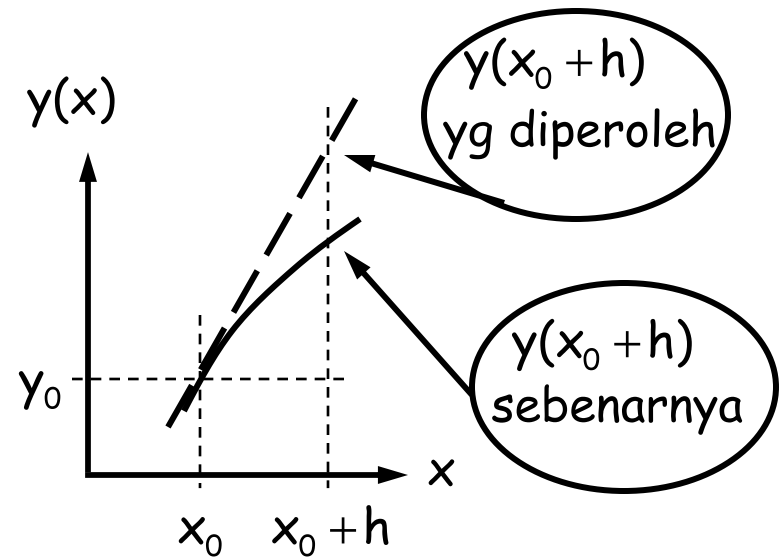
Menurut metode Euler:



Diperoleh:

$$y(x_0+h) \cong y_0 + f(x_0,y_0)\int_{x_0}^{x_0+h} dx$$

$$\cong y_0 + hf(x_0,y_0)$$



Metode Euler yang Dimodifikasi

Modifikasi dilakukan dalam memilih nilai f(x,y) yang dianggap konstan. Dipilih f(x,y) pada titik $x = x_0 + \frac{1}{2}h$:

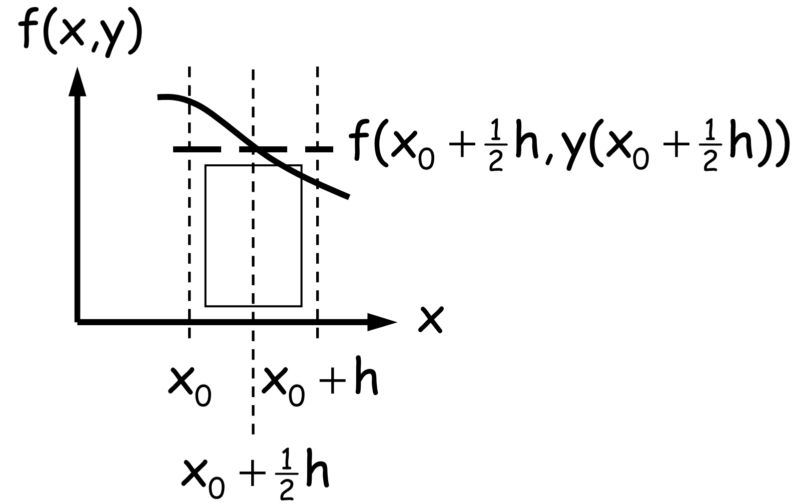
$$f(x_0 + \tfrac{1}{2}h, y(x_0 + \tfrac{1}{2}h))$$

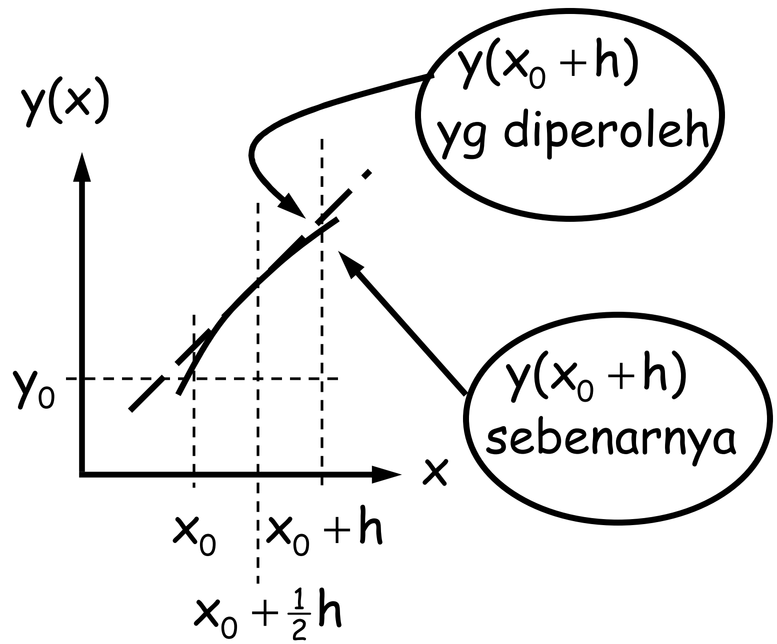
dengan $y(x_0 + \frac{1}{2}h)$ dihitung memakai metode Euler:

$$y(x_0 + \tfrac{1}{2}h) \cong y_0 + \tfrac{1}{2}hf(x_0, y_0)$$

Diperoleh:

$$\boxed{\begin{aligned} y(x_0 + h) &\cong y_0 + hf(x_0 + \tfrac{1}{2}h, y(x_0 + \tfrac{1}{2}h)) \\ &\cong y_0 + hf(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}hf(x_0, y_0)) \end{aligned}}$$

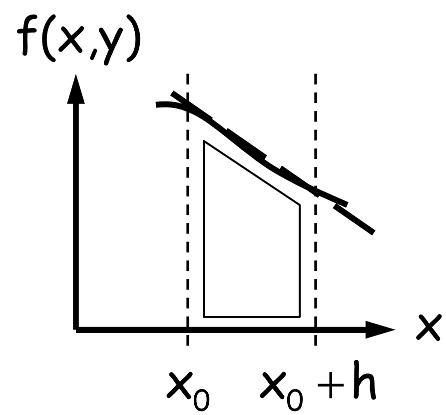




Metode Euler yang Lebih Baik (Improved)

