Fakultas Informatika
School of Computing
Telkom University

CNH2B4 / KOMPUTASI NUMERIK

TIM DOSEN

KK MODELING AND COMPUTATIONAL EXPERIMENT

3 SOLUSI PERSAMAAN NONLINEAR: METODE ITERASI TITIK-TETAP, NEWTON-RAPHSON, DAN SECANT

SOLUSI PERSAMAAN NONLINEAR

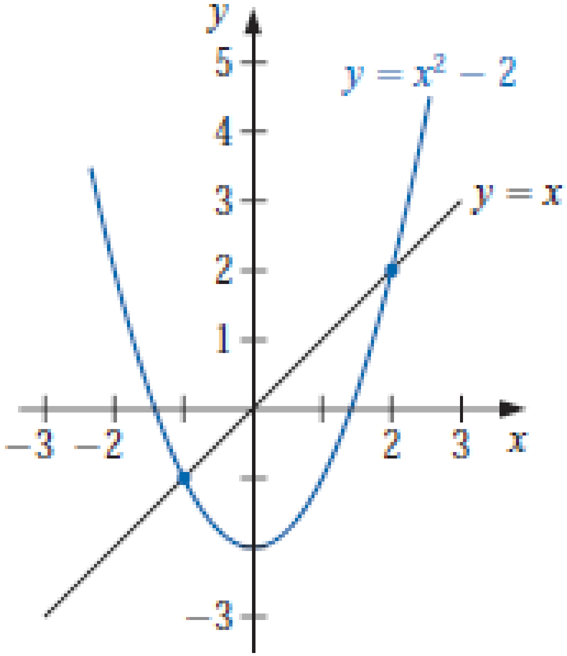
- Metode Bagidua
- Metode Regula Falsi (False Position)
- Metode Iterasi Titik-Tetap
- Metode Newton-Raphson
- Metode Secant

Metode Iterasi Titik-Tetap

Titik *p* disebut titik tetap untuk fungsi *g* jika *g*(*p*) = *p*.

Tentukan semua titik tetap dari $g(x) = x^2 - 2$.

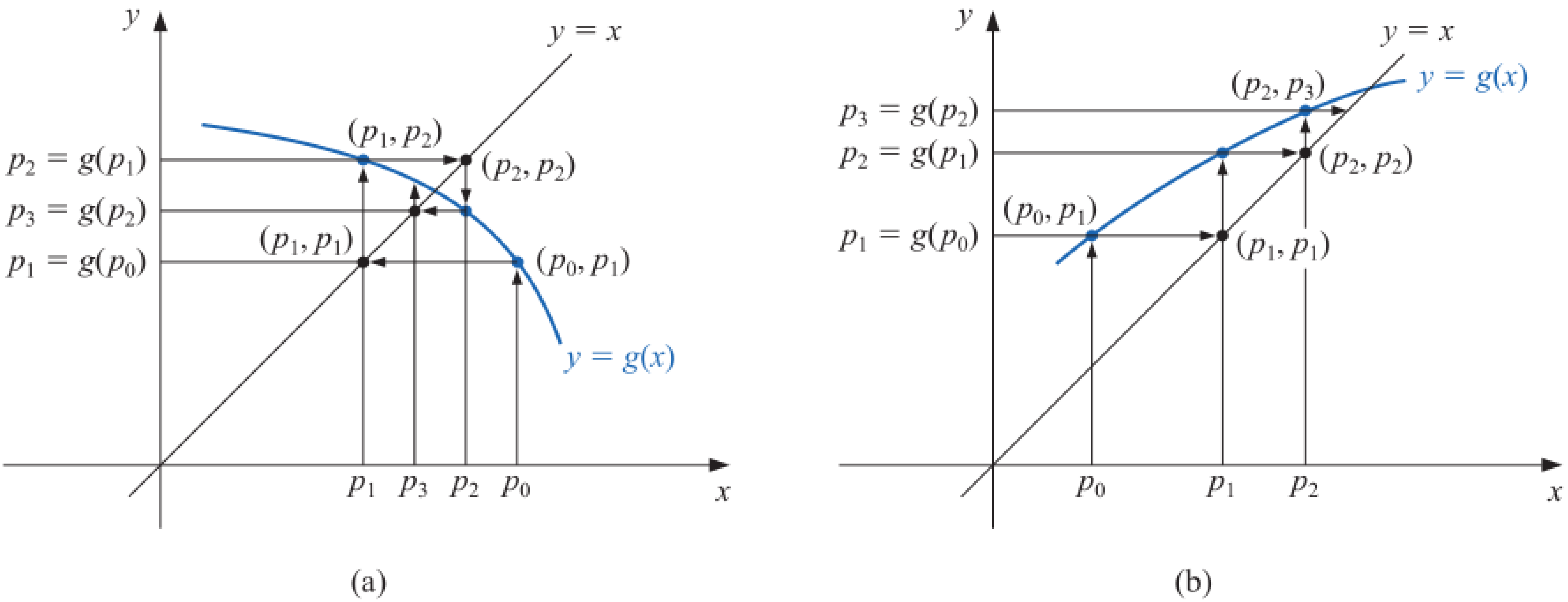
Jawab:



