

Orbital Simulation with Runge-Kutta Fourth Order Method

Bryan Herdianto
Department of Electrical Engineering
University of Indonesia
Depok, Indonesia
bryan.herdianto17@gmail.com

Abstract—Pemodelan gerak benda langit, seperti planet mengelilingi bintang, merupakan salah satu hal yang penting dalam astronomi. Untuk melakukan pemodelan, solusi yang realistis adalah menggunakan metode numerik beserta Hukum Gravitasi Universal Newton. Makalah ini berfokus pada penerapan metode Runge-Kutta orde keempat (RK4) untuk melakukan simulasi orbit planet. Persamaan gerak akan dirumuskan dari hukum gravitasi menjadi sistem persamaan diferensial (ODE) orde pertama dan menunjukkan bagaimana RK4 secara iteratif memperbarui posisi dan kecepatan planet pada setiap step size waktu. Simulasi ini akan mendemonstrasikan akurasi metode RK4 dalam mereplikasi orbit elips Keplerian.

Keywords—Runge-Kutta, Newton, Kepler, ODE, Astronomi

I. PENDAHULUAN

Sejak dahulu kala, manusia telah terpesona oleh gerakan benda-benda di langit. Pemahaman tentang orbit planet dan bintang adalah inti dari astronomi. Isaac Newton merevolusi bidang ini dengan Hukum Gravitasi, yang secara matematis menjelaskan gaya yang mengatur interaksi benda-benda di alam semesta. Untuk sistem dua benda yang ideal solusi analitis dapat diturunkan, menghasilkan orbit elips yang dijelaskan oleh Hukum Kepler. Namun, dalam banyak kasus nyata, seperti ketika mempertimbangkan banyak interaksi dengan benda langit lain, solusi analitis menjadi tidak mungkin. Di sinilah metode numerik berperan. Metode ini memungkinkan kita untuk mengaproksimasi solusi persamaan diferensial yang menggambarkan gerak, langkah demi langkah, dari kondisi awal yang diketahui. Di antara berbagai metode numerik, metode Runge-Kutta orde keempat (RK4) adalah pilihan yang populer karena keseimbangan yang baik antara akurasi dan kompleksitas komputasi. Maka, tujuan makalah ini adalah menjelaskan dan mendemonstrasikan bagaimana metode RK4 dapat digunakan untuk membuat simulasi gerak orbit sebuah planet, memberikan pemahaman yang lebih dalam tentang cara komputasi mekanika benda langit.

II. STUDI LITERATUR

A. Hukum Gravitasi Newton

Hukum gravitasi Newton menyatakan bahwa setiap objek dengan massa menarik objek lain dengan gaya gravitasi yang sebanding dengan massa kedua objek dan berbanding terbalik dengan kuadrat jarak di antara mereka. Berikut rumus dari Hukum Gravitasi Newton:

$$F = \frac{Gm_1m_2}{r^2}$$

dengan F adalah gaya antara massa, G adalah konstanta gravitasi, m_1 adalah massa benda pertama, m_2 adalah massa benda kedua, dan r adalah jarak antar pusat dari benda [1].

B. Mekanika Benda Langit

Mekanika benda langit adalah cabang ilmu yang mempelajari tentang gerak dan lintasan benda-benda langit di bawah pengaruh gravitasi. Salah satu konsep utama dalam cabang ini adalah orbit, yaitu lintasan yang dilalui oleh suatu benda langit saat bergerak mengelilingi benda langit lainnya—seperti Bumi yang mengorbit Matahari. Orbit-orbit ini dapat berbentuk lingkaran, elips, parabola, atau hiperbola tergantung pada kondisi awal gerakan dan gaya gravitasi yang bekerja [2].

Pemahaman tentang orbit tidak lepas dari hukum Kepler, yaitu tiga hukum yang dirumuskan oleh Johannes Kepler untuk menjelaskan gerak planet-planet di sekitar Matahari. Hukum-hukum ini menjelaskan bahwa lintasan planet adalah elips, bukan lingkaran, kecepatan planet bervariasi saat mengorbit, dan periode revolusi planet berbanding lurus dengan kuadrat jaraknya dari Matahari.

C. Ordinary Differential Equation

Persamaan *Differential Equation* (ODE) melibatkan turunan dari suatu fungsi terhadap satu variabel bebas. Banyak masalah dalam fisika dapat dirumuskan sebagai ODE karena perubahan suatu besaran fisik—seperti posisi, kecepatan, atau energi—sering bergantung pada waktu atau variabel lain. Sebagai contoh, gerak benda di bawah pengaruh gaya gravitasi dapat dinyatakan sebagai ODE yang menghubungkan percepatan dengan posisi. ODE orde satu (*first-order*) adalah persamaan yang hanya melibatkan turunan pertama dari fungsi terhadap variabel bebas.

