

Pemahaman Metode Numerik Dalam Simulasi Deret Taylor Implementasi Matlab

**Sovantri Putra Paskah Halawa¹ Fhadillah Br Hutagalung² Muhammad Zidane Al-
Kautsar³ Putri Harliana⁴**

Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas
Negeri Medan, Kota Medan, Provinsi Sumatera Utara, Indonesia^{1,2,3,4}

Email: sovanhalawa06@gmail.com¹ AudyFhadillah26@gmail.com² zidankece07@gmail.com³
harliana@unimed.ac.id⁴

Abstrak

Pemahaman metode numerik sangat penting dalam menyelesaikan berbagai masalah matematis yang kompleks. Penelitian ini berfokus pada pemahaman metode numerik dalam penerapan Deret Taylor dalam fungsi matematis. Deret Taylor memungkinkan representasi fungsi dalam bentuk polinomial, yang dapat meningkatkan akurasi solusi dengan menambah jumlah suku. Dalam penelitian ini, kami menggunakan Matlab untuk mengimplementasikan simulasi Deret Taylor dan menganalisis keakuratan hasilnya. Hasil simulasi menunjukkan bahwa penggunaan Deret Taylor secara efektif mendekati nilai fungsi yang kompleks, dengan tingkat kesalahan yang berkurang seiring bertambahnya suku dalam deret. Penelitian ini diharapkan dapat memperdalam pemahaman tentang metode numerik dan memberikan wawasan yang praktis mengenai aplikasi Matlab dalam penerapan simulasi deret Taylor.

Kata Kunci: Metode Numerik, Deret Taylor, Matlab



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).

PENDAHULUAN

Dalam era digital yang semakin berkembang, metode numerik memainkan peran penting dalam menyelesaikan berbagai permasalahan matematika yang kompleks, terutama yang tidak dapat diselesaikan secara analitis. Metode Numerik adalah teknik yang digunakan untuk mendapatkan solusi pendekatan dari masalah matematis, seperti permasalahan diferensial, integral, dan akar persamaan. Penggunaannya meluas di berbagai bidang, termasuk ilmu komputer, teknik, fisika, ekonomi, dan lain sebagainya. Karena kemampuannya untuk memecahkan masalah yang bersifat non-linear atau memerlukan pendekatan yang sangat akurat. Salah satu metode numerik yang sering digunakan adalah Deret Taylor, yang memungkinkan kita untuk mendekati suatu fungsi yang sulit atau bahkan mustahil diselesaikan secara eksak. Deret Taylor merepresentasikan fungsi dalam bentuk polinomial berdasarkan turunan-turunan fungsi disuatu titik tertentu. Semakin banyak suku dalam deret yang ditambahkan, semakin akurat hasil pendekatan terhadap fungsi aslinya. Metode ini banyak diaplikasikan dalam menyelesaikan persamaan diferensial dan juga digunakan dalam berbagai simulasi komputasi untuk menghitung nilai dari fungsi yang tidak memiliki solusi eksak. Namun, meskipun Deret Taylor memberikan solusi yang mendekati, proses perhitungannya secara manual memakan waktu dan cenderung menghasilkan kesalahan jika diterapkan pada masalah yang kompleks. Oleh karena itu, diperlukan bantuan perangkat lunak untuk mempercepat proses perhitungan dan meningkatkan akurasinya. Matlab adalah salah satu perangkat lunak yang sering digunakan dalam implementasi metode numerik karena memiliki kemampuan untuk menangani perhitungan matematis yang kompleks dengan cepat dan akurat. Matlab juga dilengkapi dengan berbagai fungsi bawaan yang mendukung perhitungan Deret Taylor dan metode numerik lainnya, membuatnya menjadi

pilihan yang populer di kalangan peneliti dan praktisi. Penelitian ini bertujuan untuk mengeksplorasi aplikasi Deret Taylor dalam simulasi numerik dengan menggunakan Matlab. Melalui simulasi ini, diharapkan dapat dipahami bagaimana metode Deret Taylor dapat digunakan untuk mendekati solusi dari fungsi-fungsi kompleks, serta sejauh mana akurasi pendekatan tersebut bergantung pada jumlah suku dalam deret. Selain itu, penelitian ini juga akan menganalisis kesalahan yang muncul pada setiap iterasi suku Deret Taylor dan bagaimana kesalahan tersebut dapat diminimalkan. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi dalam memperluas pemahaman mengenai metode numerik, khususnya Deret Taylor, serta memberikan panduan praktis dalam implementasinya menggunakan Matlab. Ini juga akan menjadi acuan bagi pengembangan lebih lanjut dalam analisis numerik dan pemrograman komputasi, baik di bidang akademis maupun industri.