Metode Iterasi Titik-Tetap

- Bentuk persamaan $x = g(x)$ dari persamaan $f(x) = 0$.
- Tentukan tebakan awal p_{0}.
- Buat prosedur iterasi $p_{n+1} = g(p_n)$.
- Akar ditemukan jika $f(p) = 0$ atau $p = g(p)$.
- Kondisi berhenti $|p_{n+1} - p_n| < TOL$.
- Solusi dari metode Iterasi Titik Tetap sangat bergantung pada pemilihan fungsi g(x).
- Pemilihan fungsi g(x) yang kurang tepat akan menyebabkan solusi divergen.

Metode Iterasi Titik-Tetap



Burden, Richard L., and J. Douglas Faires.
Numerical Analysis. Brooks/Cole, *USA*, 2001.

Algoritma Iterasi Titik-Tetap

Fixed-Point Iteration

To find a solution to $p = g(p)$ given an initial approximation p_0:

INPUT initial approximation p_0; tolerance TOL; maximum number of iterations N_0.

OUTPUT approximate solution p or message of failure.

Step 1 Set $i = 1$.

Step 2 While $i \leq N_0$ do Steps 3–6.

  Step 3 Set $p = g(p_0)$. (*Compute* p_i.)

  Step 4 If $|p - p_0| < TOL$ then
    OUTPUT (p); (*The procedure was successful.*)
    STOP.

  Step 5 Set $i = i + 1$.

  Step 6 Set $p_0 = p$. (*Update* p_0.)

Step 7 OUTPUT ('The method failed after N_0 iterations, $N_0 =$', N_0);
  (*The procedure was unsuccessful.*)
  STOP.

Burden, Richard L., and J. Douglas Faires. *Numerical Analysis*. Brooks/Cole, *USA*, 2001.

Contoh Kasus (1)

Tentukan akar dari f(x) = x^3 - 2x^2 – 5 dengan tebakan awal p_0 = 2 menggunakan metode Iterasi Titik Tetap.

n	p_n	f(p_n)	g(p_n)	error
0	2.000000	-5.000000	2.351335	
1	2.351335	-3.057550	2.522860	0.351335
2	2.522860	-1.672092	2.607554	0.171525
3	2.607554	-0.869035	2.649480	0.084694
4	2.649480	-0.440814	2.670249	0.041926
...	....	....	....	....
14	2.690611	-0.000403	2.690629	0.000037
15	2.690629	-0.000200	2.690638	0.000019
16	2.690638	-0.000099	2.690643	0.000009

akar dari f(x) = x^3 - 2x^2 - 5 adalah x = 2.690638

Contoh Kasus (2)

Tentukan akar dari f(x) = x^4 + 2x^2 – x - 3 dengan tebakan awal p_0 = 0 menggunakan metode Iterasi Titik Tetap.

