

**PEMAHAMAN METODE NUMERIK
MENGUNAKAN PEMROGRAMAN MATLAB
(Studi Kasus : Metode Secant)**

Melda Panjaitan

Jurusan Teknik Informatika

STMIK Budi Darma, Jln.SM.Raja No.338 Sp.Limun, Medan Sumatera Utara

meldapjt.78@gmail.com

Abstrak - Metode numerik merupakan alat bantu pemecahan masalah matematika yang sangat ampuh. dengan metode numerik, kita memperoleh solusi yang menghampiri atau mendekati solusi sejati sehingga solusi numerik dinamakan juga solusi hampiran (approxomation) atau solusi pendekatan, namun solusi hampiran dapat dibuat seteliti yang kita inginkan. Solusi hampiran jelas tidak tepat sama dengan solusi sejati, sehingga ada selisih antara keduanya. Selisih inilah yang disebut dengan galat (error). solusi dengan menggunakan metode numerik selalu berbentuk angka. Metode secant memerlukan dua taksiran awal yang harus mengurung akar persamaan.

Kata kunci - Metode Numerik, Metode Secant

Abstract - The numerical method is a very powerful tool for solving math problems. By numerical methods, we find solutions approaching or nearing true solutions so that numerical solutions are called approximate solutions or approximate solutions, but almost the solutions can be made as carefully as we want them to be. Solutions are almost certainly not exactly the same as true solutions, so there is a difference between the two. This difference is called an error (error). Solutions using numerical methods are always numerical. The secant method requires two initial estimates that must enclose the root of the equation.

Keywords - Numerical Method, Secant Method

I. PENDAHULUAN

Untuk pencarian akar-akar persamaan pada persamaan polinomial derajat dua, persamaan dapat

diselesaikan dengan rumus ABC (misalnya bentuk: $ax^2 + bx + c = 0$, persamaan ini dapat dicari akar-akarnya secara analitis):

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \dots\dots (1)$$

Sedang untuk persamaan polinomial derajat tiga atau empat, rumus-rumus yang ada sangatlah kompleks dan jarang sekali digunakan, sedang untuk persamaan dengan derajat yang lebih tinggi tidak ada rumus yang dapat digunakan untuk menyelesaikannya.

Metode numerik memberikan cara-cara untuk menyelesaikan bentuk persamaan tersebut secara perkiraan hingga didapat hasil yang mendekati penyelesaian secara benar (eksak). Penyelesaian numerik dilakukan dengan perkiraan yang berurutan (iterasi), maka tiap hasil akan lebih teliti dari perkiraan sebelumnya. Dengan berbagai iterasi yang dianggap cukup, akan didapat hasil perkiraan yang mendekati hasil yang benar (eksak) dengan toleransi yang diijinkan.

Cara lain yaitu dengan cara coba banding, yaitu dengan mencoba nilai x sembarang kemudian dievaluasi apakah nilai $f(x) = 0$, jika nilai x tidak sama dengan nol lalu dicoba nilai x yang lain, cara ini diulang terus menerus hingga didapat nilai $f(x) = 0$, untuk suatu nilai x tertentu, yang merupakan akar dari persamaan yang diselesaikan.

Metode *Secant* ini hampir sama dengan metode Newton-Raphson, namun pada metode ini kita harus membuat dua batas sebagai nilai awal dan nilai akhir seperti pada metode *bisection*, secara gafis dapat dilihat pada gambar berikut :

Dengan metode interpolasi kita dapatkan :

$$c = b - \frac{f(b)}{f(b)-f(a)} \dots\dots (2)$$

Karena

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

..... (3)

Pada dasarnya metode ini sama dengan metode Newton-Raphson, perbedaannya hanya terletak pada pendekatan untuk turunan pertama dari f saja. Pendekatan f' pada metode *Secant* didekati dengan ungkapan beda hingga yang didasarkan pada taksiran akar sebelumnya (beda mundur), yaitu metode *Secant* tidak memerlukan dua taksiran awal yang harus mengurung akar persamaan. Dalam beberapa kasus *swapping* dua taksiran awal x_1 dan x_2 dapat mengubah perilaku metode tersebut dari konvergen menjadi divergen.