Kali ini dipakai nilai f(x,y) yang merupakan rata-rata dari dua nilai f(x,y), masing-masing pada titik x_0 dan $x_0 + h$:

$$\tfrac{1}{2}\left[f(x_0, y_0) + f(x_0 + h, y(x_0 + h))\right]$$



Ini sama dengan menggunakan quadrature trapezoid untuk mengevaluasi integral:

$$\int_{x_0}^{x_0+h} f(x,y)dx \cong \tfrac{1}{2}h\left[f(x_0, y_0) + f(x_0 + h, y(x_0 + h))\right]$$

dengan $y(x_0 + h)$ dihitung memakai metode Euler:

$$y(x_0 + h) \cong y_0 + hf(x_0, y_0)$$

Diperoleh:

$$\boxed{\begin{aligned} y(x_0 + h) &\cong y_0 + \tfrac{1}{2}h\left[f(x_0, y_0) + f(x_0 + h, y(x_0 + h))\right] \\ &\cong y_0 + \tfrac{1}{2}h\left[f(x_0, y_0) + f(x_0 + h, y_0 + hf(x_0, y_0))\right] \end{aligned}}$$

Metode Runge-Kutta

Metode Euler dan variasinya sebelum ini sebetulnya termasuk metode Runge-Kutta, yang menyatakan solusi PD y(x) dalam turunannya f(x,y), yang dihitung untuk argumen x,y yang bervariasi. Sebuah metode Runge-Kutta disebut berorde n jika memiliki suku koreksi $O(h^{n+1})$ (diperoleh dari ekspansi Taylor):

$$y(x_0+h)=y_{diperoleh}+O(h^{n+1})$$

Menurut hal itu, metode Euler merupakan metode Runge-Kutta orde 1 sedangkan metode Euler yang dimodifikasi dan yang lebih baik (improved) merupakan metode Runge-Kutta orde 2:

Euler : $y(x_0+h)=y_0+hf(x_0,y_0)+O(h^2)$

Euler yg dimodifikasi: $y(x_0+h)=y_0+hf(x_0+\frac{1}{2}h,y_0+\frac{1}{2}hf(x_0,y_0))+O(h^3)$

Euler yg lebih baik : $y(x_0+h)=y_0+\frac{1}{2}h\left[f(x_0,y_0)+f(x_0+h,y_0+hf(x_0,y_0))\right]+O(h^3)$

Metode Runge-Kutta yang paling banyak digunakan orang yaitu berorde 4, yang sering diingat sebagai metode Runge-Kutta tanpa tambahan keterangan 'orde 4'.

Untuk mendapatkan rumus metode Runge-Kutta orde 4, orang bisa memulai dengan mengevaluasi integral f(x,y) memakai quadrature Simpson:

$$\int_{x_0}^{x_0+h} f(x,y)dx \cong \tfrac{1}{6}h\left[f(x_0,y_0)+4f(x_0+\tfrac{1}{2}h,y(x_0+\tfrac{1}{2}h))+f(x_0+h,y(x_0+h))\right]$$

$$\cong \tfrac{1}{6}h\left(f_0+2f_1+2f_2+f_3\right)$$

dengan:

$$f_0 = f(x_0,y_0) \qquad f_2 = f(x_0+\tfrac{1}{2}h,y(x_0+\tfrac{1}{2}h))$$

$$f_1 = f(x_0+\tfrac{1}{2}h,y(x_0+\tfrac{1}{2}h)) \qquad f_3 = f(x_0+h,y(x_0+h))$$

f_1 dan f_2 memiliki nilai berbeda, karena dihitung untuk nilai argumen $y(x_0+\tfrac{1}{2}h)$ yang berbeda: menurut metode Euler, $y(x_0+\tfrac{1}{2}h)$ dapat diperoleh melalui 2 persamaan:

(1) $y(x_0+\tfrac{1}{2}h) \cong y_0+\tfrac{1}{2}hf_0 \longrightarrow f_1 = f(x_0+\tfrac{1}{2}h,y_0+\tfrac{1}{2}hf_0)$

atau

(2) $y_0 \cong y(x_0+\tfrac{1}{2}h)-\tfrac{1}{2}hf(x_0+\tfrac{1}{2}h,y(x_0+\tfrac{1}{2}h))$

$\cong y(x_0+\tfrac{1}{2}h)-\tfrac{1}{2}hf(x_0+\tfrac{1}{2}h,y_0+\tfrac{1}{2}hf_0)$

$\longrightarrow y(x_0+\tfrac{1}{2}h) \cong y_0+\tfrac{1}{2}hf_1 \longrightarrow f_2 = f(x_0+\tfrac{1}{2}h,y_0+\tfrac{1}{2}hf_1)$

Untuk f_3 , digunakan metode Euler yang dimodifikasi untuk mencari $y(x_0+h)$:

$$\begin{aligned} y(x_0+h) &\cong y_0 + hf(x_0+\tfrac{1}{2}h, y(x_0+\tfrac{1}{2}h)) \\ &\cong y_0 + hf(x_0+\tfrac{1}{2}h, y_0+\tfrac{1}{2}hf_1) \\ &\cong y_0 + hf_2 \longrightarrow f_3 = f(x_0+h, y_0+hf_2) \end{aligned}$$