n	p_n	f(p_n)	g(p_n)	error
0	0.000000	-3.000000	1.224745	
1	1.224745	1.025255	1.098667	1.224745
2	1.098667	-0.227510	1.130491	0.126078
3	1.130491	0.058838	1.122524	0.031824
4	1.122524	-0.014651	1.124524	0.007967
5	1.124524	0.003683	1.124022	0.002000
6	1.124022	-0.000924	1.124148	0.000502
7	1.124148	0.000232	1.124117	0.000126
8	1.124117	-0.000058	1.124125	0.000032
9	1.124125	0.000015	1.124123	0.000008

akar dari f(x) = x^4+2x^2-x-3 adalah x = 1.124125

Metode Newton-Raphson

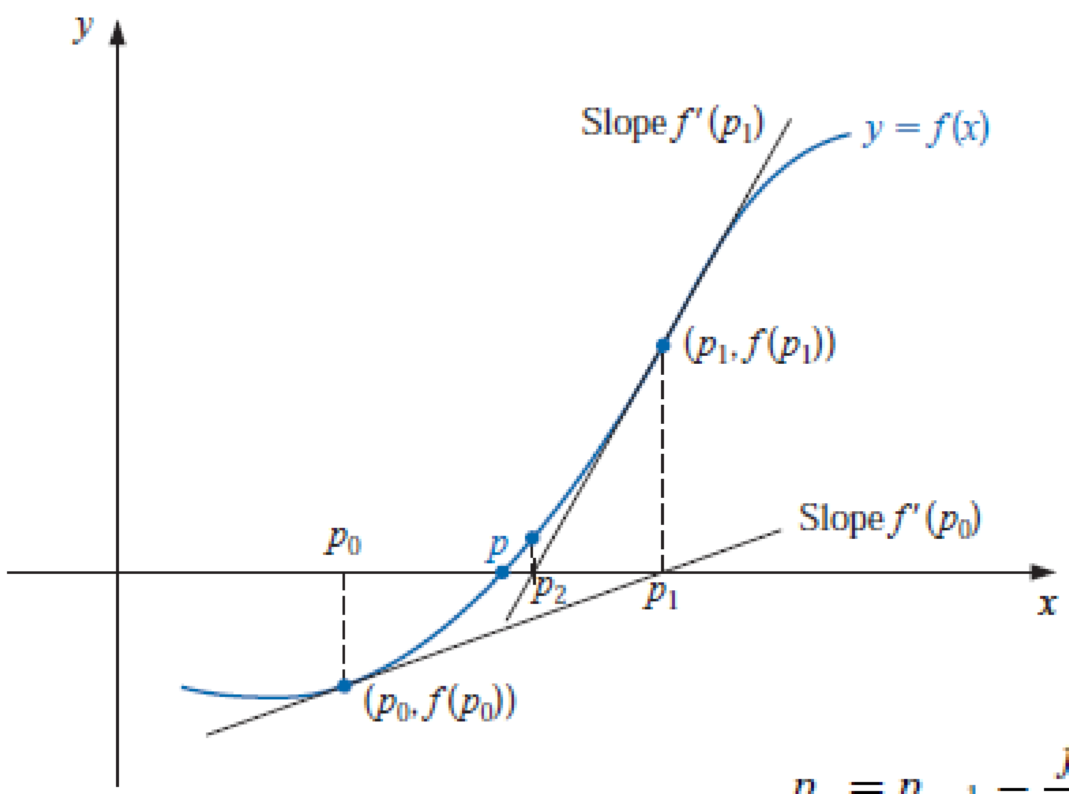
- Prosedur pencarian akar mirip dengan iterasi titik tetap
- Prosedur iterasi yang digunakan didapatkan melalui Deret Taylor:

$$f(p) = f(p_0) + (p - p_0)f'(p_0) + \frac{(p - p_0)^2}{2} f''(\xi(p)),$$

$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \quad \text{for } n \geq 1.$$

- Kondisi berhenti yang digunakan sama dengan pada iterasi titik-tetap

Metode Newton-Raphson



$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \quad \text{for } n \geq 1.$$

Burden, Richard L., and J. Douglas Faires. *Numerical Analysis*. Brooks/Cole, *USA*, 2001.