Berdasarkan pembahasan di atas, rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana efektivitas metode Deret Taylor dalam menghitung fungsi-fungsi matematika non-linear seperti e^x , $\sin(x)$, dan $\ln(1+x)$?
2. Seberapa besar pengaruh jumlah suku dalam Deret Taylor terhadap tingkat akurasi perhitungan fungsi tersebut?
3. Bagaimana implementasi metode Deret Taylor dalam pemrograman Matlab dapat membantu meminimalkan error absolut dalam pendekatan fungsi-fungsi matematika tersebut?
4. Apakah Deret Taylor dapat memberikan hasil yang memadai untuk fungsi-fungsi tertentu dalam rentang input tertentu, dan bagaimana tingkat konvergensinya saat digunakan untuk fungsi-fungsi yang berbeda?

METODE PENELITIAN

Penelitian ini menggunakan metode kuantitatif dengan pendekatan simulasi numerik untuk menganalisis penerapan Deret Taylor dalam mendekati fungsi matematis yang kompleks. Simulasi dilakukan dengan menggunakan perangkat lunak Matlab, yang dikenal dengan kemampuan numeriknya yang handal. Proses penelitian dibagi menjadi beberapa tahap utama, yaitu: Identifikasi masalah, perancangan simulasi, implementasi algoritma Deret Taylor, dan analisis hasil simulasi.

Identifikasi Masalah

Penelitian ini bertujuan untuk mempelajari efektivitas metode Deret Taylor dalam mendekati nilai suatu fungsi non-linear yang sulit atau mustahil diselesaikan secara analitis. Beberapa fungsi yang dipilih untuk penelitian ini antara lain fungsi eksponensial, trigonometri, dan logaritma. Fokus utama penelitian dari penelitian ini adalah untuk menghitung nilai pendekatan suatu fungsi menggunakan Deret Taylor, mengevaluasi tingkat akurasi berdasarkan jumlah suku dalam deret, dan menentukan hubungan antara jumlah suku dalam deret dan kesalahan yang dihasilkan (error).

Perancangan Simulasi

Setelah identifikasi masalah dilakukan, langkah selanjutnya adalah merancang simulasi yang akan dijalankan. Tahapan ini mencakup:

- Pemilihan fungsi yang akan disimulasikan, yaitu fungsi e^x , $\sin(x)$, dan $\ln(1+x)$.
- Penentuan titik pengembangan Deret Taylor, umumnya diambil titik $x = 0$ atau $x = a$ yang diinginkan

- Penggunaan Matlab untuk memodelkan perhitungan Deret Taylor hingga beberapa iterasi suku.

Implementasi Algoritma Deret Taylor dalam Matlab

Tahap ini adalah implemenatasi dari perancangan simulasi yang telah dibuat. Proses ini melibatkan pemrograman fungsi-fungsi numerik dalam Matlab untuk menghitung nilai fungsi menggunakan Deret Taylor. Langkah-langkah yang dilakukan antara lain:

- Mendefinisikan fungsi asli yang akan disimulasikan, seperti e^x , $\sin(x)$, dan $\ln(1 + x)$.
- Mengembangkan algoritma Deret Taylor dinyatakan sebagai:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} \dots + \frac{x^n}{n!}$$

- Menghitung nilai pendekatan fungsi menggunakan Deret Taylor dengan berbagai jumlah suku, dan mencatat hasilnya.
- Menghitung error selisih antara nilai pendekatan dengan nilai asli pada setiap iterasi, dengan formula error absolut: $\text{Error} = |\text{Nilai Asli} - \text{Nilai Pendekatan}|$

Analisis Hasil Simulasi

Setelah simulasi dijalankan, tahap ini melibatkan analisis terhadap hasil yang diperoleh. Hasil simulasi akan ditampilkan dalam bentuk tabel yang menunjukkan hubungan antara jumlah suku dalam Deret Taylor dan tingkat akurasi (kesalahan), perbandingan fungsi asli dengan hasil pendekatan Deret Taylor, dan Penurunan nilai error seiring bertambahnya suku dalam deret.

Validasi dan Verifikasi

Untuk memastikan akurasi dan validitas hasil penelitian ini juga melakukan verifikasi dengan membandingkan hasil simulasi Deret Taylor dengan hasil dari fungsi bawaan Matlab untuk fungsi yang sama. Hasil ini kemudian dievaluasi untuk melihat apakah metode yang diterapkan berjalan dengan benar.

