Bab 3

Solusi Persamaan Nirlanjar

Saya tidak tahu bagaimana saya tampak pada dunia; tetapi bagi saya sendiri saya nampaknya hanyalah seperti seorang anak laki-laki yang bermain-main di pantai, dan mengalihkan diri sendiri sekarang dan kemudian menemukan koral yang lebih halus atau kerang yang lebih indah daripada yang biasa, sementara samudera besar kebenaran semuanya terbentang di hadapan saya tak terungkapkan.
(Isaac Newton)

Dalam bidang sains dan rekayasa, para ahli ilmu alam dan rekayasawan sering berhadapan dengan persoalan mencari solusi persamaan –lazim disebut **akar persamaan** (*roots of equation*) atau **nilai-nilai nol**– yang berbentuk $f(x) = 0$. Beberapa persamaan sederhana mudah ditemukan akarnya. Misalnya $2x – 3 = 0$, pemecahannya adalah dengan memindahkan -3 ke ruas kanan sehingga menjadi $2x = 3$, dengan demikian solusi atau akarnya adalah $x = 3/2$. Begitu juga persaman kuadratik seperti $x^2 – 4x – 5 = 0$, akar-akarnya mudah ditemukan dengan cara pemfaktoran menjadi $(x – 5)(x + 1) = 0$ sehingga $x_1 = 5$ dan $x_2 = -1$.

Umumnya persamaan yang akan dipecahkan muncul dalam bentuk nirlanjar (*non linear*) yang melibatkan bentuk sinus, cosinus, eksponensial, logaritma, dan fungsi transenden lainnya. Misalnya,

1. Tentukan akar riil terkecil dari

$$9.34 - 21.97x + 16.3x^3 - 3.704x^5 = 0$$

2. Kecepatan ke atas sebuah roket dapat dihitung dengan memakai rumus berikut:

$$v = u \ln\left|\frac{m_0}{m_0 - qt}\right| - gt$$

yang dalam hal ini v adalah kecepatan ke atas, u adalah kecepatan pada saat bahan bakar dikeluarkan relatif terhadap roket, m_0 massa awal roket pada saat t = 0, q laju pemakaian bahan bakar, dan g percepatan gravitasi (= 9.8 m/det^2). Jika u = 2200 m/det, m_0 = 160000 kg, dan q = 2680 kg/det, hitunglah waktu saat v = 1000 m/det. (Nyatakan persamaan dengan ruas kanan sama dengan 0:

$$v - u \ln[m_0/(m_0 - qt)] + gt = 0 \quad)$$

3. Dalam teknik kelautan, persamaan gelombang berdiri yang dipantulkan oleh dermaga pelabuhan diberikan oleh

$$h = h_0 \{ \sin(2px/l) \cos(2ptv/l) + e^{-x} \}$$

Tentukan x jika $h = 0.5h_0$, l = 20, t = 10 dan v = 50!
(Nyatakan persamaan dengan ruas kanan sama dengan 0:

$$h - h_0 \{ \sin(2px/l) \cos(2ptv/l) + e^{-x} \} = 0 \quad)$$

4. Suatu arus osilasi dalam rangkaian listrik diberikan oleh

$$I = 10e^{-t} \sin(2p\,t)$$

yang dalam hal ini t dalam detik. Tentukan semua nilai t sedemikan sehingga I = 2 ampere.
(Nyatakan persamaan dengan ruas kanan sama dengan 0:

$$I - 10e^{-t} \sin(2p\,t) = 0 \quad)$$

5. Dalam bidang teknik lingkungan, persamaan berikut ini dapat digunakan untuk menghitung tingkat oksigen pada hilir sungai dari tempat pembuangan limbah:

$$c = 10 - 15(e^{-0.1x} - e^{-0.5x})$$

yang dalam hal ini x adalah jarak hilir sungai ke tempat pembuangan limbah. Tentukan jarak hilir sungai tersebut bila pembacaan pertama pada alat pengukur tingkat oksigen adalah 4 bila pengukur berada 5 mil dari pembuangan.
(Nyatakan persamaan dengan ruas kanan sama dengan 0:

$$c - 10 - 15(e^{-0.1x} - e^{-0.5x}) = 0$$

)

6. Reaksi kesetimbangan

$$2A + B \Leftrightarrow C$$

dapat dicirikan oleh hubungan setimbang

$$K = \frac{[C]}{[A]^2[B]}$$

yang dalam hal ini [.] menyatkan konsentrasi zat kimia. Andaikan bahwa kita mendefenisikan peubah x sebagai jumlah mol C yang dihasilkan. Hukum kekekalan massa dapat dipakai untuk merumuskan ulang hubungan keseimbangan itu sebagai

$$K = \frac{[C_0] + x}{([A_0] - 2x)([B_0] - x)}$$

yang dalam hal ini indeks 0 menounjukkan konsentrasi awal tiap unsur. Jika diketahui tetapan kesetimbangan $K = 1.25 \times 10^{-2}$, dan konsentrasi larutan $[A_0]$ = 50, $[B_0]$ = 40, dan $[C_0]$ = 5, hitunglah x.
(Nyatakan persamaan dengan ruas kanan sama dengan 0:

$$K - \frac{[C_0] + x}{([A_0] - 2x)([B_0] - x)} = 0 \qquad)$$

Keenam contoh di atas memperlihatkan bentuk persamaan yang rumit/kompleks yang tidak dapat dipecahkan secara analitik (seperti persamaan kuadratik pada paragraf awal). Bila metode analitik tidak dapat menyelesaikan persamaan, maka kita masih bisa mencari solusinya dengan menggunakan metode numerik.

3.1 Rumusan Masalah

Persoalan mencari solusi persamaan nirlanjar dapat dirumuskan secara singkat sebagai berikut: tentukan nilai x yang memenuhi persamaan

$$f(x) = 0 \qquad \text{(P.3.1)}$$

yaitu nilai $x = s$ sedemikian sehingga $f(s)$ sama dengan nol.

3.2 Metode Pencarian Akar

Dalam metode numerik, pencarian akar $f(x) = 0$ dilakukan secara lelaran (iteratif). Sampai saat ini sudah banyak ditemukan metode pencarian akar. Secara umum, semua metode pencarian akar tersebut dapat dikelompokkan menjadi dua golongan besar:

1. **Metode tertutup** atau metode pengurung (*bracketing method*)

   Metode yang termasuk ke dalam golongan ini mencari akar di dalam selang [a, b]. Selang [a, b] sudah dipastikan berisi minimal satu buah akar, karena itu metode jenis ini selalu berhasil menemukan akar. Dengan kata lain, lelarannya selalu konvergen (menuju) ke akar, karena itu metode tertutup kadang-kadang dinamakan juga **metode konvergen.**

2. **Metode terbuka**

   Berbeda dengan metode tertutup, metode terbuka tidak memerlukan selang [a, b] yang mengandung akar. Yang diperlukan adalah tebakan (*guest*) awal akar, lalu, dengan prosedur lelaran, kita menggunakannya untuk menghitung hampiran akar yang baru. Pada setiap kali lelaran, hampiran akar yang lama dipakai untuk menghitung hampiran akar yang baru. Mungkin saja hampiran akar yang baru mendekati akar sejati (konvergen), atau mungkin juga menjauhinya (divergen). Karena itu, metode terbuka tidak selalu berhasil menemukan akar, kadang-kadang konvergen, kadangkala ia divergen.

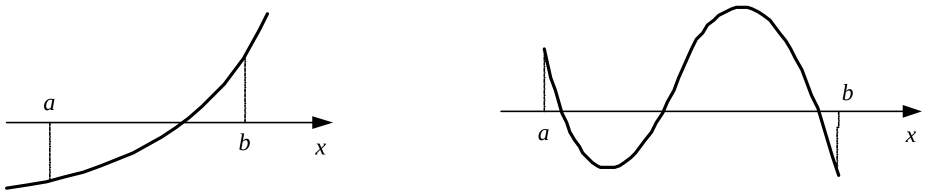
3.3 Metode Tertutup

Seperti yang telah dijelaskan, metode tertutup memerlukan selang [a,b] yang mengandung akar. Sebagaimana namanya, selang tersebut “mengurung” akar sejati. Tata-ancang (*strategy*) yang dipakai adalah mengurangi lebar selang secara sistematis sehingga lebar selang tersebut semakin sempit, dan karenanya menuju akar yang benar.

Dalam sebuah selang mungkin terdapat lebih dari satu buah akar atau tidak ada akar sama sekali. Secara grafik dapat ditunjukkan bahwa jika:

(1) $f(a)f(b) < 0$

maka terdapat akar sebanyak bilangan ganjil (Gambar 3.1).