Algoritma Newton-Raphson

Newton's

To find a solution to $f(x) = 0$ given an initial approximation p_0:

INPUT initial approximation p_0; tolerance TOL; maximum number of iterations N_0.

OUTPUT approximate solution p or message of failure.

Step 1 Set $i = 1$.

Step 2 While $i \le N_0$ do Steps 3–6.

Step 3 Set $p = p_0 - f(p_0)/f'(p_0)$. (*Compute* p_i.)

Step 4 If $|p - p_0| < TOL$ then
OUTPUT (p); (*The procedure was successful.*)
STOP.

Step 5 Set $i = i + 1$.

Step 6 Set $p_0 = p$. (*Update* p_0.)

Step 7 OUTPUT ('The method failed after N_0 iterations, $N_0 =$', N_0);
(*The procedure was unsuccessful.*)
STOP. ■

Burden, Richard L., and J. Douglas Faires. *Numerical Analysis*. Brooks/Cole, *USA*, 2001.

Contoh Kasus (1)

Tentukan akar dari f(x) = x^3 - 2x^2 – 5 dengan tebakan awal p_0 = 2 menggunakan metode Newton-Raphson.

n	p_n	f(p_n)	f'(p_n)	error
0	2.000000	-5.000000	4.000000	
1	3.250000	8.203125	18.687500	1.250000
2	2.811037	1.408754	12.461636	0.438963
3	2.697990	0.080768	11.045484	0.113047
4	2.690677	0.000325	10.956522	0.007312
5	2.690647	0.000000	10.956161	0.000030
6	2.690647	0.000000	10.956161	0.000000

akar dari f(x) = x^3 - 2x^2 - 5 adalah x = 2.690647

Contoh Kasus (2)

Tentukan akar dari f(x) = sin(x) – exp(-x) dengan tebakan awal p_0 = 3.5 metode Newton-Raphson.

n	p_n	f(p_n)	f'(p_n)	error
0	3.500000	-0.380981	-0.906259	
1	3.079612	0.015964	-0.952103	0.420388
2	3.096379	-0.000014	-0.953765	0.016767
3	3.096364	-0.000000	-0.953764	0.000015
4	3.096364	-0.000000	-0.953764	0.000000

akar dari f(x) = sin(x) - exp(-x) adalah x = 3.096364

Metode Secant

- Tidak semua fungsi dapat dihitung dengan mudah turunannya.
- Turunan pada metode Newton-Raphson dihitung dengan menggunakan suatu hampiran:

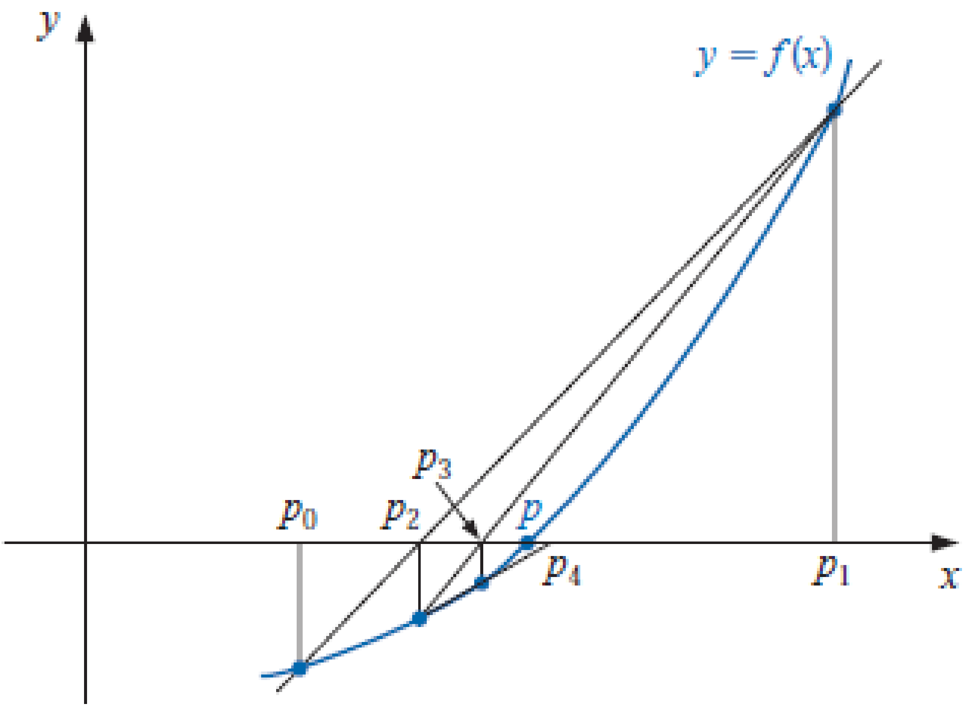
$$f'(p_{n-1}) \approx \frac{f(p_{n-2}) - f(p_{n-1})}{p_{n-2} - p_{n-1}} = \frac{f(p_{n-1}) - f(p_{n-2})}{p_{n-1} - p_{n-2}}.$$