HASIL PENELITIAN DAN PEMBAHASAN

Penelitian ini melakukan simulasi numerik untuk mendekati tiga fungsi matematis menggunakan Deret Taylor e^x , $\sin(x)$, dan $\ln(1 + x)$. Hasil simulasi diperoleh dengan menggunakan Matlab, dan hasil dari pendekatan Deret Taylor dibandingkan dengan nilai fungsi yang dihitung menggunakan Matlab.

Implementasi Matlab

Kode Program simulasi Deret Taylor untuk fungsi e^x

```
% Pendekatan Deret Taylor untuk e^x
x = 1; % Nilai x yang akan dihitung
max_n = 5; % Jumlah suku maksimum Deret Taylor
errors = zeros(max_n, 1); % Array untuk menyimpan nilai error
approximations = zeros(max_n, 1); % Array untuk menyimpan nilai pendekatan

% Hitung nilai asli dari e^x menggunakan fungsi Matlab
exact_value = exp(x);

% Hitung nilai pendekatan menggunakan Deret Taylor
for n = 1:max_n
    taylor_sum = 0; % Reset jumlah Deret Taylor
```

```
for k = 0:n-1
taylor_sum = taylor_sum + (x^k) / factorial(k); % Hitung suku Deret Taylor
end
approximations(n) = taylor_sum; % Simpan nilai pendekatan
errors(n) = abs(exact_value - taylor_sum); % Hitung error absolut
end

% Tampilkan hasil
fprintf('Tabel Hasil Simulasi untuk e^x pada x = 1\n');
fprintf('-----\n');
fprintf('| Jumlah Suku (n) | Nilai Pendekatan | Nilai Asli | Error Absolut |\n');
fprintf('-----\n');
for n = 1:max_n
    fprintf('| %16d | %16.6f | %11.6f | %14.6f |\n', n, approximations(n), exact_value, errors(n));
end
fprintf('-----\n');
```

Kode Program simulasi Deret Taylor untuk fungsi sin(x)

```
% Pendekatan Deret Taylor untuk sin(x)
x = pi / 4; % Nilai x yang akan dihitung (45 derajat)
max_n = 5; % Jumlah suku maksimum Deret Taylor
errors = zeros(max_n, 1); % Array untuk menyimpan nilai error
approximations = zeros(max_n, 1); % Array untuk menyimpan nilai pendekatan
```

```
% Hitung nilai asli dari sin(x) menggunakan fungsi Matlab
exact_value = sin(x);
```

```
% Hitung nilai pendekatan menggunakan Deret Taylor
for n = 1:max_n
    taylor_sum = 0; % Reset jumlah Deret Taylor
    for k = 0:n-1
        % Menggunakan suku Deret Taylor untuk sin(x)
        taylor_sum = taylor_sum + ((-1)^k * (x^(2*k + 1))) / factorial(2*k + 1);
    end
    approximations(n) = taylor_sum; % Simpan nilai pendekatan
    errors(n) = abs(exact_value - taylor_sum); % Hitung error absolut
end
```

```
% Tampilkan hasil
fprintf('Tabel Hasil Simulasi untuk sin(x) pada x = pi/4\n');
fprintf('-----\n');
fprintf('| Jumlah Suku (n) | Nilai Pendekatan | Nilai Asli | Error Absolut |\n');
fprintf('-----\n');
for n = 1:max_n
    fprintf('| %16d | %16.6f | %11.6f | %14.6f |\n', n, approximations(n), exact_value, errors(n));
end
fprintf('-----\n');
```

Kode Program simulasi Deret Taylor untuk fungsi $\ln(1+x)$

```
% Pendekatan Deret Taylor untuk ln(1+x)
x = 0.5; % Nilai x yang akan dihitung
max_n = 5; % Jumlah suku maksimum Deret Taylor
errors = zeros(max_n, 1); % Array untuk menyimpan nilai error
approximations = zeros(max_n, 1); % Array untuk menyimpan nilai pendekatan

% Hitung nilai asli dari ln(1+x) menggunakan fungsi MATLAB
exact_value = log(1 + x);

% Hitung nilai pendekatan menggunakan Deret Taylor
for n = 1:max_n
    taylor_sum = 0; % Reset jumlah Deret Taylor
    for k = 1:n
        % Menggunakan suku Deret Taylor untuk ln(1+x)
        taylor_sum = taylor_sum + ((-1)^(k-1) * (x^k)) / k;
    end
    approximations(n) = taylor_sum; % Simpan nilai pendekatan
    errors(n) = abs(exact_value - taylor_sum); % Hitung error absolut
end

% Tampilkan hasil
fprintf('Tabel Hasil Simulasi untuk ln(1+x) pada x = 0.5\n');
fprintf('-----\n');
fprintf('| Jumlah Suku (n) | Nilai Pendekatan | Nilai Asli | Error Absolut |\n');
fprintf('-----\n');
for n = 1:max_n
    fprintf('| %16d | %16.6f | %11.6f | %14.6f |\n', n, approximations(n), exact_value, errors(n));
end
fprintf('-----\n');
```