Jadi, menurut metode Runge-Kutta orde 4:

$$\boxed{y(x_0+h) = y_0 + \tfrac{1}{6}h\left(f_0 + 2f_1 + 2f_2 + f_3\right)}$$

dengan:

$$\boxed{\begin{array}{ll} f_0 = f(x_0, y_0) & f_2 = f(x_0+\tfrac{1}{2}h, y_0+\tfrac{1}{2}hf_1) \\ f_1 = f(x_0+\tfrac{1}{2}h, y_0+\tfrac{1}{2}hf_0) & f_3 = f(x_0+h, y_0+hf_2) \end{array}}$$

Berangkat dengan quadrature Simpson, orang juga bisa memperoleh rumus metode Runge-Kutta orde 3:

$$\int_{x_0}^{x_0+h} f(x,y)dx \cong \tfrac{1}{6}h\left(f_0 + 4f_1 + f_2\right)$$

dengan: $f_0 = f(x_0, y_0)$ $f_1 = f(x_0 + \tfrac{1}{2}h, y(x_0 + \tfrac{1}{2}h))$ $f_2 = f(x_0 + h, y(x_0 + h))$

$y(x_0 + \tfrac{1}{2}h)$ dicari dengan metode Euler dan $y(x_0 + h)$ dengan metode Euler yang dimodifikasi:

$$y(x_0 + \tfrac{1}{2}h) \cong y_0 + \tfrac{1}{2}hf_0 \longrightarrow f_1 = f(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}hf_0)$$

$$y(x_0 + h) \cong y_0 + hf_1 \longrightarrow f_2 = f(x_0 + h, y_0 + hf_1)$$

Jadi, menurut metode Runge-Kutta orde 3:

$$\boxed{\begin{aligned} y(x_0 + h) &= y_0 + \tfrac{1}{6}h\left(f_0 + 4f_1 + f_2\right) \\ f_0 &= f(x_0, y_0) \\ f_1 &= f(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}hf_0) \\ f_2 &= f(x_0 + h, y_0 + hf_1) \end{aligned}}$$

PD Orde 2

Bentuk umum PD orde 2: $y'' = \frac{d^2y}{dx^2} = f(x, y, y')$

Diketahui: $y(x_0) = y_0, \quad y'(x_0) = y'_0 \longrightarrow y(x) = ?$

Definisikan fungsi baru u:

$$\begin{aligned} u &= y' \\ u_0 &= y'_0 \end{aligned} \longrightarrow \begin{aligned} y' &= u(x, y) \\ u' &= f(x, y, u) \end{aligned}$$

Masalah PD orde 2
berubah menjadi
masalah PD orde 1.

Contoh penyelesaian dengan metode Euler yang lebih baik (improved):

$u' = f(x,y,u)$	$y' = u(x,y)$
$u(x_0+h) = u_0 + \frac{1}{2}h(f_0 + f_1)$	$y(x_0+h) = y_0 + \frac{1}{2}h(u_0 + u_1)$
$f_0 = f(x_0, y_0, u_0)$	$u_0 = y'_0$
$f_1 = f(x_0+h, y_0+hu_0, u_1)$	$u_1 = u_0 + hf_0$

Alur perhitungan:

$$
\begin{tikzcd}
y_0,\ u_0 \arrow[r] & f_0 \arrow[r] & u_1 \arrow[r] & f_1,\ y(x_0+h),\ u(x_0+h) \arrow[d] \\
 & x_0+h \rightarrow x_0,\ u(x_0+h) \rightarrow u_0,\ y(x_0+h) \rightarrow y_0 \arrow[ul] & & \arrow[ll]
\end{tikzcd}
$$

Contoh penyelesaian dengan metode Runge-Kutta orde 4:

$$u' = f(x,y,u) \qquad\qquad y' = u(x,y)$$

$$u(x_0+h) = u_0 + \tfrac{1}{6}h\left(f_0 + 2f_1 + 2f_2 + f_3\right) \qquad y(x_0+h) = y_0 + \tfrac{1}{6}h\left(u_0 + 2u_1 + 2u_2 + u_3\right)$$

$$f_0 = f(x_0, y_0, u_0) \qquad u_0 = y'_0$$

$$f_1 = f(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}hu_0, u_1) \qquad u_1 = u_0 + \tfrac{1}{2}hf_0$$

$$f_2 = f(x_0 + \tfrac{1}{2}h, y_0 + \tfrac{1}{2}hu_1, u_2) \qquad u_2 = u_0 + \tfrac{1}{2}hf_1$$

$$f_3 = f(x_0 + h, y_0 + hu_2, u_3) \qquad u_3 = u_0 + hf_2$$

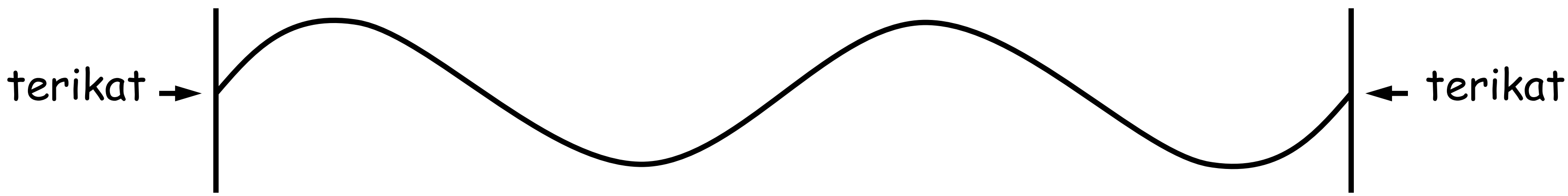
Alur perhitungan:

$$y_0,\ u_0 \longrightarrow f_0 \longrightarrow u_1 \longrightarrow f_1 \longrightarrow u_2 \longrightarrow f_2 \longrightarrow u_3 \longrightarrow f_3,\ y(x_0+h),\ u(x_0+h)$$

$$x_0 + h \to x_0,\ u(x_0+h) \to u_0,\ y(x_0+h) \to y_0$$

PD dengan Syarat Batas

Contoh, gelombang yang merambat di sepanjang tali bisa digambarkan dengan PD orde 2. Jika ujung-ujung tali itu diikat sehingga tidak bisa bergerak, maka kita temui kasus PD dengan syarat batas.