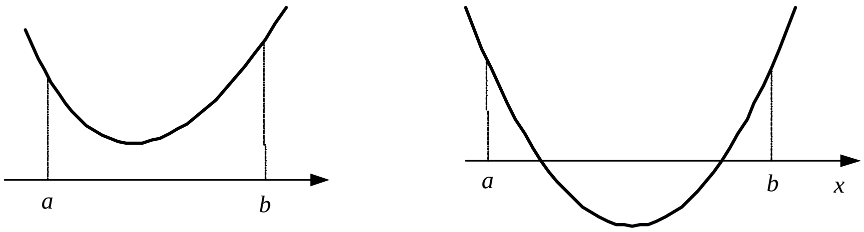


(a) (b)

Gambar 3.1 Banyaknya akar ganjil

(2) $f(a)f(b) > 0$

maka terdapat akar sebanyak bilangan genap atau tidak ada akar sama sekali (Gambar 3.2).



(a) (b)

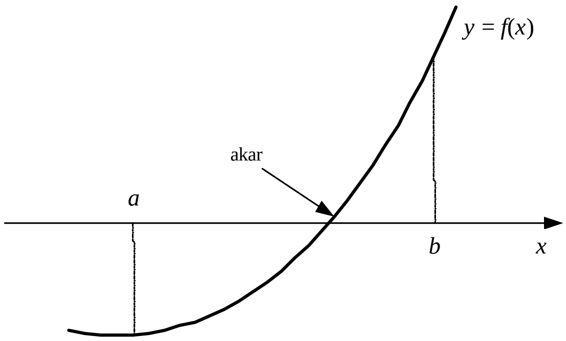
Gambar 3.2 Banyaknya akar genap

Syarat Cukup Keberadaan Akar

Gambar 3.1 memperlihatkan bahwa selalu ada akar di dalam selang [a, b] jika nilai fungsi berbeda tanda (+/-) di $x = a$ dan $x = b$. Tidak demikian halnya jika nilai fungsi di ujung-ujung selang sama tandanya, yang mengisyaratkan mungkin ada akar atau tidak ada sama sekali. Jadi, jika nilai fungsi berbeda tanda tanda di ujung-ujung selang, pastilah terdapat paling sedikit satu buah akar di dalam selang tersebut. Dengan kata lain, syarat cukup keberadaan akar persamaan kita tulis sebagai berikut:

> Jika $f(a)\ f(b) < 0$ dan f(x) menerus di dalam selang [a, b], maka
> <u>paling sedikit</u> terdapat satu buah akar persamaan $f(x) = 0$ di dalam selang [a, b].

Syarat ini disebut syarat cukup[1] -bukan syarat perlu- sebab meskipun nilai-nilai di ujung selang tidak berbeda tanda, mungkin saja terdapat akar di dalam selang tersebut (seperti ditunjukkan pada Gambar 3.2). Syarat cukup keberadaan akar ini ditunjukkan pada Gambar 3.3.



Gambar 3.3 Lokasi akar

Ada dua masalah yang terjadi karena ketidaktepatan mengambil selang [a, b]. Masalah pertama adalah bila di dalam selang [a, b] terdapat lebih dari satu buah akar. Sekali suatu metode tertutup digunakan untuk mencari akar di dalam selang [a, b], ia hanya menemukan sebuah akar saja. Karena itu, bila kita mengambil selang [a, b] yang mengandung lebih dari satu akar, hanya satu buah akar saja yang berhasil ditemukan (lihat kembali Gambar 3.1(b)).

Masalah kedua adalah bila mengambil selang [a, b] yang tidak memenuhi syarat cukup. Adakalanya kita dapat "kehilangan" akar karena selang [a, b] yang diambil ternyata tidak memenuhi syarat cukup $f(a)f(b) < 0$. Sehingga, kita mungkin sampai pada kesimpulan tidak terdapat akar di dalam selang [a, b] tersebut, padahal seharusnya ada (lihat kembali Gambar 3.2 (b)).

Untuk mengatasi kedua masalah di atas, pengguna metode tertutup disarankan mengambil selang yang berukuran cukup kecil yang memuat hanya satu akar. Ada dua pendekatan yang dapat kita gunakan dalam memilih selang tersebut.

---

[1] Bentuk implikasi "jika p maka q" bisa dibaca sebagai "p adalah syarat cukup untuk q". Di dalam kalkulus proposisi, pernyataan "jika p maka q" (dilambangkan dengan $p \rightarrow q$) adalah benar kecuali jika p benar dan q salah. Jadi, pernyataan tersebut tetap benar meskipun $f(a)f(b) > 0$ dan di dalam selang [a, b] terdapat paling sedikit satu buah akar atau tidak terdapat akar sama sekali. Pernyataan tersebut jelas salah bila $f(a)f(b) > 0$ dan di dalam selang [a, b] terdapat paling sedikit satu buah akar (tidak mungkin).

Pendekatan pertama adalah membuat grafik fungsi di bidang *X*-*Y*, lalu melihat di mana perpotongannya dengan sumbu-*X*. Dari sini kita dapat mengira-ngira selang yang memuat titik potong tersebut. Grafik fungsi dapat dibuat dengan program yang ditulis sendiri, atau lebih praktis menggunakan paket program yang dapat membuat grafik fungsi.

Pendekatan yang kedua adalah dengan mencetak nilai fungsi pada titik-titik absis yang berjarak tetap. Jarak titik ini dapat diatur cukup kecil. Jika tanda fungsi berubah pada sebuah selang, pasti terdapat minimal satu akar di dalamnya. Program 3.1 berisi prosedur untuk menemukan selang yang cukup kecil yang mengandung akar. Program ini mencetak tabel titik-titik sepanjang selang $[a, b]$. Dari tabel tersebut kita dapat menentukan upaselang yang nilai fungsi di ujung-ujungnya berbeda tanda. Keberhasilan dari pendekatan ini bergantung pada jarak antara titik-titik absis. Semakin kecil jarak titik absis, semakin besar peluang menemukan selang yang mengandung hanya sebuah akar.

Program 3.1 Menemukan selang kecil yang mengandung akar

```
procedure Cari_SelangKecilYangMengandungAkar(a, b, h: real);
{ Menentukan dan mencetak nilai-nilai fungsi untuk absis x di dalam
selang [a, b]. Jarak antara tiap absis adalah h.
K.Awal: a dan b adalah ujung-ujung selang, nilainya sudah terdefenisil;
        h adalah jarak antara tiap absis x
K.Akhir: tabel yang berisi x dan f(x) dicetak ke layar
}
var
   x : real;
begin
   x:=a;
   writeln('----------------------');
   writeln('     x          f(x)   ');
   writeln('----------------------');
   while x <= b do begin
        writeln(x:5:2, f(x):10:6);
        x:=x+h;
   end;
   { x > b }
   writeln('----------------------');
end;
```

Bila Program 3.1 digunakan untuk mencari selang kecil yang mengandung akar pada fungsi $f(x) = e^x - 5x^2$ mulai dari $a = -0.5$ sampai $b = 1.4$ dengan kenaikan absis sebesar $h = 0.1$, maka hasilnya tampak pada tabel berikut:

x	$f(x)$
-0.50	-0.643469
-0.40	-0.129680
-0.30	0.290818
-0.20	0.618731
-0.10	0.854837
0.00	1.000000
0.10	1.055171
0.20	1.021403
0.30	0.899859
0.40	0.691825
0.50	0.398721
0.60	0.022119
0.70	-0.436247
0.80	-0.974459
0.90	-1.590397
1.00	-2.281718
1.10	-3.045834
1.20	-3.879883
1.30	-4.780703
1.40	-5.744800

Berdasarkan tabel di atas, selang yang cukup kecil yang mengandung akar adalah

[-0.40, -0.30] dan [0.60, 0.70]

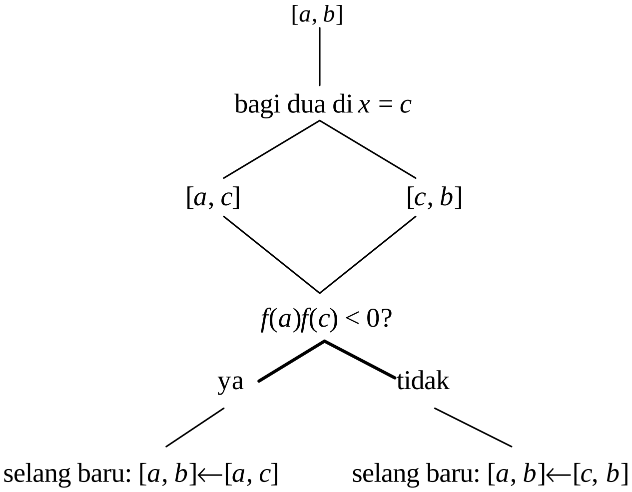
karena nilai fungsi berubah tanda di ujung-ujung selangnya. Selang [0.00, 1.00] juga dapat kita ambil tetapi cukup lebar, demikian juga [-0.50, 1.40], [-0.30, 0.80], dan seterusnya.