- Prosedur iterasi metode Secant adalah

$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}.$$

- Kondisi berhenti menggunakan ketentuan yang sama dengan Newton-Raphson.

Metode Secant



$$p_n = p_{n-1} - \frac{f(p_{n-1})(p_{n-1} - p_{n-2})}{f(p_{n-1}) - f(p_{n-2})}.$$

Burden, Richard L., and J. Douglas Faires. *Numerical Analysis*. Brooks/Cole, *USA*, 2001.

Algoritma Metode Secant

Secant

To find a solution to $f(x) = 0$ given initial approximations p_0 and p_1:

INPUT initial approximations p_0, p_1; tolerance TOL; maximum number of iterations N_0.

OUTPUT approximate solution p or message of failure.

Step 1 Set $i = 2$;
$q_0 = f(p_0)$;
$q_1 = f(p_1)$.

Step 2 While $i \leq N_0$ do Steps 3–6.

Step 3 Set $p = p_1 - q_1(p_1 - p_0)/(q_1 - q_0)$. (*Compute* p_i.)

Step 4 If $|p - p_1| < TOL$ then
OUTPUT (p); (*The procedure was successful.*)
STOP.

Step 5 Set $i = i + 1$.

Step 6 Set $p_0 = p_1$; (*Update* p_0, q_0, p_1, q_1.)
$q_0 = q_1$;
$p_1 = p$;
$q_1 = f(p)$.

Step 7 OUTPUT ('The method failed after N_0 iterations, $N_0 =$', N_0);
(*The procedure was unsuccessful.*)
STOP. ■

Burden, Richard L., and J. Douglas Faires. *Numerical Analysis*. Brooks/Cole, *USA*, 2001.

Contoh Kasus (1)

Tentukan akar dari f(x) = x^3 - 2x^2 – 5 dengan tebakan awal p_0 = 2 dan p_1 = 4 menggunakan metode Secant.

n	p_n	f(p_n)	error
0	2.000000	-5.000000	
1	4.000000	27.000000	
2	2.312500	-3.328857	1.687500
3	2.497718	-1.894940	0.185218
4	2.742486	0.584403	0.244768
5	2.684792	-0.063949	0.057694
6	2.690482	-0.001810	0.005691
7	2.690648	0.000006	0.000166
8	2.690647	-0.000000	0.000001
9	2.690647	-0.000000	0.000000

akar dari f(x) = x^3 - 2x^2 - 5 adalah x = 2.690647

Contoh Kasus (2)

Tentukan akar dari f(x) = sin(x) – exp(-x) dengan tebakan awal p_0 = 3 dan p_1 = 4 menggunakan metode Secant.

```
-------------------------------------------
 n      p_n          f(p_n)        error
-------------------------------------------
 0    3.000000    0.091333
 1    4.000000   -0.775118
 2    3.105410   -0.008632    0.894590
 3    3.095336    0.000980    0.010074
 4    3.096364    0.000000    0.001028
 5    3.096364   -0.000000    0.000000
 6    3.096364   -0.000000    0.000000
-----------------------------------
akar dari f(x) = sin(x) - exp(-x) adalah x =   3.096364
```

Fakultas Informatika
School of Computing
Telkom University

THANK YOU