Bentuk umum PD orde 1 & 2 linear: (1) $y' = f(x,y) = d(x) - e(x)y$

(2) $y'' = g(x,y,y') = a(x) - b(x)y - c(x)y'$

Diketahui:

$$x_0 \le x \le x_n$$
$$y(x_0) = y_0 \longrightarrow y(x) = ?$$
$$y(x_n) = y_n$$

Dicari $y_i = y(x_i)$ pada titik $x_i = x_0 + ih$ $(i = 1,...,n-1)$ dengan $h = \dfrac{x_n - x_0}{n}$.

Metode Finite Differences

(1) $y' + e(x)y = d(x)$

(2) $y'' + c(x)y' + b(x)y = a(x)$

$$y'_i \cong \frac{y_{i+1} - y_{i-1}}{2h}$$

$$y''_i \cong \frac{y_{i+1} - 2y_i + y_{i-1}}{h^2}$$

(1) $y'_i + e_i y_i = d_i$

(2) $y''_i + c_i y'_i + b_i y_i = a_i$

(1) $\dfrac{y_{i+1} - y_{i-1}}{2h} + e_i y_i \cong d_i$

(2) $\dfrac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + c_i \dfrac{y_{i+1} - y_{i-1}}{2h} + b_i y_i \cong a_i$

Jadi, pada akhirnya ditemui masalah sistem persamaan linear:

$$\boxed{\begin{array}{l} (1)\quad -y_{i-1} + 2e_i h y_i + y_{i+1} \cong 2d_i h \\ (2)\quad \left(1 - \dfrac{c_i h}{2}\right) y_{i-1} - \left(2 - b_i h^2\right) y_i + \left(1 + \dfrac{c_i h}{2}\right) y_{i+1} \cong a_i h^2 \end{array}}$$

yang dapat diselesaikan menggunakan metode, contoh, iterasi Jacobi dan Gauss-Siedel.

Aplikasi Iterasi Jacobi dan Gauss-Siedel pada PD dengan Syarat Batas

PD orde 1: (1) $y' = d(x) - e(x)y \longrightarrow -y_{i-1} + 2e_i h y_i + y_{i+1} \cong 2d_i h$

PD orde 2: (2) $y'' = a(x) - b(x)y - c(x)y' \rightarrow \left(1 - \frac{c_i h}{2}\right) y_{i-1} - \left(2 - b_i h^2\right) y_i + \left(1 + \frac{c_i h}{2}\right) y_{i+1} \cong a_i h^2$

Iterasi Jacobi:

$$\boxed{\begin{aligned} &(1)\quad y_i^{(k)} \cong \frac{1}{2e_i h}\left(2d_i h + y_{i-1}^{(k-1)} - y_{i+1}^{(k-1)}\right) \\ &(2)\quad y_i^{(k)} \cong \frac{1}{\left(2 - b_i h^2\right)}\left(-a_i h^2 + \left(1 - \frac{c_i h}{2}\right) y_{i-1}^{(k-1)} + \left(1 + \frac{c_i h}{2}\right) y_{i+1}^{(k-1)}\right) \end{aligned}}$$

Iterasi Gauss-Siedel (contoh untuk i membesar, i = 1, …, n-1):

$$\boxed{\begin{aligned} &(1)\quad y_i^{(k)} \cong \frac{1}{2e_i h}\left(2d_i h + y_{i-1}^{(k)} - y_{i+1}^{(k-1)}\right) \\ &(2)\quad y_i^{(k)} \cong \frac{1}{\left(2 - b_i h^2\right)}\left(-a_i h^2 + \left(1 - \frac{c_i h}{2}\right) y_{i-1}^{(k)} + \left(1 + \frac{c_i h}{2}\right) y_{i+1}^{(k-1)}\right) \end{aligned}}$$

Catatan, sesuai syarat batas: $\boxed{y_0^{(k)} = y_0 \qquad y_n^{(k)} = y_n}$

Persamaan Differensial Parsial

Pada bagian ini disampaikan metode numerik untuk menyelesaikan persamaan differensial parsial 2 dimensi tipe eliptik, parabolik dan hiperbolik.

Persamaan Differensial Eliptik

Bentuk umum PD eliptik: $\nabla^2\psi(\vec{r}) = -4\pi\rho(\vec{r})$

Untuk kasus 2 dimensi: $\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\psi(x,y) = -4\pi\rho(x,y)$

Gunakan metode finite differences:

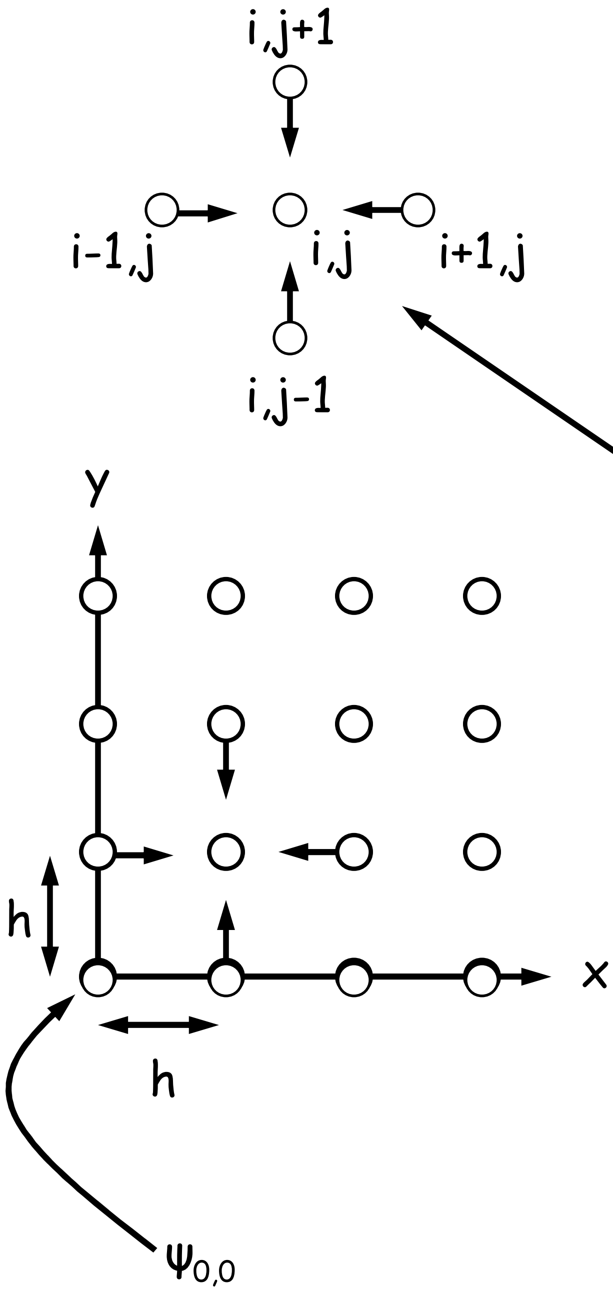
$$\left.\frac{\partial^2}{\partial x^2}\psi(x,y)\right|_{x_i,y_j} \approx \frac{\psi(x_{i+1},y_j) - 2\psi(x_i,y_j) + \psi(x_{i-1},y_j)}{h^2} = \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{h^2}$$

$$\left.\frac{\partial^2}{\partial y^2}\psi(x,y)\right|_{x_i,y_j} \approx \frac{\psi(x_i,y_{j+1}) - 2\psi(x_i,y_j) + \psi(x_i,y_{j-1})}{h^2} = \frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{h^2}$$

$$(h = x_{i+1} - x_i = y_{j+1} - y_j)$$

Diperoleh: $\boxed{\psi_{i,j} = h^2\pi\rho_{i,j} + \tfrac{1}{4}\left[\psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1}\right]}$

Dicari distribusi spasial ψ .



Langkah:

1. Buat grid pada bidang xy, dengan jarak terdekat antar titik h.
2. (Dianggap nilai pada batas-batas bidang xy diketahui.)

   Hitung dengan rumus :

   $$\psi_{i,j} = h^2 \pi \rho_{i,j} + \tfrac{1}{4}\left[\psi_{i+1,j} + \psi_{i-1,j} + \psi_{i,j+1} + \psi_{i,j-1}\right]$$

   secara berurutan $\psi_{i,j}$ untuk i = 1 & j = 1, 2, 3, ..., lalu i = 2 & j = 1, 2, 3, ..., i = 3 & j = 1, 2, 3, ... dan seterusnya.
3. Jika dalam langkah 2 ditemui nilai $\psi_{i,j}$ yang belum diketahui, gunakan nilai tebakan.
4. Ulangi langkah 2 – 3 sampai dicapai kestabilan untuk nilai $\psi_{i,j}$ di semua titik:

$$\left|\psi_{i,j}^{(\text{iterasi sebelum})} - \psi_{i,j}^{(\text{iterasi berikutnya})}\right| < \varepsilon \text{ (bilangan kecil)}$$

Persamaan Differensial Parabolik

Bentuk umum PD parabolik: $$\left(\nabla^2 - \frac{1}{\Gamma}\frac{\partial}{\partial t}\right)\psi(\vec{r},t) = -4\pi\rho(\vec{r},t)$$

Untuk kasus 2 dimensi: $$\left(\frac{\partial^2}{\partial x^2} - \frac{1}{\Gamma}\frac{\partial}{\partial t}\right)\psi(x,t) = -4\pi\rho(x,t)$$

Gunakan metode finite differences:

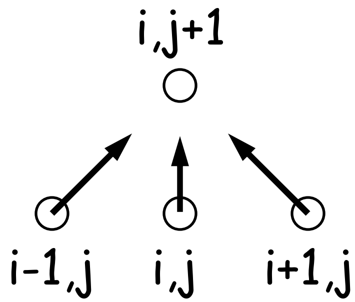
$$\frac{\partial^2}{\partial x^2}\psi(x,t)\bigg|_{x_i,t_j} \approx \frac{\psi(x_{i+1},t_j) - 2\psi(x_i,t_j) + \psi(x_{i-1},t_j)}{h_x^2} = \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{h_x^2}$$

$$\frac{\partial}{\partial t}\psi(x,t)\bigg|_{x_i,t_j} \approx \frac{\psi(x_i,t_{j+1}) - \psi(x_i,t_j)}{h_t} = \frac{\psi_{i,j+1} - \psi_{i,j}}{h_t}$$

$$(h_x = x_{i+1} - x_i,\ h_t = t_{j+1} - t_j)$$

Diperoleh: $$\boxed{\psi_{i,j+1} = 4\Gamma h_t \pi \rho_{i,j} + \psi_{i,j} + \frac{\Gamma h_t}{h_x^2}\left(\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}\right)}$$

Untuk tiap posisi dicari perubahan ψ terhadap waktu.