II. METODE PENELITIAN

A. Algoritma Metode *Secant*

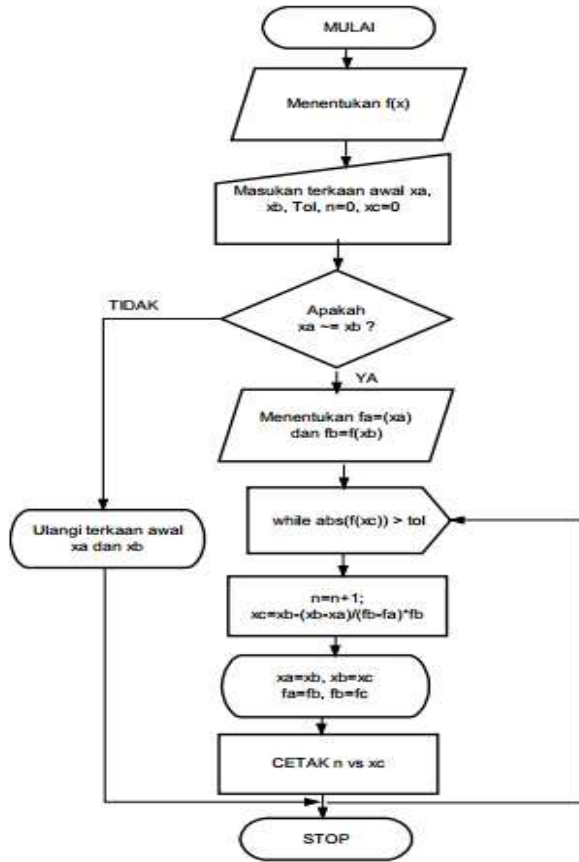
Algoritma *Metode Secant* dapat dilakukan dengan cara :

1. Berikan dua terkaan awal X_a dan X_b

2. Hitung X_c dengan cara :

$$X_c = X_b - \frac{X_b - X_a}{f(X_b) - f(X_a)} f(X_b)$$

3. Set $X_a = X_b$, $f_a = f_b$ dan $X_b = X_c$, $f_b = f_c$



Gambar 2.1 Diagram alir Metode Secant

B. Kriteria Terminasi dan Taksiran Kesalahan

Mengembangkan suatu kriteria objektif untuk menentukan kapan metode ini berhenti. Kita memerlukan suatu taksiran kesalahan yang tidak ditentukan oleh pengetahuan tentang akar itu sebelumnya. Suatu kesalahan aproksimasi dapat dihitung

$$|\epsilon_a| = \left| \frac{x_r \text{ baru} - x_r \text{ lama}}{x_r \text{ baru}} \right| \times 100\%$$

Dimana x_r baru adalah akar dari iterasi sekarang, dan x_r lama adalah akar iterasi sebelumnya.

Harga absolut dipakai karena kita biasanya cenderung memakai besarnya ϵ_a ketimbang tandanya. Bila $|\epsilon_a|$ menjadi lebih kecil daripada suatu kriteria penghentian praspesifikasi ϵ_s , komputasi dihentikan.

Metode ini paling banyak digunakan dalam mencari akar-akar persamaan, jika perkiraan awal dari akar adalah x_i , maka suatu garis singgung dapat dibuat dari titik $(x_i, f(x_i))$. Titik dari garis singgung tersebut memotong sumbu-x, biasanya memberikan perkiraan yang lebih dekat dari nilai akar.

$$f'(x_i) = \frac{f(x_i) - 0}{x_i - x_{i+1}} \dots (2)$$

$$\text{atau } x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)} \dots (3)$$

C. Galat

Di dalam pemakaian praktis, penyelesaian akhir yang diperlukan berbentuk numerik. Misalnya, set dari tabulasi data yang diberikan dan kesimpulan-kesimpulan yang dimiliki gambar dari data tersebut atau penyelesaian suatu sistem persamaan linear yang diberikan. Metode yang ditempuh dengan melibatkan angka tertentu dikenal dengan metode numerik.

Tujuan dari metode numerik adalah memberikan metode-metode yang efisien untuk memperoleh jawaban numerik dari bermacam-macam permasalahan. Untuk menyelesaikan suatu masalah biasanya dimulai dengan sebarang data awal kemudian dihitung, selanjutnya dengan langkah-langkah

(pengolahan) tertentu, akhirnya diperoleh suatu penyelesaian.

Data numerik adalah suatu aproksimasi (taksiran) yang sesuai sampai dengan dua, tiga, atau lebih tempat desimal. Kadang metode yang digunakanpun, adalah suatu aproksimasi. Oleh sebab itu galat dalam hasil perhitungan mungkin disebabkan oleh galat data, galat di dalam pemakaian suatu metode, atau kedua-duanya. Dalam bagian ini akan dibicarakan ide dasar tentang galat.

D. Toleransi

Dalam menyikapi galat yang dijumpai perlu adanya batasan nilai galat yang diterima yang disebut dengan nilai toleransi. Toleransi biasa didefinisikan sebagai batas penerimaan suatu galat. Dari pengertian ini yang dimaksud dengan Toleransi Galat Mutlak adalah nilai mutlak dari selisih nilai eksak (nilai sebenarnya) dengan nilai aproksimasi.