Ada dua metode klasik yang termasuk ke dalam metode tertutup, yaitu **metode bagidua** dan **metode regula-falsi**. Masing-masing metode kita bahas lebih rinci di bawah ini.

3.3.1 Metode Bagidua[2]

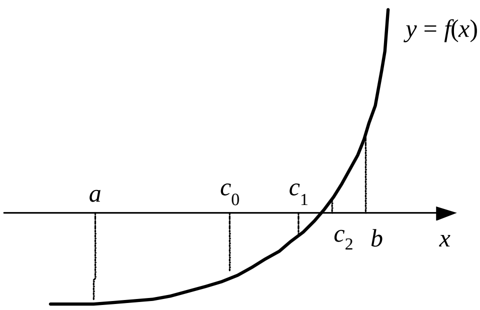
Misalkan kita telah menentukan selang $[a, b]$ sehingga $f(a)f(b) < 0$. Pada setiap kali lelaran, selang $[a, b]$ kita bagi dua di $x = c$, sehingga terdapat dua buah upaselang yang berukuran sama, yaitu selang $[a, c]$ dan $[c, b]$. Selang yang diambil untuk lelaran berikutnya adalah upaselang yang memuat akar, bergantung pada apakah $f(a)f(c) < 0$ atau $f(c)f(b) < 0$.

[2]Nama lainnya adalah *metode Bolzano*



Selang yang baru dibagi dua lagi dengan cara yang sama. Begitu seterusnya sampai ukuran selang yang baru sudah sangat kecil (lihat Gambar 3.4). Kondisi berhenti lelaran dapat dipilih salah satu dari tiga kriteria berikut:

1. Lebar selang baru: $|a - b| < e$, yang dalam hal ini e adalah nilai toleransi lebar selang yang mengurung akar.
2. Nilai fungsi di hampiran akar: $f(c) = 0$. Beberapa bahasa pemrograman membolehkan pembandingan dua buah bilangan riil, sehingga perbandingan $f(c) = 0$ dibenarkan. Namun kalau kita kembali ke konsep awal bahwa dua buah bilangan riil tidak dapat dibandingkan kesamaannya karena representasinya di dalam mesin tidak tepat, maka kita dapat menggunakan bilangan yang sangat kecil (misalnya epsilon mesin) sebagai pengganti nilai 0. Dengan demikian, menguji kesamaan $f(c) = 0$ dapat kita hampiri dengan $f(c)$ < epsilon_mesin.
3. Galat relatif hampiran akar: $|(c_{baru} - c_{lama})/c_{baru}| < d$, yang dalam hal ini d adalah galat relatif hampiran yang diinginkan.



Gambar 3.4 Proses pembagian selang [*a*, *b*] dengan metode bagidua

Program 3.2 berisi algoritma metode bagidua. Di dalam algoritma tersebut, format penulisan keluaran tidak dituliskan untuk menghindari kerumitan algoritma dari hal-hal yang tidak esensial.

Program 3.2 Metode bagidua

```
procedure BagiDua(a,b: real);
{ Mencari akar f(x)=0 di dalam selang [a,b] dengan metode bagidua
  K.Awal : a dan b adalah ujung-ujung selang sehingga f(a)*f(b) < 0,
           nilai a dan b sudah terdefinisi.
  K.Akhir : Hampiran akar tercetak di layar.
}
const
   epsilon1 = 0.000001;       {batas lebar selang akhir lelaran}
   epsilon2 = 0.00000001;     {bilangan yang sangat kecil, mendekati nol}
begin
   repeat
        c:=(a+b)/2;    { titik tengah [a,b]}
        if f(a)*f(c) < 0 then
          b:=c        {selang baru [a,b]=[a,c]}
        else
          a:=c;        {selang baru [a,b]=[c,b]}
   until (ABS(a-b)< epsilon1) or (f(c)) < epsilon2);
   { c adalah akar persamaan }
   writeln('Hampiran kar = ', x:10:6);
end;
```

Kasus yang Mungkin Terjadi pada Penggunaan Metode Bagidua

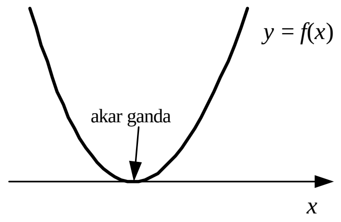
1. Jumlah akar lebih dari satu

   Bila dalam selang [a, b] terdapat lebih dari satu akar (banyaknya akar ganjil), hanya satu buah akar yang dapat ditemukan (lihat kembali Gambar 3.1(b)).
   Cara mengatasinya: gunakan selang [a,b] yang cukup kecil yang memuat hanya satu buah akar.

2. Akar ganda.

   Metode bagidua tidak berhasil menemukan akar ganda. Hal ini disebabkan karena tidak terdapat perbedaan tanda di ujung-ujung selang yang baru (Gambar 3.5).

   Contoh: $f(x) = (x - 3)^2 = (x - 3)(x - 3)$, mempunyai dua akar yang sama, yaitu $x = 3$.

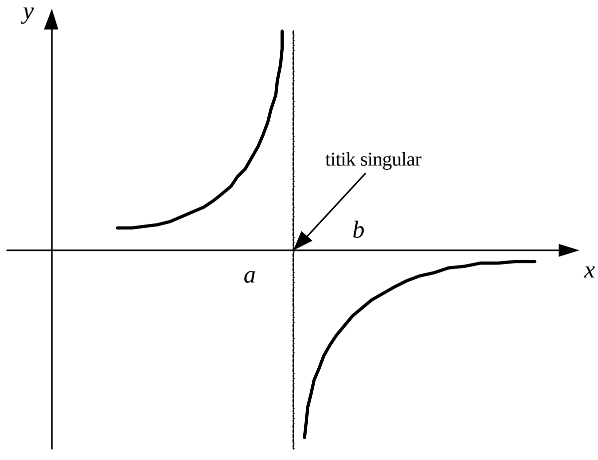


Gambar 3.4 Akar ganda

Cara mengatasinya: akan dibahas pada upabab 3.5.

3. Singularitas.

   Pada titik singular, nilai fungsinya tidak terdefinisi. Bila selang [a, b] mengandung titik singular, lelaran metode bagidua tidak pernah berhenti. Penyebabnya, metode bagidua menganggap titik singular sebagai akar karena lelaran cenderung konvergen. Yang sebenarnya, titik singular bukanlah akar, melainkan *akar semu* (Gambar 3.6)



Gambar 3.6 Fungsi singular

Cara mengatasinya: periksa nilai $|f(b) - f(a)|$. Jika $|f(b) - f(a)|$ konvergen ke nol, akar yang dicari pasti akar sejati, tetapi jika $|f(b) - f(a)|$ divergen, akar yang dicari merupakan titik singular (akar semu).

Pada setiap lelaran pada metode bagidua, kita mencatat bahwa selisih antara akar sejati dengan akar hampiran tidak pernah melebihi setengah panjang selang saat itu. Pernyataan ini dinyatakan degan teorema berikut.

TEOREMA 3.1. Jika $f(x)$ menerus di dalam selang $[a,b]$ dengan $f(a)\,f(b) < 0$ dan $s \in [a, b]$ sehingga $f(s) = 0$ dan $c_r = (a_r + b_r)/2$, maka selalu berlaku dua ketidaksamaan berikut:

(i) $|s - c_r| \le |b_r - a_r| / 2$

dan

(ii) $|s - c_r| \le \dfrac{|b - a|}{2^{r+1}}, \quad r = 0, 1, 2, \ldots$

Bukti:

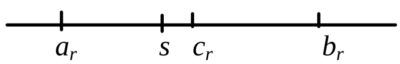
Misalkan pada lelaran ke-r kita mendapatkan selang $[a_r, b_r]$ yang panjangnya setengah panjang selang sebelumnya, $[a_{r-1}, b_{r-1}]$.
Jadi,

$$|b_r - a_r| = |b_{r-1} - a_{r-1}| / 2$$

Jelaslah bahwa

$$|b_1 - a_1| = |b_0 - a_0|/2 = |b - a|/2$$
$$|b_2 - a_2| = |b_1 - a_1|/2 = |b - a|/2^2$$
$$|b_3 - a_3| = |b_2 - a_2|/2 = |b - a|/2^3$$
$$\ldots$$
$$|b_r - a_r| = |b_{r-1} - a_{r-1}|/2 = |b - a|/2^r$$

Pada lelaran ke-r, posisi c_r (akar hampiran) dan s (akar sejati) adalah seperti diagram berikut:



Berdasarkan diagram di atas jelaslah bahwa

$$|s - c_r| \le \frac{|b_r - a_r|}{2}$$

Selanjutnya,

$$|s - c_r| \le \frac{|b_r - a_r|}{2} = \frac{1}{2}\frac{|b - a|}{2^r} = \frac{|b - a|}{2^{r+1}} \qquad \blacksquare$$

Jadi, selisih antara akar sejati dengan akar hampiran tidak pernah lebih dari setengah epsilon.