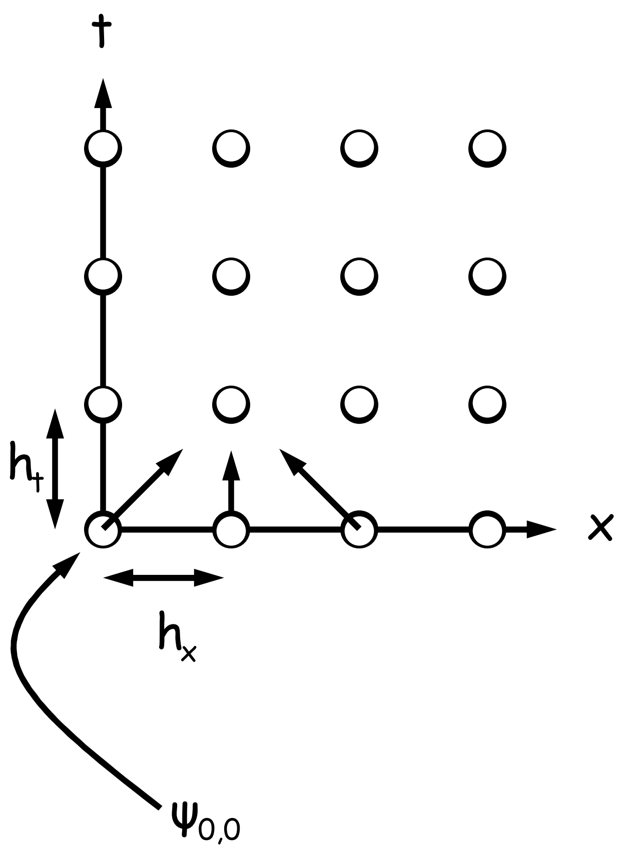
Langkah:

1. Buat grid pada bidang xt, dengan lebar h_x untuk arah x dan h_t untuk arah t.
2. (Dianggap nilai awal dan nilai pada batas-batas daerah x diketahui.)

   Hitung dengan rumus :

$$\psi_{i,j+1} = 4\Gamma h_t \pi \rho_{i,j} + \psi_{i,j} + \frac{\Gamma h_t}{h_x^2}\left(\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}\right)$$

   secara berurutan $\psi_{i,j+1}$ untuk j = 0 & i = 1, 2, 3, ..., lalu j = 1 & i = 1, 2, 3, ..., j = 2 & i = 1, 2, 3, ... dan seterusnya.



Kasus khusus:

Jika ρ dan nilai pada batas-batas daerah x tetap (tidak bergantung waktu), maka akan tercapai suatu waktu t, bahwa $\psi_{i,j+1}$ tidak berubah lagi (atau berubah hanya sedikit, sehingga dapat diabaikan):

$$\left|\psi_{i,j+2} - \psi_{i,j+1}\right| < \varepsilon \text{ (bilangan kecil)}$$

Persamaan Differensial Hiperbolik

Bentuk umum PD hiperbolik: $\left(\nabla^2 - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right)\psi(\vec{r}) = -4\pi\rho(\vec{r})$

Untuk kasus 2 dimensi: $\left(\frac{\partial^2}{\partial x^2} - \frac{1}{c^2}\frac{\partial^2}{\partial t^2}\right)\psi(x,t) = -4\pi\rho(x,t)$

Gunakan metode finite differences:

$$\left.\frac{\partial^2}{\partial x^2}\psi(x,t)\right|_{x_i,t_j} \approx \frac{\psi(x_{i+1},t_j) - 2\psi(x_i,t_j) + \psi(x_{i-1},t_j)}{h_x^2} = \frac{\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}}{h_x^2}$$

$$\left.\frac{\partial^2}{\partial t^2}\psi(x,t)\right|_{x_i,t_j} \approx \frac{\psi(x_i,t_{j+1}) - 2\psi(x_i,t_j) + \psi(x_i,t_{j-1})}{h_t^2} = \frac{\psi_{i,j+1} - 2\psi_{i,j} + \psi_{i,j-1}}{h_t^2}$$

$$(h_x = x_{i+1} - x_i,\ h_t = t_{j+1} - t_j)$$

Untuk tiap posisi dicari perubahan ψ terhadap waktu.

Diperoleh: $\boxed{\psi_{i,j+1} = 4c^2h_t^2\pi\rho_{i,j} + 2\psi_{i,j} - \psi_{i,j-1} + \frac{c^2h_t^2}{h_x^2}\left(\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}\right)}$

Untuk j = 0 diperoleh $\psi_{i,1}$ sebagai berikut:

$$\psi_{i,1} = 4c^2h_t^2\pi\rho_{i,0} + 2\psi_{i,0} - \psi_{i,-1} + \frac{c^2h_t^2}{h_x^2}\left(\psi_{i+1,0} - 2\psi_{i,0} + \psi_{i-1,0}\right)$$