D. Metode Runge-Kutta Orde Empat (RK4)

Algoritma Runge-Kutta orde 4 (RK4) adalah metode numerik yang digunakan untuk menyelesaikan ODE dengan pendekatan yang akurat. Metode ini menghitung empat kemiringan atau slope pada setiap langkah. k_1 adalah nilai turunan di awal interval, k_2 dan k_3 adalah kemiringan di titik tengah menggunakan estimasi dari k_1 dan k_2 , sedangkan k_4 adalah kemiringan di akhir interval berdasarkan k_3 . Keempat kemiringan ini kemudian digabungkan dengan bobot tertentu untuk memperkirakan nilai fungsi di langkah berikutnya. Pendekatan ini memberikan keseimbangan antara akurasi dan efisiensi komputasi sehingga banyak digunakan dalam simulasi orbit [3-8].

III. PENJELASAN DATA

Dalam studi ini, akan digunakan empat set data yang merepresentasikan kondisi awal orbit planet-planet di Tata Surya, yaitu Bumi, Mars, Merkurius, dan Jupiter. Setiap planet diinisialisasi dengan posisi (x, y) dan kecepatan awal (v_x, v_y) yang berbeda sesuai karakteristik orbit nyatanya. Setiap planet juga akan disertai periode orbit nya masing-

masing. Sementara itu, konstanta gravitasi universal (G) dan nilai massa Matahari (M_{\odot}) digunakan secara tetap, yaitu:

- $G = 6.6743 \times 10^{-11} \text{ m}^3 \text{ kg}^{-1} \text{ s}^{-2}$
- $M_{\odot} = 1.989 \times 10^{30} \text{ kg}$

TABLE I. DATA-DATA PLANET

Planet	Parameter		
	Periode (detik)	Jarak (m)	Kecepatan (m/s)
Merkurius	7.6×10^6	4.6×10^{10}	5.9×10^4
Bumi	3.2×10^7	1.47×10^{11}	3.03×10^4
Mars	5.9×10^7	2.07×10^{11}	2.65×10^4
Jupiter	3.7×10^8	7.4×10^{11}	1.31×10^4

Kombinasi ini mencerminkan variasi lintasan orbit akibat perbedaan jarak terhadap Matahari dan kecepatan awal gerakannya. Data digunakan sebagai input dalam simulasi numerik berbasis metode Runge-Kutta orde 4 (RK4), yang menghasilkan lintasan orbit tiap planet dalam bentuk koordinat posisi sepanjang waktu simulasi.

IV. PENJELASAN METODE

Pertama-tama, menggunakan Hukum Newton tentang Gravitasi, kita bisa menemukan rumus model untuk percepatan gravitasi:

$$\vec{a} = -\frac{GM_{\odot}}{r^3} \vec{r}$$

dimana r merupakan jarak dari planet ke matahari atau pusat, yaitu $\sqrt{x^2 + y^2}$. Setelah mengetahui rumus percepatan gravitasi, kita juga membuat sistem persamaan ODE:

$$\begin{aligned} \frac{dx}{dt} &= v_x \\ \frac{dy}{dt} &= v_y \\ \frac{dv_x}{dt} &= \frac{d^2x}{dt^2} = -\frac{GM_{\odot}}{r^3} x \\ \frac{dv_y}{dt} &= \frac{d^2y}{dt^2} = -\frac{GM_{\odot}}{r^3} y \end{aligned}$$

Berikutnya, kita akan menggunakan metode numerik Runge-Kutta orde empat (RK4). RK4 adalah metode numerik untuk menyelesaikan ODE dengan akurasi tinggi. Metode ini menggunakan empat pendekatan kemiringan (slope) pada setiap langkah waktu: awal, dua titik tengah, dan akhir. Nilai-nilai ini kemudian digabungkan secara tertimbang untuk memperkirakan keadaan sistem pada langkah berikutnya. RK4 banyak digunakan karena seimbang antara ketelitian dan efisiensi komputasi. Initial value problem (IVP) dari pemodelan orbit ini adalah:

$$S_n = (x, y, v_x, v_y)$$

Berikut perhitungan untuk slope pertama dari RK4:

$$\begin{aligned} k_1^x &= v_x \\ k_1^y &= v_y \end{aligned}$$

$$k_1^{vx} = -\frac{GM}{r^3} x$$

$$k_1^{vy} = -\frac{GM}{r^3} y$$

Lalu, menggunakan slope pertama, akan dicari nilai dari S_n yang terbaru:

$$k_2 = f\left(S_n + \frac{h}{2} k_1\right)$$

$$x' = x + \frac{h}{2} k_1^x = x + \frac{h}{2} v_x$$

$$y' = y + \frac{h}{2} k_1^y = y + \frac{h}{2} v_y$$

$$v_x' = v_x + \frac{h}{2} k_1^{vx} = v_x + \frac{h}{2} a_x$$

$$v_y' = v_y + \frac{h}{2} k_1^{vy} = v_y + \frac{h}{2} a_y$$

$$k_2^x = v_x'$$

$$k_2^y = v_y'$$

$$k_2^{vx} = -\frac{GM}{r'^3} x'$$

$$k_2^{vy} = -\frac{GM}{r'^3} y'$$

Hal di atas dilakukan secara berulang untuk:

$$k_3 = f\left(S_n + \frac{h}{2} k_2\right)$$

$$k_4 = f(S_n + h k_3)$$

Setelah mendapatkan itu semua, maka bisa digabungkan masing-masing slope sesuai dengan bobot tertentu untuk mendapatkan nilai $x_{n+1}, y_{n+1}, v_{x_{n+1}}, v_{y_{n+1}}$.

$$x_{n+1} = x_n + \frac{h}{6} (k_1^x + 2k_2^x + 2k_3^x + k_4^x)$$

$$y_{n+1} = y_n + \frac{h}{6} (k_1^y + 2k_2^y + 2k_3^y + k_4^y)$$

$$v_{x_{n+1}} = v_{x_n} + \frac{h}{6} (k_1^{vx} + 2k_2^{vx} + 2k_3^{vx} + k_4^{vx})$$

$$v_{y_{n+1}} = v_{y_n} + \frac{h}{6} (k_1^{vy} + 2k_2^{vy} + 2k_3^{vy} + k_4^{vy})$$

Perhitungan tersebut di atas dilakukan berulang-ulang sesuai dengan jumlah iterasi. Jumlah iterasi ditentukan oleh total waktu simulasi yang telah ditentukan oleh user, lalu dibagi dengan step size dari waktu yang juga ditentukan user. Semakin besar total waktu simulasi, maka iterasi yang dilakukan semakin banyak. Sebaliknya, jika semakin besar step size waktu, maka total iterasi yang dilakukan semakin kecil. Iterasi yang dilakukan mengikuti rumusan berikut:

$$n_{iterasi} = \frac{\text{total waktu simulasi}}{\text{step size waktu}}$$

Untuk menghitung rumus-rumus yang sudah tertera, akan digunakan cara pemrograman. Salah satu keuntungan metode numerik adalah kemampuannya untuk diimplementasikan dalam bentuk program. Dalam hal ini, program akan dibuat menggunakan bahasa C/C++ yang terkenal karena

kecepatannya. Pertama-tama, program akan meminta input berupa posisi awal (x,y) dan kecepatan awal (vx,vy) dari objek. Setelah itu, pengguna diminta untuk memasukkan total waktu simulasi dan step size waktu.

V. DISKUSI & ANALISIS HASIL

Hasil dari pemrograman disimpan dalam sebuah file CSV yang berisi data waktu dan koordinat posisi objek dari waktu ke waktu, serta visualisasi lintasan menggunakan plot gambar yang dihasilkan dengan bantuan library GNUPlot. Data simulasi diujikan pada empat planet, yaitu Bumi, Merkurius, Mars, dan Jupiter. Untuk keperluan penyajian, hanya 5 baris pertama dari file CSV yang ditampilkan, mengingat file tersebut memiliki baris data yang banyak.

Menggunakan data asli mengenai planet Bumi, maka berikut adalah koordinat posisi nya dari waktu ke waktu:

TABLE II. KOORDINAT BUMI

Waktu (detik)	Koordinat	
	X (m)	Y (m)
0	1.47e11	0
86400	1.4698e11	2.618e9
172800	1.4691e11	5.231e9
259200	1.4679e11	7.85e9
345600	1.466e11	1.046e10