Dengan mengingat kriteria berhenti adalah $|b_r - a_r| < \varepsilon$, maka dari (i) terlihat bahwa

$$|s - c_r| < a / 2$$

sehingga

$$\frac{|b-a|}{2^{r+1}} < \frac{e}{2}$$

$$\Leftrightarrow\ 2^r > |b-a| / e$$

$$\Leftrightarrow\ r \ln(2) > \ln\left(|b-a|\right) - \ln(e)$$ ket: ln adalah logaritma natural

$$\Leftrightarrow\ r > \frac{\ln\left(|b-a|\right) - \ln(e)}{\ln(2)}$$

$$\Leftrightarrow\ R > \frac{\ln\left(|b-a|\right) - \ln(e)}{\ln(2)}$$

yang dalam hal ini R adalah jumlah lelaran (jumlah pembagian selang) yang dibutuhkan untuk menjamin bahwa c adalah hampiran akar yang memiliki galat kurang dari e.

Contoh 3.1

Temukan akar $f(x) = e^x - 5x^2$ di dalam selang [0, 1] dan e = 0.00001.

Penyelesaian: Tabel lelaran menggunakan metode bagidua:

r	*a*	*c*	*b*	*f*(*a*)	*f*(*c*)	*f*(*b*)	Selang baru	Lebarnya
0	0.000000	0.500000	1.000000	1.000000	0.398721	-2.281718	[c, b]	0.500000
1	0.500000	0.750000	1.000000	0.398721	-0.695500	-2.281718	[a, c]	0.250000
2	0.500000	0.625000	0.750000	0.398721	-0.084879	-0.695500	[a, c]	0.125000
3	0.500000	0.562500	0.625000	0.398721	0.173023	-0.084879	[c, b]	0.062500
4	0.562500	0.593750	0.625000	0.173023	0.048071	-0.084879	[c, b]	0.031250
5	0.593750	0.609375	0.625000	0.048071	-0.017408	-0.084879	[a, c]	0.015625
6	0.593750	0.601563	0.609375	0.048071	0.015581	-0.017408	[c, b]	0.007813
7	0.601563	0.605469	0.609375	0.015581	-0.000851	-0.017408	[a, c]	0.003906
8	0.601563	0.603516	0.605469	0.015581	0.007380	-0.000851	[c, b]	0.001953
9	0.603516	0.604492	0.605469	0.007380	0.003268	-0.000851	[c, b]	0.000977

10	0.604492	0.604980	0.605469	0.003268	0.001210	-0.000851	[c, b]	0.000488
11	0.604980	0.605225	0.605469	0.001210	0.000179	-0.000851	[c, b]	0.000244
12	0.605225	0.605347	0.605469	0.000179	-0.000336	-0.000851	[a, c]	0.000122
13	0.605225	0.605286	0.605347	0.000179	-0.000078	-0.000336	[a, c]	0.000061
14	0.605225	0.605255	0.605286	0.000179	0.000051	-0.000078	[c, b]	0.000031
15	0.605255	0.605270	0.605286	0.000051	-0.000014	-0.000078	[a, c]	0.000015
1 6	0.605255	0.605263	0.605270	0.000051	0.000018	-0.000014	[c, b]	0.000008

Jadi, hampiran akarnya adalah $x = 0.605263$

Jumlah lelaran yang dibutuhkan

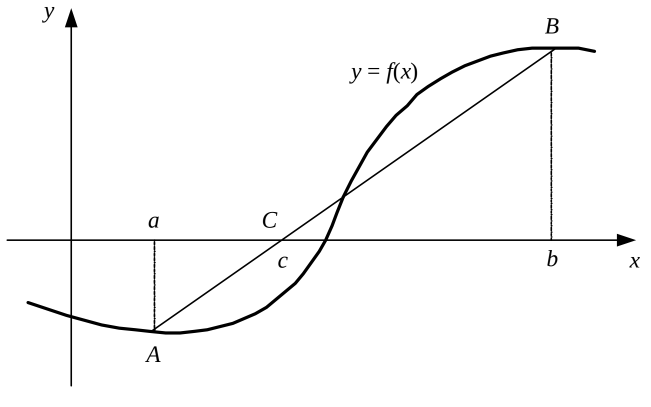
$$R > \frac{\ln(|1-0|) - \ln(0.00001)}{\ln(2)}$$

$$> 16.60964$$

Jadi, dibutuhkan minimal 17 kali lelaran (r =0 sampai dengan r =16), sesuai dengan jumlah lelaran pada tabel, agar galat akar hampiran kurang dari e. ■

3.3.2 Metode Regula-Falsi

Meskipun metode bagidua selalu berhasil menemukan akar, tetapi kecepatan konvergensinya sangat lambat. Kecepatan konvergensi dapat ditingkatkan bila nilai $f(a)$ dan $f(b)$ juga turut diperhitungkan. Logikanya, bila $f(a)$ lebih dekat ke nol daripada $f(b)$ tentu akar lebih dekat ke $x = a$ daripada ke $x = b$. Metode yang memanfaatkan nilai $f(a)$ dan $f(b)$ ini adalah **metode regula-falsi** (bahasa Latin) atau **metode posisi palsu**. (*false position method*). Dengan metode regula-falsi, dibuat garis lurus yang menghubungkan titik $(a, f(a))$ dan $(b, f(b))$. Perpotongan garis tersebut dengan sumbu-x merupakan taksiran akar yang diperbaiki. Garis lurus tadi seolah-olah berlaku menggantikan kurva $f(x)$ dan memberikan posisi palsu dari akar.



Gambar 3.7 Metode regula-falsi

Perhatikan Gambar 3.7:

gradien garis AB = gradien garis BC

$$\frac{f(b) - f(a)}{b - a} = \frac{f(b) - 0}{b - c}$$

yang dapat disederhanakan menjadi

$$c = b - \frac{f(b)(b - a)}{f(b) - f(a)} \qquad \text{(P.3.2)}$$

Algoritma regula-falsi (lihat Program 3.3) hampir sama dengan algoritma bagidua kecuali pada perhitungan nilai c.

Program 3.3 Metode regula-falsi

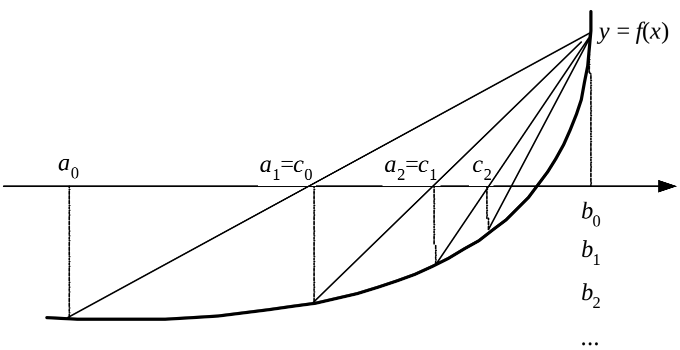
```
procedure regula_falsi(a, b: real);
{ Mencari akar f(x)=0 di dalam selang [a,b] dengan metode regulafalsi
  K.Awal : a dan b adalah ujung-ujung selang sehingga f(a)*f(b) < 0,
           harga a dan b sudah terdefenisi
  K.Akhir : Hampiran akar tercetak di layar
}
const
   epsilon1 = 0.00001;        {batas lebar selang akhir lelaran}
   epsilon2 = 0.000001;       {bilangan yang sangat kecil, bisa diganti }
begin
   repeat
        c:=b-(f(b)*(b-a)/(f(b)-f(a)));
        if abs(f(c))< epsilon2 then      {f(c) = 0, c adalah akar}
          begin
           a:=c;
           b:=c;
         end
        else
           if f(a)*f(c) < 0 then
                b:=c;   {selang baru [a,b]=[a,c]}
           else
            a:=c;     {selang baru [a,b]=[c,b]}
   until ABS(a-b)< epsilon1;
   { c adalah hampiran akar }
   writeln('Hampiran akar : ', c:10:6);
end;
```