Hasil Kode Program Implementasi Matlab

Pada bagian ini, hasil dari implementasi kode program Matlab ditampilkan untuk menghitung nilai pendekatan fungsi e^x , $\sin(x)$, dan $\ln(1+x)$ menggunakan metode Deret Taylor. Hasilnya menunjukkan perhitungan nilai fungsi berdasarkan jumlah suku yang digunakan dalam Deret serta nilai error absolut dibandingkan dengan nilai asli. Semakin banyak suku yang digunakan, semakin akurat hasil pendekatannya.

```
Tabel Hasil Simulasi untuk e^x pada x = 1
-----

| Jumlah Suku (n) | Nilai Pendekatan | Nilai Asli | Error Absolut |
-----
| 1 | 1.000000 | 2.718282 | 1.718282 |
| 2 | 2.000000 | 2.718282 | 0.718282 |
| 3 | 2.500000 | 2.718282 | 0.218282 |
| 4 | 2.666667 | 2.718282 | 0.051615 |
| 5 | 2.708333 | 2.718282 | 0.009948 |
-----
```

Gambar 1. Output Program Matlab Fungsi e^x

Tabel Hasil Simulasi untuk $\sin(x)$ pada $x = \pi/4$

Jumlah Suku (n)	Nilai Pendekatan	Nilai Asli	Error Absolut
1	0.785398	0.707107	0.078291
2	0.704653	0.707107	0.002454
3	0.707143	0.707107	0.000036
4	0.707106	0.707107	0.000000
5	0.707107	0.707107	0.000000

Gambar 2. Output Program Matlab Fungsi $\sin(x)$

Tabel Hasil Simulasi untuk $\ln(1+x)$ pada $x = 0.5$

Jumlah Suku (n)	Nilai Pendekatan	Nilai Asli	Error Absolut
1	0.500000	0.405465	0.094535
2	0.375000	0.405465	0.030465
3	0.416667	0.405465	0.011202
4	0.401042	0.405465	0.004423
5	0.407292	0.405465	0.001827

Gambar 1.3 Output Program Matlab Fungsi $\ln(1 + x)$

Daftar Tabel

Tabel 1. Hasil Simulasi Untuk Fungsi e^x

Jumlah Suku (n)	Nilai Pendekatan	Nilai Asli	Error Absolut
1	1.000000	2.718282	1.718282
2	2.000000	2.718282	0.718282
3	2.500000	2.718282	0.218282
4	2.666667	2.718282	0.051615
5	2.708333	2.718282	0.009948

Tabel 2. Hasil Simulasi Untuk Fungsi $\sin(x)$

Jumlah Suku (n)	Nilai Pendekatan	Nilai Asli	Error Absolut
1	0.785398	0.707107	0.078291
2	0.704653	0.707107	0.002454
3	0.707143	0.707107	0.000036
4	0.707106	0.707107	0.000000
5	0.707107	0.707107	0.000000

Tabel 3. Hasil Simulasi Untuk Fungsi $\ln(1 + x)$

Jumlah Suku (n)	Nilai Pendekatan	Nilai Asli	Error Absolut
1	0.500000	0.405465	0.094535
2	0.375000	0.405465	0.030465
3	0.416667	0.405465	0.011202
4	0.401042	0.405465	0.004423
5	0.407292	0.405465	0.001827