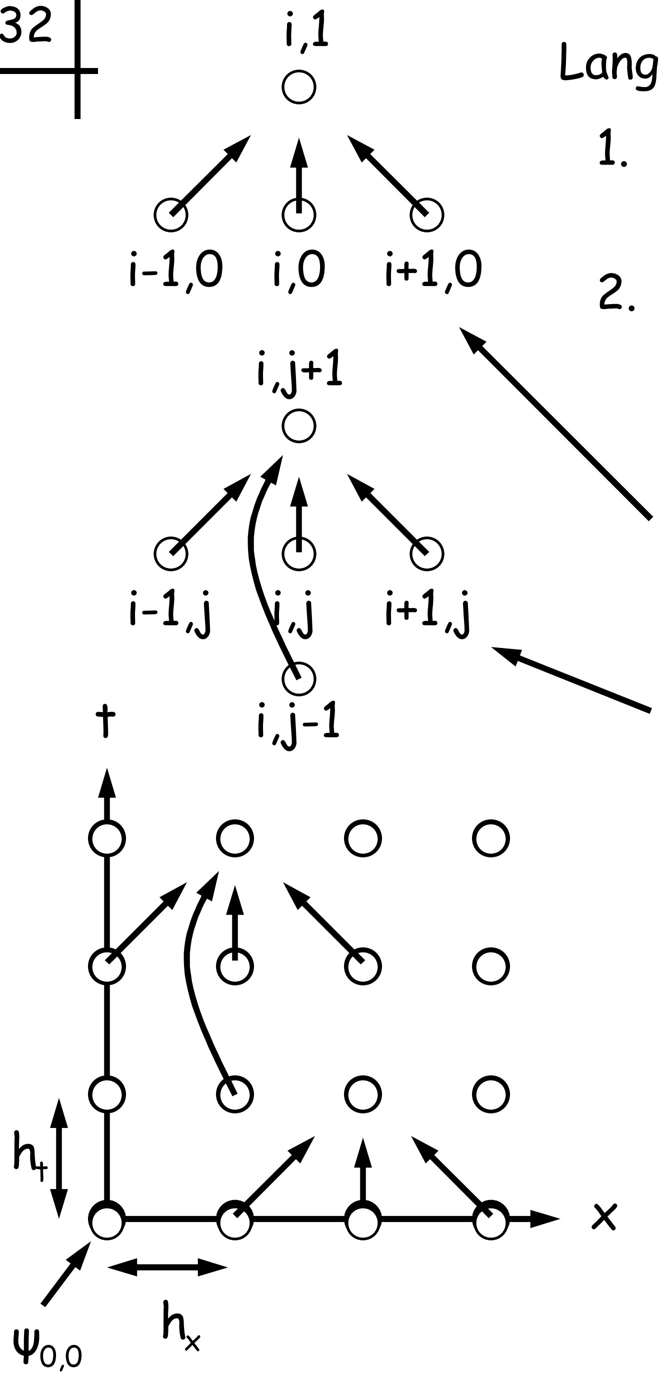
?

Anggap $\frac{\partial}{\partial t}\psi(x,t)$ pada semua x dan t = 0 diketahui: $\left.\frac{\partial}{\partial t}\psi(x,t)\right|_{x_i,t_0} = b_i$

Maka gunakan: $\left.\frac{\partial}{\partial t}\psi(x,t)\right|_{x_i,t_0} \approx \frac{\psi(x_i,t_1) - \psi(x_i,t_{-1})}{2h_t} = \frac{\psi_{i,1} - \psi_{i,-1}}{2h_t}$, shg: $\psi_{i,-1} = \psi_{i,1} - 2b_ih_t$

Dengan demikian:

- untuk j = 0: $\psi_{i,1} = 2c^2h_t^2\pi\rho_{i,0} + b_ih_t + \psi_{i,0} + \frac{c^2h_t^2}{2h_x^2}\left(\psi_{i+1,0} - 2\psi_{i,0} + \psi_{i-1,0}\right)$
- untuk j > 0: $\psi_{i,j+1} = 4c^2h_t^2\pi\rho_{i,j} + 2\psi_{i,j} - \psi_{i,j-1} + \frac{c^2h_t^2}{h_x^2}\left(\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}\right)$



Langkah:

1. Buat grid pada bidang xt, dengan lebar h_x untuk arah x dan h_t untuk arah t.
2. (Dianggap nilai awal dan nilai pada batas-batas daerah x diketahui.)

   Hitung dengan rumus :

$$\psi_{i,1} = 2c^2 h_t^2 \pi \rho_{i,0} + b_i h_t + \psi_{i,0} + \frac{c^2 h_t^2}{2h_x^2}\left(\psi_{i+1,0} - 2\psi_{i,0} + \psi_{i-1,0}\right)$$

$$\psi_{i,j+1} = 4c^2 h_t^2 \pi \rho_{i,j} + 2\psi_{i,j} - \psi_{i,j-1} + \frac{c^2 h_t^2}{h_x^2}\left(\psi_{i+1,j} - 2\psi_{i,j} + \psi_{i-1,j}\right)$$

secara berurutan $\psi_{i,j+1}$ untuk j = 0 & i = 1, 2, 3, ..., lalu j = 1 & i = 1, 2, 3, ..., j = 2 & i = 1, 2, 3, ... dan seterusnya.

Persamaan Eigenvalue

Contoh, lagi, gelombang pada tali yang kedua ujungnya diikat. Pada suatu waktu simpangan di sepanjang tali y(x) memenuhi PD orde 2:

$$f(x)\frac{d^2}{dx^2}y(x)=ky(x)$$

dengan k berhubungan dengan frekwensi, yang nilainya tidak sembarang, yang menunjukkan modus gelombang. Untuk tiap-tiap modus/frekwensi/k yang mungkin, berlaku simpangan y(x) tertentu. Dengan kata lain, k merupakan eigenvalue untuk eigenfunction y(x). Persamaan di atas disebut persamaan eigenvalue.

Dengan metode Finite Differences, PD di atas menjadi: $\frac{f_i}{h^2}(y_{i+1}-2y_i+y_{i-1})=ky_i$

yang membentuk persamaan matriks:

$$\begin{pmatrix} \ddots & \ddots & & & \\ \ddots & \ddots & \ddots & & \\ & \frac{f_i}{h^2} & \frac{-2f_i}{h^2} & \frac{f_i}{h^2} & \\ & & \ddots & \ddots & \ddots \\ & & & \ddots & \ddots \end{pmatrix}\begin{pmatrix} \vdots \\ y_{i-1} \\ y_i \\ y_{i+1} \\ \vdots \end{pmatrix}=k\begin{pmatrix} \vdots \\ y_{i-1} \\ y_i \\ y_{i+1} \\ \vdots \end{pmatrix} \longrightarrow \begin{matrix} k=? \\ y=? \end{matrix}$$

Metode Pangkat (Power Method)

Persamaan eigenvalue: $A\,u_i = \lambda_i\,u_i$, $\quad u_i = \text{eigenfunction}$, $\lambda_i = \text{eigenvalue}$

Jika A matriks n x n, maka i = 1, …, n.