Secara umum, metode regula-falsi lebih cepat konvergensinya dibandingkan dengan metode bagidua. Namun, pada beberapa kasus kecepatan konvergensinya justru lebih lambat. Bila kita memakai Program 3.4 untuk menghitung akar $f(x) = e^x - 5x^2$ di dalam selang [0, 1] dan $\varepsilon = 0.00001$, maka tabel lelarannya yang dihasilkan adalah sebagai berikut:

r	a	c	b	$f(a)$	$f(c)$	$f(b)$	Selang baru	Lebarnya
0	0.000000	0.304718	1.000000	1.000000	0.891976	-2.281718	[c,b]	0.695282
1	0.304718	0.500129	1.000000	0.891976	0.398287	-2.281718	[c,b]	0.499871
2	0.500129	0.574417	1.000000	0.398287	0.126319	-2.281718	[c,b]	0.425583
3	0.574417	0.596742	1.000000	0.126319	0.035686	-2.281718	[c,b]	0.403258
4	0.596742	0.602952	1.000000	0.035686	0.009750	-2.281718	[c,b]	0.397048
5	0.602952	0.604641	1.000000	0.009750	0.002639	-2.281718	[c,b]	0.395359
6	0.604641	0.605098	1.000000	0.002639	0.000713	-2.281718	[c,b]	0.394902
7	0.605098	0.605222	1.000000	0.000713	0.000192	-2.281718	[c,b]	0.394778
8	0.605222	0.605255	1.000000	0.000192	0.000052	-2.281718	[c,b]	0.394745
9	0.605255	0.605264	1.000000	0.000052	0.000014	-2.281718	[c,b]	0.394736
10	0.605264	0.605266	1.000000	0.000014	0.000004	-2.281718	[c,b]	0.394734
11	0.605266	0.605267	1.000000	0.000004	0.000001	-2.281718	[c,b]	0.394733
12	0.605267	0.605267	1.000000	0.000001	0.000000	-2.281718	[c,b]	0.394733
13	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
14	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
15	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
16	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
17	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
18	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
19	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
20	0.605267	0.605267	1.000000	0.000000	0.000000	-2.281718	[c,b]	0.394733
21	0.605267	0.605267	1.000000	0.000000	-0.000000	-2.281718	[a,c]	0.000000

Hampiran akar x = 0.605267

Jumlah lelaran tabel di atas = 22, lebih banyak daripada jumlah lelaran metode bagidua. Bila diperhatikan, dari lelaran 12 sampai lelaran 21, nilai a, b, c tidak pernah berubah, padahal $f(c)$ sudah sangat kecil ($\approx$0). Kasus seperti ini akan terjadi bila kurva fungsinya cekung (konkaf) di dalam selang [a, b]. Akibatnya, garis potongnya selalu terletak di atas kurva (bila kurvanya cekung ke atas) atau selalu terletak di bawah kurva (bila kurvanya cekung ke bawah). Perhatikan Gambar 3.8.



Gambar 3.8 Garis potong selalu terletak di atas kurva $y = f(x)$

Pada kondisi yang paling ekstrim, $|b - a_r|$ tidak pernah lebih kecil dari e, sebab salah satu titik ujung selang, dalam hal ini b, selalu tetap untuk setiap lelaran r = 0, 1, 2, ... Titik ujung selang yang tidak pernah berubah itu dinamakan **titik mandek** (*stagnant point*). Pada titik mandek,

$$|b_r - a_r| = |b - a_r| \quad r = 0, 1, 2, ...$$

yang dapat mengakibatkan program mengalami *looping*. Untuk mengatasi hal ini, kondisi berhenti pada algoritma regula-falsi harus kita tambah dengan memeriksa apakah nilai $f(c)$ sudah sangat kecil sehingga mendekati nol. Jadi, kondisi pada `repeat-until` menjadi

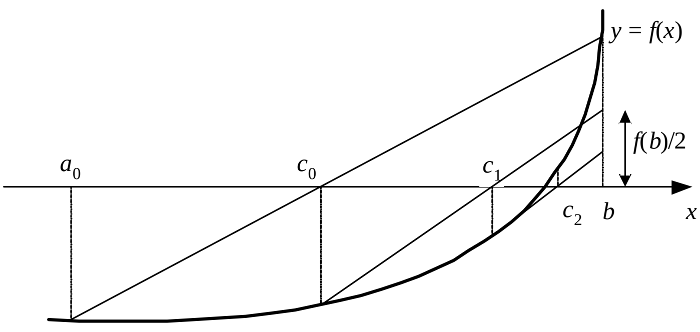
```
until (ABS(a-b) < epsilon1) or (ABS(f(c)) < epsilon2)
```

Bila perubahan ini diterapkan pada soal pencarian akar di atas dengan `epsilon2 = 0.000001`, lelarannya akan berhenti pada r = 12 dengan akar x = 0.605267.

Perbaikan Metode Regula-Falsi

Untuk mengatasi kemungkinan kasus titik mandek, metode regula-falsi kemudian diperbaiki (*modified false position method*). Caranya, pada akhir lelaran r = 0, kita sudah memperoleh selang baru akan dipakai pada lelaran r = 1. Berdasarkan selang baru tersebut, tentukan titik ujung selang yang tidak berubah (jumlah perulangan > 1) - yang kemudian menjadi titik mandek. Nilai f pada titik mandek itu diganti menjadi setengah kalinya, yang akan dipakai pada lelaran r = 1.

Misalkan fungsi $f(x)$ cekung ke atas di dalam selang $[a, b]$ seperti yang ditunjukkan pada Gambar 3.9.



Gambar 3.9 Perbaikan metode regula-falsi

Setelah menghitung nilai c_0 pada lelaran $r = 0$, ujung selang b untuk lelaran $r = 1$ tidak berubah. Titik b menjadi titik mandek. Karena itu, untuk lelaran $r = 1$, nilai $f(b)$ yang dipakai adalah $f(b)/2$. Begitu juga untuk lelaran $r = 2$, nilai $f(b)$ yang dipakai adalah setengah dari nilai $f(b)$ sebelumnya. Pada akhir lelaran $r = 2$, c_2 sudah terletak di bawah kurva $y = f(x)$. Selang yang dipakai selanjutnya adalah $[c_1, c_2]$. Dengan cara ini kita dapat menghilangan titik mandek yang berkepanjangan. Program 3.3 kita modifikasi menjadi Program 3.4.

Program 3.4 Metode regula-falsi yang diperbaiki

```
procedure perbaikan_regula_falsi(a, b: real);
{ Mencari akar f(x)=0 di dalam selang [a,b] dengan metode regula-falsi
  yang diperbaiki
  K.Awal : a dan b adalah ujung-ujung selang sehingga f(a)*f(b) < 0,
           harga a dan b sudah terdefenisi
  K.Akhir : akar persamaan tercetak di layar
}
const
   epsilon1 = 0.00001;   {batas lebar selang akhir lelaran}
   epsilon2 = 0.000001;       {batas galat nilai fungsi di hampiran akar}
var
   FA, FB, simpan : real;
   mandek_kiri, mandek_kanan : integer; {jumlah perulangan titik
                                         ujung selang}
begin
   FA:=f(a); FB:=f(b);
   mandek_kiri:=1; mandek_kanan:=1;
   repeat
        c:=b-(FB*(b-a)/(FB-FA));
        if abs(f(c)) < epsilon2 then  {f(c) = 0, c adalah akar}
          begin
            a:=c;
            b:=c;
         end
        else


          begin
             if f(a)*f(c) < 0 then
                 begin
                   b:=c        {selang baru [a,b]=[a,c]}
                   FB:=f(c);
                   mandek_kiri:=mandek_kiri + 1;
                   mandek_kanan:=0;
                   if mandek_kiri > 1 then
                        FA:=FA/2; {a menjadi titik mandek }
                 end
             else
              begin
                   a:=c;              {selang baru [a,b]=[c,b]}
                   FA:=f(c);
                   mandek_kanan:=mandek_kanan + 1;
                   mandek_kiri:=0;
                   if mandek_kanan > 1 then
                        FB:=FB/2; {b menjadi titik mandek}
             end;
```

```
        end;
    until (ABS(a-b)< epsilon1) OR (ABS(f(c)) < epsilon2);
    { c adalah taksiran akar )
    writeln('Hampiran akar : ', c:10:6);
end;
```

Tabel lelaran dari Program 3.4 untuk menghitung akar $f(x) = e^x - 5x^2$ di dalam selang [0, 1] dengan e = 0.00001 dan d = 0.000001 adalah sebagai berikut:

r	*a*	*c*	*b*	*f*(*a*)	*f*(*c*)	*f*(*b*)	Selang baru	Lebarnya
0	0.000000	0.304718	1.000000	1.000000	0.891976	-2.281718	[c,b]	0.695282
						(*/2)		
1	0.304718	0.609797	1.000000	0.891976	-0.019205	-1.140859	[a,c]	0.305079
2	0.304718	0.603367	0.609797	0.891976	0.008005	-0.019205	[c,b]	0.006430
3	0.603367	0.605259	0.609797	0.008005	0.000035	-0.019205	[c,b]	0.004538
						(*/2)		
4	0.605259	0.605275	0.609797	0.000035	-0.000035	-0.009602	[a,c]	0.000017
5	0.605259	0.605267	0.605275	0.000035	0.000000	-0.000035	[c,b]	0.000008

Hampiran akar *x* = 0.605267

Terlihat bahwa jumlah lelarannya berkurang menjadi sepertiga semula. Harus dicatat bahwa metode regula-falsi yang diperbaiki tetap berlaku untuk fungsi yang tidak cekung sekalipun. Jadi, jika anda memprogram dengan metode regula-falsi, pakailah Program 3.4 ini untuk semua kemungkinan kasus fungsi.