III. HASIL DAN PEMBAHASAN

A. Desain Dan Implementasi

1. Algoritma

Input :

- a. Masukkan persamaan non-linear $f(x)$ dan $f'(x)$.
- b. Masukkan toleransi yang diinginkan dalam persen (%). Toleransi merupakan batas kesalahan (galat) yang diinginkan, semakin mendekati nilai 0 semakin baik.
- c. Masukkan maksimum iterasi yang diinginkan. Iterasi awal = 0

d. Masukkan nilai pendekatan awal x_0 .

Proses :

- a. Hitung x_0
Dengan metode Newton Raphson
- b. Nilai *iterasi* = *iterasi + 1* atau $i = i + 1$
- c. Hitung nilai x_1 dengan kembali ke langkah a
- d. Hitung nilai galat E_a
- e. Jika iterasi < maksimum iterasi lanjutkan proses, jika tidak proses berhenti.
- f. Jika nilai $E_a < E_s$, lanjutkan ke proses selanjutnya
Jika nilai $E_a > E_s$, kembali ke proses

Output :

- a. Tampilkan tabel iterasi, x, f(x), $f'(x)$, galat.
- b. Tampilkan akar persamaan.
- c. Tampilkan grafik.

3.2 Implementasi

```
clear;
clc;
bawah = input ('Batas Bawah : ');
atas = input ('Batas Atas : ');
nilai = 1;
no = 0;
galat = 0.001;
m0=0;
fprintf ('Taksiran Batas bawah : %5.3f\n', bawah);
fprintf ('Taksiran Batas Atas : %5.3f\n', atas);
fprintf
('=====\n');
fprintf ('Iterasi   Akar_baru   Galat   Interval\n');
fprintf
```

```
(=====
=====
=====\n');
while nilai>=galat
    no = no +1;
    fungsi1=(bawah^3+bawah^2-8*bawah-10);
    fungsi2=(atas^3+atas^2-8*atas-10);
    m=[(bawah*fungsi2-atas*fungsi1)]/(fungsi2-fungsi1);
    fungsi3=feval('fbs',m);
    if fungsi1*fungsi3==0
        disp('m adalah akarnya');
    elseif fungsi1*fungsi3<0
        atas=m;
    else
        bawah=m;
        fungsi1=fungsi3;
    end
    fprintf (' %3d  %8.5f  %8.5f  [%8.5f ;
%8.5f]\n', no, m, nilai, bawah, atas);
    nilai=abs(m0-m);
    m0=m;
end
fprintf
(=====
=====
=====\n');
fprintf('Pada Iterasi ke-%1d, Selisih Interval < %5.3f\n',
no, galat);
fprintf ('Jadi, akar persamaannya adalah %7.5f\n', m);
```

IV. KESIMPULAN

Metode numerik digunakan untuk menyelesaikan persoalan dimana perhitungan secara

analitik tidak dapat digunakan. Metode numerik ini berangkat dari pemikiran bahwa permasalahan dapat diselesaikan dengan menggunakan pendekatan-pendekatan yang dapat dipertanggung-jawabkan secara analitik. Metode numerik ini disajikan dalam bentuk algoritma-algoritma yang dapat dihitung secara cepat dan mudah.

V. SARAN

Agar dilakukan penelitian selanjutnya mengenai penerapan metode numerik untuk kasus-kasus penerapan pengembangan yang lain dan metode-metode yang berhubungan dalam beberapa penentuan hasil dan perhitungan.

UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada perguruan tinggi STMIK Budi Darma dan Universitas Asahan dan beberapa pihak-pihak lain yang telah memberi dukungan terhadap penelitian dan penerbitan jurnal ini.

DAFTAR PUSTAKA

[1] Mahmudah, 2013, *Modifikasi Metode Newton Raphson dalam menyelesaikan Optimasi Non-Linear Multivariabel Berkendala*. Skripsi Program Studi Matematika.Fakultas Sains dan Teknologi. Univeritas Islam Negeri.Sunan Kalijaga, Yogyakarta

[2] Santiyasa, 2012, *Algoritma Newton Raphson dengan Fungsi Non-Linier*. Program Studi Teknik

Informatika, Jurusan Ilmu Komputer.Fakultas
Matematika dan Ilmu Pengetahuan Alam
Universitas Udayana

- [3] Sirait, 2013. Penaksir Maksimum *likelihood*
dengan Metode iterasi Newton-Raphson.Dosen
Program Studi Matematika FMIPA Universitas
Riau. Kumpulan Makalah Seminar Semirata 2013
Fakultas MIPA Universitas Lampung