Sebagai eigenfunction (atau disebut juga eigenvector), u_i bersifat orthogonal dan juga komplit yaitu, sembarang fungsi (vector) x dapat dtulis sebagai kombinasi linear u_i :

$$\text{orthogonal:} \quad u_i^T u_j = \delta_{ij} \qquad \text{komplit:} \quad x = \sum_{i=1}^{n} c_i u_i$$

Bermula dengan sembarang vector x, dilakukan iterasi berikut:

$$\begin{tikzcd} A\,x = y \arrow[r] & y \to x \arrow[l, bend left] \end{tikzcd}$$

Untuk kali pertama: $$y^{(1)} = A\,x = A\sum_{i=1}^{n} c_i u_i = \sum_{i=1}^{n} c_i A u_i = \sum_{i=1}^{n} c_i \lambda_i u_i$$

Setelah m kali iterasi diperoleh: $$y^{(m)} = A^m x = A^m \sum_{i=1}^{n} c_i u_i = \sum_{i=1}^{n} c_i A^m u_i = \sum_{i=1}^{n} c_i \lambda_i^m u_i$$

Anggap λ_k merupakan eigenvalue terbesar: $\left|\frac{\lambda_{i\neq k}}{\lambda_k}\right| < 1$

Maka, jika m besar (banyak iterasi):

$$y^{(m)} = A^m x = c_k \lambda_k^m u_k + \sum_{i\neq k} c_i \lambda_i^m u_i = \lambda_k^m \left(c_k u_k + \sum_{i\neq k} c_i \frac{\lambda_i^m}{\lambda_k^m} u_i \right) \cong c_k \lambda_k^m u_k$$

λ_k diperoleh dengan jalan:

$$x^T y^{(m)} = x^T A^m x \cong c_k \lambda_k^m \sum_{i=1}^{n} c_i u_i^T u_k \cong c_k \lambda_k^m \sum_{i=1}^{n} c_i \delta_{ik} \cong c_k^2 \lambda_k^m \longrightarrow \boxed{\lambda_k = \frac{x^T y^{(m)}}{x^T y^{(m-1)}}}$$

u_k diperoleh dengan jalan:

$$\left|y^{(m)}\right|^2 = y^{(m)T} y^{(m)} \cong \left(c_k \lambda_k^m\right)^2 u_k^T u_k \cong \left(c_k \lambda_k^m\right)^2 \longrightarrow \boxed{u_k \cong \frac{y^{(m)}}{\left|y^{(m)}\right|}}$$

Jika $\lambda_k^{(m)}$ merupakan nilai λ_k yang diperoleh setelah iterasi sebanyak m kali, maka iterasi dihentikan setelah dicapai nilai yang konvergen:

$$\left|1 - \frac{\lambda_k^{(m-1)}}{\lambda_k^{(m)}}\right| < \varepsilon, \qquad \varepsilon = \text{bilangan kecil}$$

Untuk mencari eigenvalue terbesar kedua, hilangkan u_k dari perhitungan.
Jadi, dipakai vector awal baru x':

$$x' = (A - \lambda_k)\,x = \sum_{i=1}^{n} c_i (A - \lambda_k) u_i = \sum_{i=1}^{n} c_i (\lambda_i - \lambda_k) u_i = \sum_{i \neq k} c_i (\lambda_i - \lambda_k) u_i = \sum_{i \neq k} d_i u_i \quad \left(d_i \equiv c_i (\lambda_i - \lambda_k)\right)$$

Iterasi:

$$A\,x' = y' \longrightarrow y' \to x'$$

Setelah m kali iterasi diperoleh: $\quad y'^{(m)} = A^m x' = \sum_{i \neq k} d_i \lambda_i^{\,m} u_i$

Anggap λ_l merupakan eigenvalue terbesar kedua: $\quad \left| \dfrac{\lambda_{i \neq k \neq l}}{\lambda_l} \right| < 1$

sehingga setelah banyak iterasi: $\quad y'^{(m)} \cong d_l \lambda_l^{\,m} u_l$

Memperoleh λ_l dan u_l: $\quad \boxed{\lambda_l = \dfrac{x'^T y'^{(m)}}{x'^T y'^{(m-1)}} \qquad u_l \cong \dfrac{y'^{(m)}}{\left| y'^{(m)} \right|}}$

Pola yang sama berlaku untuk mencari eigenvalue terbesar berikutnya.

Metode Pangkat Kebalikan (Inverse Power Method)

Dengan metode pangkat didapat eigenvalue terbesar. Untuk mencari eigenvalue terkecil digunakan metode pangkat kebalikan.

$$A\,u_i = \lambda_i\,u_i \longrightarrow A^{-1}A\,u_i = \lambda_i\,A^{-1}u_i \longrightarrow A^{-1}u_i = \lambda_i^{-1}u_i$$

Bermula dengan sembarang vector x, dilakukan iterasi berikut:

$$\begin{tikzcd} A^{-1}\,x = y \arrow[r] & y \to x \arrow[l, bend left] \end{tikzcd}$$

Setelah m kali iterasi diperoleh: $y^{(m)} = (A^{-1})^m x = \sum_{i=1}^{n} c_i(\lambda_i^{-1})^m u_i$

Jika λ_s eigenvalue terkecil, maka setelah banyak kali iterasi: $y^{(m)} \cong c_s(\lambda_s^{-1})^m u_s$

Jadi, λ_s diperoleh sebagai: $\boxed{\lambda_s = \dfrac{x^T y^{(m-1)}}{x^T y^{(m)}}}$ dan u_s : $\boxed{u_s \cong \dfrac{y^{(m)}}{\left|y^{(m)}\right|}}$