3.4 Metode Terbuka

Tidak seperti pada metode tertutup, metode terbuka tidak memerlukan selang yang mengurung akar. Yang diperlukan hanya sebuah tebakan awal akar atau dua buah tebakan yang tidak perlu mengurung akar. Inilah alasan mengapa metodenya dinamakan metode terbuka. Hampiran akar sekarang didasarkan pada hampiran akar sebelumnya melalui prosedur lelaran. Kadangkala lelaran konvergen ke akar sejati, kadangkala ia divergen. Namun, apabila lelarannya konvergen, konvergensinya itu berlangsung sangat cepat dibandingkan dengan metode tertutup.

Yang termasuk ke dalam metode terbuka:

1. Metode lelaran titik-tetap (*fixed-point iteration*)
2. Metode Newton-Raphson
3. Metode *secant*

3.4.1. Metode Lelaran Titik-Tetap

Metode ini kadang-kadang dinamakan juga metode lelaran sederhana, metode langsung, atau metode sulih beruntun. Kesederhanaan metode ini karena pembentukan prosedur lelarannya mudah dibentuk sebagai berikut:

Susunlah persamaan $f(x) = 0$ menjadi bentuk $x = g(x)$. Lalu, bentuklah menjadi prosedur lelaran

$$x_{r+1} = g(x_r) \tag{P.3.3}$$

dan terkalah sebuah nilai awal x_0, lalu hitung nilai x_1 , x_2 , x_3 , ..., yang mudah-mudahan konvergen ke akar sejati s sedemikian sehingga

$$f(s) = 0 \quad \text{dan } s = g(s).$$

Kondisi berhenti lelaran dinyatakan bila

$$|x_{r+1} - x_r| < e$$

atau bila menggunakan galat relatif hampiran

$$\left|\frac{x_{r+1} - x_r}{x_{r+1}}\right| < d$$

dengan e dan d telah ditetapkan sebelumnya. Program lelaran titik-tetap ditunjukkan oleh Program 3.5.

Program 3.5 Metode lelaran titik-tetap

```
procedure lelaran_titik_tetap(x:real);
{ mencari akar f(x) = 0 dengan metode lelaran titik-tetap
  K.Awal : x adalah tebakan awal akar, nilainya sudah terdefinisi
  K.Akhir: akar persamaan tercetak di layar
}
const
  epsilon = 0.000001;
var
  x_sebelumnya: real;

  function g(x:real): real;
  {mengembalikan nilai g(x). Definisikan g(x), ), lihat Contoh 3.2  }

begin
  repeat
     x_sebelumnya:=x;
```

```
    x:=g(x);
  until ABS(x-x_sebelumnya) < epsilon;
  { x adalah hampiran akar }

  write('Hampiran akar x = ', x:10:6);
end;
```

Program 3.5 hanya menangani lelaran yang konvergen. Program harus dimodifikasi menjadi Program 3.6 untuk menangani lelaran yang divergen. Salah satu cara penanganannya adalah dengan membatasi jumlah maksimum lelaran (*Nmaks*). Jika jumlah lelaran lebih besar dari *Nmaks*, maka diasumsikan lelarannya divergen.

Program 3.6 Metode lelaran titik-tetap (dengan penanganan kasus divergen)

```
procedure lelaran_titik_tetap(x:real);
{ mencari akar f(x) = 0 dengan metode lelaran titik-tetap
  K.Awal : x adalah tebakan awal akar, nilainya sudah terdefinisi
  K.Akhir: akar persamaan tercetak di layar
}
const
  epsilon = 0.000001;
  Nmaks = 30;
var
  x_sebelumnya: real;  { hampiran nilai akar pada lelaran sebelumnya }
  i : integer;         { pencacah jumlah lelaran }

  function g(x:real): real;
  {mengembalikan nilai g(x). Definisikan g(x) di sini, lihat Contoh 3.2 }

begin
  i:=0;
  repeat
    x_sebelumnya:=x;
    x:=g(x);
    i:=i+1;
  until (ABS(x-x_sebelumnya) < epsilon) or (i > Nmaks);
  { x adalah hampiran akar }

  if i > Nmaks then
    write('Divergen!')
  else
    write('Hampiran akar x = ', x:10:6);
end;
```

Contoh 3.2

Carilah akar persamaan $f(x) = x^2 - 2x - 3 = 0$ dengan metode lelaran titik-tetap. Gunakan e = 0.000001.

Penyelesaian:

Terdapat beberapa kemungkinan prosedur lelaran yang dapat dibentuk.

(a) $x^2 - 2x - 3 = 0$

$x^2 = 2x + 3$
$x = \sqrt{(2x + 3)}$

Dalam hal ini, $g(x) = \sqrt{(2x + 3)}$. Prosedur lelarannya adalah $x_{r+1} = \sqrt{(2x_r + 3)}$. Ambil terkaan awal x_0=4

Tabel lelarannya:

r	x_r	$\lvert x_{r+1} - x_r \rvert$
0	4.000000	-
1	3.316625	0.683375
2	3.103748	0.212877
3	3.034385	0.069362
4	3.011440	0.022945
5	3.003811	0.007629
6	3.001270	0.002541
7	3.000423	0.000847
8	3.000141	0.000282
9	3.000047	0.000094
10	3.000016	0.000031
11	3.000005	0.000010
12	3.000002	0.000003
13	3.000001	0.000001
14	3.000000	0.000000

Hampiran akar x = 3.000000 *(konvergen monoton)*

(b) $x^2 - 2x - 3 = 0$
$x(x-2) = 3$
$x = 3/(x - 2)$

Dalam hal ini, $g(x) = 3/(x - 2)$. Prosedur lelarannya adalah $x_{r+1} = 3/(x_r - 2)$. Ambil terkaan awal $x_0 = 4$

Tabel lelarannya:

i	x_r	$\lvert x_{r+1} - x_r \rvert$
0	4.000000	-
1	1.500000	2.500000
2	-6.000000	7.500000
3	-0.375000	5.625000
4	-1.263158	0.888158
5	-0.919355	0.343803
6	-1.027624	0.108269

7	-0.990876	0.036748
8	-1.003051	0.012175
9	-0.998984	0.004066
10	-1.000339	0.001355
11	-0.999887	0.000452
12	-1.000038	0.000151
13	-0.999987	0.000050
14	-1.000004	0.000017
15	-0.999999	0.000006
16	-1.000000	0.000002
17	-1.000000	0.000001

Hampiran akar x = -1.000000 *(konvergen berosilasi)*

(c) $x^2 - 2x - 3 = 0$

$$x = (x^2 - 3)/2$$

Prosedur lelarannya adalah $x_{r+1} = (x_r^2 - 3)/2$. Ambil terkaan awal x_0=4

Tabel lelarannya:

i	x_r	$\lvert x_{r+1} - x_r \rvert$
0	4.000000	-
1	6.500000	2.500000
2	19.625000	13.125000
3	191.070313	171.445312
4	18252.432159	18061.361847
	...	