KESIMPULAN

Pada penelitian ini, telah dilakukan simulasi numerik untuk menghitung fungsi eksponensial e^x , $\sin(x)$, dan $\ln(1+x)$ menggunakan pendekatan Deret Taylor. Hasil simulasi menunjukkan bahwa metode Deret Taylor efektif dalam mendekati nilai asli dari fungsi-fungsi tersebut dengan akurasi yang semakin meningkat seiring bertambahnya jumlah suku yang digunakan dalam deret. Untuk fungsi e^x , pendekatan Deret Taylor dengan jumlah suku yang semakin besar mampu mengurangi error absolut secara signifikan, hingga mendekati nilai asli secara presisi. Hal serupa terjadi pada fungsi $\sin(x)$, di mana penambahan suku dalam deret menghasilkan nilai pendekatan yang sangat dekat dengan nilai asli setelah beberapa suku saja. Pada simulasi untuk fungsi $\ln(1+x)$, hasil yang diperoleh juga menunjukkan bahwa Deret Taylor dapat memberikan hasil yang akurat dalam menghitung fungsi logaritma. Metode Deret Taylor terbukti menjadi alat yang ampuh dalam pendekatan fungsi-fungsi matematika yang kompleks, terutama pada fungsi non-linear seperti eksponensial, trigonometri, dan logaritma. Keuntungan utama dari metode ini adalah kesederhanaannya dalam implementasi, terutama dalam pemrograman numerik seperti Matlab, dan kemampuannya untuk menghasilkan hasil yang semakin akurat dengan penambahan suku deret. Meskipun metode ini sangat efisien untuk beberapa fungsi dalam rentang tertentu, perlu dicatat bahwa konvergensi Deret Taylor bergantung pada jumlah suku yang digunakan serta nilai x . Secara keseluruhan, simulasi ini memperkuat pemahaman akan pentingnya metode numerik dalam menghitung nilai fungsi yang kompleks, dan menunjukkan bagaimana pendekatan Deret Taylor dapat menjadi salah satu teknik yang dapat diandalkan dalam analisis dan pemrograman numerik.

DAFTAR PUSTAKA

- Angga, V. N. (2022). *Implementasi Numerik Gerak Suatu Objek Menggunakan Program Web dan Matlab* (Doctoral dissertation, Universitas Andalas).
- Enkhar, L. R., Saputra, S. D., Zahra, N., & Ansgeria, M. P. (2023). Implementasi Penerapan Python dalam Menyelesaikan Pertidaksamaan Kalkulus. *ALKHAWARIZMI: Jurnal Matematika, Algoritma dan Sains*, 1(1), 90-94.
- Marlina, E., & Ruhiat, D. (2018). Penerapan sub pokok fungsi pada matematika ekonomi terhadap fungsi permintaan dan fungsi penawaran. *AKURAT/ Jurnal Ilmiah Akuntansi FE UNIBBA*, 9(2), 90-96.
- Pandia, W., & Sitepu, I. (2021). Penentuan Galat Persamaan Diferensial Biasa Orde 1 dengan Metode Numerik. *Jurnal Mutiara Pendidikan Indonesia*, 6(1), 31-37.
- Pane, H. (2020). Perbandingan Galat Persamaan Diferensial Biasa Orde 1 Dengan Metode Deret Taylor Metode Runge-Kutta Metode Heun Metode Euler (Doctoral Dissertation, Universitas Quality).
- Resmawan, R., Rosyidah, B. M., & Handayani, R. P. (2023). Komparasi Skema Numerik Euler, Runge-Kutta dan Adam-Basforth-Moulton untuk Menyelesaikan Solusi Persamaan Osilator Harmonik. *Euler: Jurnal Ilmiah Matematika, Sains dan Teknologi*, 11(2), 282-292.
- Sinurat, M. E. B., Ismanto, A., & Hariyadi, H. (2016). *Analisis Pola Sebaran Tumpahan Minyak Mentah (Crude Oil) Dengan Pendekatan Model Hidrodinamika Dan Spill Analysis Di Perairan Balongan, Indramayu, Jawa Barat* (Doctoral dissertation, Diponegoro University).
- Sugiarti, Y. (2012). Implementasi Program Java Dan Metode Numerik Dalam Mengatasi Kesulitan Pembelajaran Matematika. *Jurnal Teknodik*, 456-469.
- Sunilawati, N. M., Dantes, N., & Candiasa, I. M. (2013). *Pengaruh model pembelajaran kooperatif tipe STAD terhadap hasil belajar matematika ditinjau dari kemampuan numerik siswa kelas IV SD* (Doctoral dissertation, Ganesha University of Education)

- Wahyuni, F. (2024). Implementasi Algoritma Deret Taylor pada Komputer Berdaya Rendah. *Jurnal Dunia Ilmu*, 4(7).
- Wulandari, N., Erfiani, E., Irzaman, I., & Syafitri, U. D. (2022). Rancangan G-Optimal pada Peningkatan Kadar Kemurnian Silikon Dioksida. *Sainmatika: Jurnal Ilmiah Matematika dan Ilmu Pengetahuan Alam*, 19(2), 149-155.