Ternyata lelarannya divergen! ■

Contoh 3.3

Apa yang terjadi dengan pemilihan beragam nilai x_0 pada pencarian akar persamaan

$$x^3 + 6x - 3 = 0$$

dengan prosedur lelaran

$$x_{r+1} = \frac{-x_r^{\,3} + 3}{6} \qquad \text{[PUR84]}$$

Cobakan dengan: x_0= 0.5,
x_0= 1.5,
x_0= 2.2,
x_0= 2.7

Penyelesaian:

Tabel lelarannya adalah sebagai berikut:

r	x_r	r	x_r	r	x_r	r	x_r
0	0.5	0	1.5	0	2.2	0	2.7
1	0.4791667	1	-0.0625	1	-1.2744667	1	-2.7805
2	0.4816638	2	0.5000407	2	0.8451745	2	4.0827578
3	0.4813757	3	0.4791616	3	0.3993792	3	-10.842521
…	…	4	0.4816644	4	0.4893829	4	212.9416
7	0.4814056	…	…	…	…	5	-16909274.5
8	0.4814056	9	0.4814056	9	0.4814054		
		10	0.4814056	10	0.4814056		
				11	0.4814056		

Konvergen Divergen

Terlihat dengan pengambilan x_0 yang cukup dekat ke akar sejati, proses akan konvergen, tetapi jika kita mengambil x_0 terlalu jauh dari akar sejati, ia akan divergen. ■

Kadang-kadang lelaran konvergen, kadang-kadang ia divergen. Adakah suatu "tanda" bagi kita untuk mengetahui kapan suatu lelaran konvergen dan kapan divergen?

Kriteria konvergensi

Diberikan prosedur lelaran

$$x_{r+1} = g(x_r) \qquad \text{(P.3.4)}$$

Misalkan $x = s$ adalah solusi $f(x) = 0$ sehingga $f(s) = 0$ dan $s = g(s)$. Selisih antara x_{r+1} dan s adalah

$$\begin{aligned} x_{r+1} - s &= g(x_r) - s \\ &= \frac{g(x_r) - s}{(x_r - s)}(x_r - s) \end{aligned} \qquad \text{(P.3.5)}$$

Terapkan teorema nilai rata-rata pada persamaan (P.3.5) sehingga

$$x_{r+1} - s = g'(t)(x_r - s) \qquad \text{(P.3.6)}$$

yang dalam hal ini $x_{r+1} < t < s$. Misalkan galat pada lelaran ke-r dan lelaran ke-(r+1) adalah

$$e_r = x_r - s \text{ dan } e_{r+1} = x_{r+1} - s$$

Persamaan (P.4.6) dapat kita tulis menjadi

$$e_{r+1} = g'(t)\, e_r \qquad \text{(P.3.7)}$$

atau dalam tanda mutlak

$$|e_{r+1}| = |g'(t)|\, |e_r| \leq K |e_r|$$

Berapakah batas-batas nilai K itu?

Misalkan x_0 dan x berada di dalam selang sejauh $2h$ dari s, yaitu $s - h < x < s + h$. Jika lelaran konvergen di dalam selang tersebut, yaitu x_0, x_1, x_2, x_3, ... menuju s, maka galat setiap lelaran berkurang. Jadi, haruslah dipenuhi kondisi

$$|e_{r+1}| \leq K|e_r| \leq K^2|e_{r-1}| \leq K^3|e_{r-2}| \leq \ldots \leq K^{r+1}|e_0|$$

Kondisi tersebut hanya berlaku jika

$$g'(x) \leq K < 1$$

Karena $K < 1$, maka $K^{r+1} \to 0$ untuk $r \to \infty$; di sini $|x_{r+1} - s| \to 0$.

TEOREMA 3.2. Misalkan $g(x)$ dan $g'(x)$ menerus di dalam selang $[a,b] = [s-h, s+h]$ yang mengandung titik tetap s dan nilai awal x_0 dipilih dalam selang tersebut. Jika $|g'(x)| < 1$ untuk semua $x \in [a, b]$ maka lelaran $x_{r+1} = g(x_r)$ akan konvergen ke s. Pada kasus ini s disebut juga *titik atraktif*. Jika $|g'(x)| > 1$ untuk semua $x \in [a, b]$ maka lelaran $x_{r+1} = g(x_r)$ akan divergen dari s.

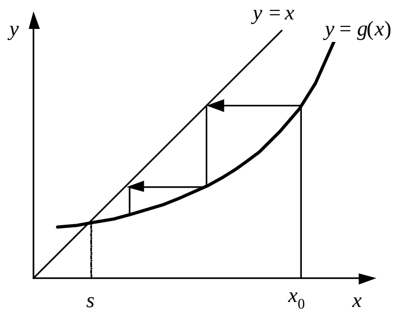
Teorema 3.2 dapat kita sarikan sebagai berikut:

Di dalam selang $I = [s-h, s+h]$, dengan s titik tetap,

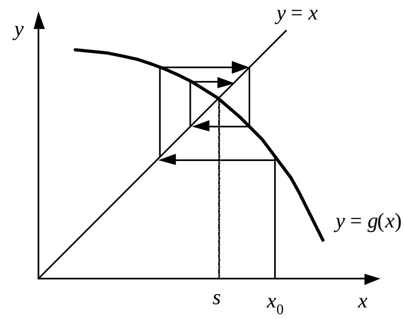
jika $0 < g'(x) < 1$ untuk setiap $x \in I$, maka lelaran *konvergen monoton;*
jika $-1 < g'(x) < 0$ untuk setiap $x \in I$, maka lelaran *konvergen bersosilasi;*
jika $g'(x) > 1$ untuk setiap $x \in I$, maka lelaran *divergen monoton;*
jika $g'(x) < -1$ untuk setiap $x \in I$, maka lelaran *divergrn berosilasi*.

Semuanya dirangkum seperti pada Gambar 3.10..

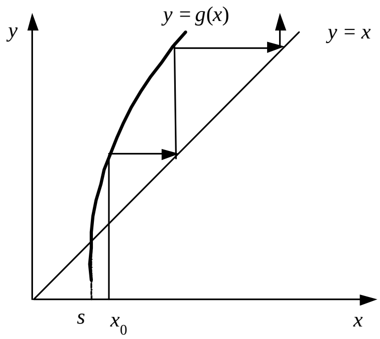
Sebagai catatan, keadaan $|g'(x)| = 1$ tidak didefinisikan. Catat juga bahwa semakin dekat nilai $|g'(x)|$ ke nol di dekat akar, semakin cepat kekonvergenan metode lelaran titik-tetap ini [PUR84].



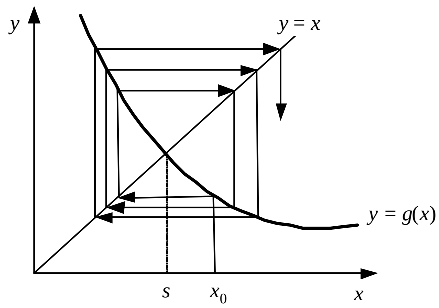
(a) Konvergen monoton: $0 < g'(x) < 1$



(b) Konvergen berosilasi: $-1 < g'(x) < 0$



(c) Divergen monoton: $g'(x) > 1$



(d) Divergen berosilasi: $g'(x) < -1$

Gambar 3.10 Jenis-jenis kekonvergenan

Sekarang, mari kita analisis mengapa pencarian akar persamaan $x^2 - 2x - 3 = 0$ pada Contoh 3.2 dan pencarian akar persamaan $x^3 + 6x - 3 = 0$ pada Contoh 3.3 dengan bermacam-macam prosedur lelaran dan tebakan awal kadang-kadang konvergen dan kadang-kadang divergen.

(i) Prosedur lelaran pertama: $x_{r+1} = \sqrt{2x_r + 3}$

$$g(x) = \sqrt{(2x+3)}$$

$$g'(x) = \frac{1}{2\sqrt{(2x+3)}}$$

Terlihat bahwa $|g'(x)| < 1$ untuk x di sekitar titik-tetap $s = 3$. Karena itu, pengambilan tebakan awal $x_0 = 4$ akan menghasilkan lelaran yang konvergen sebab $|g'(4)| = \left|1/[2\sqrt{(8+3)}\right| = 0.1508 < 1$.

(ii) Prosedur lelaran kedua: $x_{r+1} = 3/(x_r - 2)$

$g(x) = 3/(x-2)$

$g'(x) = -3/(x-2)^2$

Terlihat bahwa $|g'(x)| < 1$ untuk x di sekitar titik-tetap $s = 3$. Karena itu, pengambilan tebakan awal $x_0 = 4$ akan menghasilkan lelaran yang konvergen sebab

$|g'(4)| = |-3/(4-2)^2| = 0.75 < 1$.

(iii) Prosedur lelaran ketiga $x_{r+1} = (x_r^2 - 3)/2$

$g(x) = (x^2 - 3)/2$

$g'(x) = x$

Terlihat bahwa $|g'(x)| > 1$ untuk x di sekitar titik-tetap $s = 3$. Karena itu, pengambilan tebakan awal $x_0 = 4$ akan menghasilkan lelaran yang divergen sebab

$|g'(4)| = |4| = 4 > 1$.

(iv) Prosedur lelaran pada Contoh 3.3: $x_{r+1} = (-x_r^{\ 3} + 3)/6$

$g(x) = (-x^{3} + 3)/6$

$g'(x) = -x^2/2$

Terlihat bahwa $|g'(x)| < 1$ untuk x di sekitar titik-tetap $s = 0.48$. Pemilihan $x_0 = 0.5$ akan menjamin lelaran konvergen sebab $|g'(x_0)| < 1$. Untuk $x_0 = 1.5$ dan $x_0 = 2.2$ memang nilai $|g'(x_0)| > 1$ tetapi lelarannya masih tetap konvergen, namun $x_0 = 2.7$ terlalu jauh dari titik-tetap sehingga lelarannya divergen. Dapatkah kita menentukan batas-batas selang yang menjamin prosedur lelaran akan konvergen di dalamnya? Temukan jawabannya pada Contoh 3.4 di bawah ini.

Contoh 3.4

Pada Contoh 3.3 di atas, tentukan selang sehingga prosedur lelaran

$$x_{r+1} = (-x_r^{\ 3} + 3)/6$$

konvergen?

Penyelesaian:

$g(x) = (-x^{\ 3} + 3)/6$
$g\,'(x) = -x^2/2$

Syarat konvergen adalah $|g'(x)| < 1$. Jadi,

$\Leftrightarrow\ |-x^2/2| < 1$

$\Leftrightarrow\ -1 < -x^2/2 < 1$

$\Leftrightarrow\ 2 > x^2 > -2$

$\Leftrightarrow\ -2 < x^2 < 2$

Urai satu per satu:

(i) $x^2 > -2$ { tidak ada x yang memenuhi)

(ii) $x^2 < 2$, dipenuhi oleh

$\Leftrightarrow\ x^2 - 2 < 0$

$\Leftrightarrow\ -\sqrt{2} < x < \sqrt{2}$

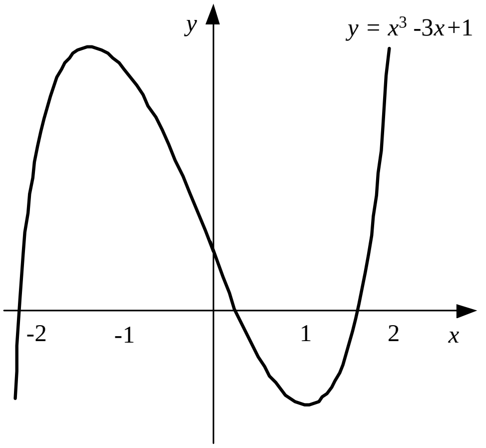
Jadi, prosedur lelaran $x_{r+1} = (-x_r^{\ 3} + 3)/6$ konvergen di dalam selang $-\sqrt{2} < x < \sqrt{2}$. Kita dapat memilih x_0 dalam selang tersebut yang menjamin lelaran akan konvergen. ■

Contoh 3.5

Gunakan metode lelaran titik-tetap untuk mencari akar persamaan

$x^3 - 3x + 1$ dalam selang [1, 2]   [PUR84]

Catatan : selang [1, 2] ini sebenarnya tidak digunakan dalam proses lelaran sebagaimana halnya pada metode bagidua. Selang ini diberikan untuk memastikan bahwa suatu prosedur lelaran titik-tetap konvergen di dalamnya. Kurva fungsi $y = x^3 - 3x + 1$ diperlihatkan pada Gambar 3.11.



Gambar 3.11 Kurva $y = x^3 - 3x + 1$

Penyelesaian:

(i) $x_{r+1} = (x_r^3 + 1)/3$

Tetapi, karena $|g'(x)| = |x^2| > 1$ dalam selang [1, 2], maka prosedur lelaran ini tidak digunakan.

(ii) $x_{r+1} = -1/(x_r^2 - 3)$

Tetapi, karena $|g'(x)| = |2x/(x^2 - 3)^3| > 1$ dalam selang [1, 2], maka prosedur lelaran ini tidak digunakan.

(iii) $x_{r+1} = 3/x_r - 1/x_r^2$

Ternyata $|g'(x)| = |(-3x + 2)/x^3| \leq 1$ di dalam selang [1, 2], yaitu, $g'(x)$ naik dari $g'(1) = -1$ ke $g'(2) = -1/2$. Jadi, $|g'(x)|$ lebih kecil dari 1 dalam selang [1, 2]. Dengan mengambil $x = 1.5$, prosedur lelarannya konvergen ke akar $x = 1.5320889$ seperti pada tabel berikut ini.

r	*x*
0	1.5
1	1.5555556
2	1.5153061
…	...
43	1.5320888
44	1.5320889
45	1.5320889

■

Contoh 3.5 menunjukkan bahwa ada dua hal yang mempengaruhi kekonvergenan prosedur lelaran:

1. Bentuk formula $x_{r+1} = g(x_r)$
2. Pemilihan tebakan awal x

Catatan: Meskipun $|g'(x)| > 1$ menyatakan lelaran divergen, tetapi kita harus hati-hati dengan pernyataan ini. Sebabnya, walaupun x_r divergen dari suatu akar, runtunan lelarannya mungkin konvergen ke akar yang lain. Kasus seperti ini ditunjukkan pada Contoh 3.6 di bawah ini.

Contoh 3.6

Tentukan akar persamaan $f(x) = x^2 - 4x + 3 = 0$ dengan prosedur lelaran

$$x_{r+1} = (x_r^2 + 3)/4$$

Penyelesaian:

Jika prosedur lelaran $x_{r+1} = (x_r^2 + 3)/4$ konvergen ke titik-tetap s, maka

$$\lim_{r \to \infty} x_r = s$$

sehingga

$$s = (s^2 + 3)/4$$
$$s^2 - 4s + 3 = 0$$
$$(s - 3)(s - 1) = 0$$

yang memberikan $s_1 = 1$ atau $s_2 = 3$. Jadi, lelaran konvergen ke akar $x = 1$ atau akar $x = 3$. Dari

$$g(x) = (x^2 + 3)/4$$

diperoleh

$$g'(x) = x/2$$

Gambarkan kurva $y = x$ dan $y = (x^2 + 3)/4$ seperti pada Gambar 3.12.

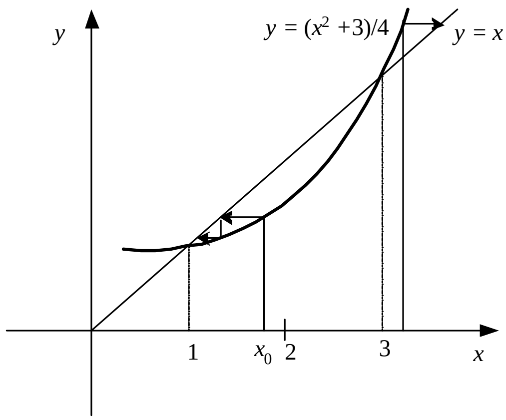
Prosedar lelaran akan konvergen bila

$$\Leftrightarrow |g'(x)| > 1$$
$$\Leftrightarrow -1 < x/2 < 1$$

atau

$$-2 < x < 2$$

Sehingga pemilihan x_0 dalam selang $-2 < x < 2$ menjamin lelaran konvergen ke akar $x = 1$. Dari Gambar 3.12 terlihat bahwa lelaran juga konvergen ke akar $x = 1$ untuk pemilihan x_0 dalam selang $2 < x < 3$. Padahal, kalau dihitung, dalam selang $2 < x < 3$, $|g'(x)| > 1$ yang menyatakan bahwa lelarannya divergen. Lelaran divergen dari akar $x = 3$ tetapi konvergen ke akar $x = 1$. ■



Gambar 3.12 Kurva $y = x$ dan $y = (x^2 + 3)/4$

Sebagai contoh terakhir metode lelaran titik-tetap, mari kita hitung akar fungsi pada Contoh 3.1, yaitu $f(x) = e^x - 5x^2$.

Contoh 3.7

Hitunglah akar $f(x) = e^x - 5x^2$ dengan metode lelaran titik-tetap. Gunakan $\varepsilon = 0.00001$. Tebakan awal akar $x_0=1$.

Penyelesaian:

Salah satu prosedur lelaran yang dapat dibuat adalah

$$e^x - 5x^2 = 0$$
$$e^x = 5x^2$$
$$x = \sqrt{e^x/5}$$
$$x_{r+1} = \text{SQRT}(\text{EXP}(x_r)/5)$$

Tabel lelarannya:

i	x_r	$\lvert x_{r+1} - x_r \rvert$
0	0.500000	-
1	0.574234	0.074234