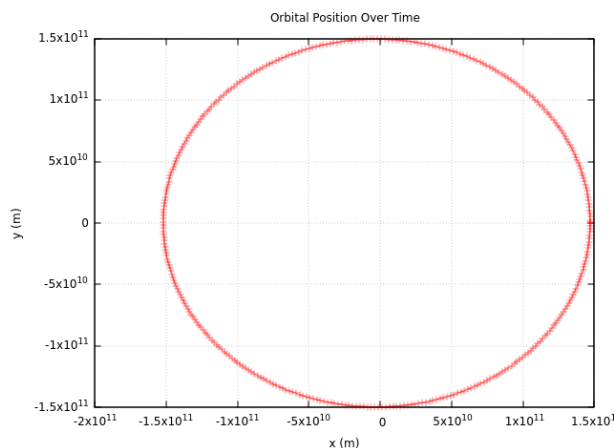


Fig. 1. Orbit Bumi dalam Satu Periode

Dari data gambar, dapat dilihat bahwa Bumi memiliki orbit yang berbentuk elips. Selain itu, Bumi memiliki jarak semimayor dan semiminor orbit yang mirip (mendekati nilai 1.5×10^{11} meter). Hal ini berarti eksentrisitas cukup kecil.

Berikutnya, menggunakan data asli mengenai planet Mars, maka berikut adalah koordinat posisi nya dari waktu ke waktu:

TABLE III. KOORDINAT MARS

Waktu (detik)	Koordinat	
	X (m)	Y (m)
0	2.07e11	0
86400	2.0698e11	2.289e9
172800	2.0695e11	4.579e9
259200	2.0689e11	6.868e9

Waktu (detik)	Koordinat	
	X (m)	Y (m)
345600	2.0681e11	9.156e9

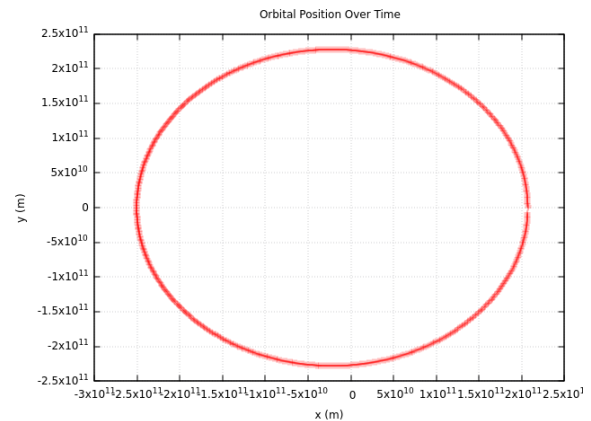


Fig. 2. Orbit Mars dalam Satu Periode

Dari data, Mars memiliki orbit yang berbentuk elips. Selain itu, Mars memiliki jarak semimayor dan semiminor orbit yang beda, tidak seperti di Bumi. Hal ini berarti eksentrisitas Mars lebih besar dari Bumi sehingga bisa dibilang lebih “elips”. Dari gambar, Mars memiliki jarak semimayor sebesar 2×10^{11} meter dan semiminor sebesar 2.25×10^{11} meter.

Berikutnya, menggunakan data asli mengenai planet Merkurius, maka berikut adalah koordinat posisi nya dari waktu ke waktu:

TABLE IV. KOORDINAT MERKURIUS

Waktu (detik)	Koordinat	
	X (m)	Y (m)
0	4.6e10	0
86400	4.5766e10	5.088e9
172800	4.506e10	1.012e10
259200	4.391e10	1.506e9
345600	4.233e10	1.985e9

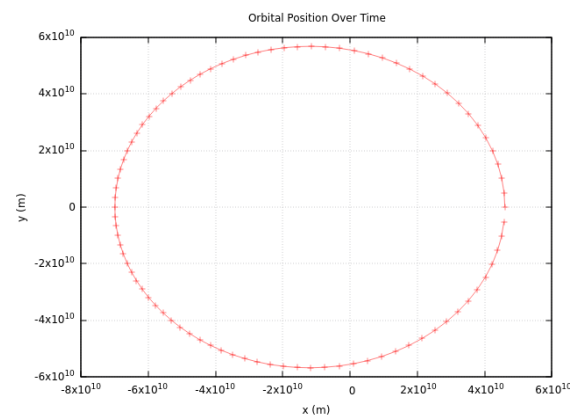


Fig. 3. Orbit Merkurius dalam Satu Periode

Dari data table dan gambar, dapat dilihat bahwa Merkurius memiliki orbit yang berbentuk elips. Selain itu, Merkurius memiliki jarak semimayor dan semiminor orbit yang tidak

sama, tidak seperti di Bumi. Hal ini berarti eksentrisitas Merkurius lebih besar dari Bumi sehingga bisa dibilang lebih “elips”. Menurut data yang nyata, Merkurius pun memiliki eksentrisitas terbesar dibanding planet-planet lainnya.

Berikutnya, menggunakan data asli mengenai planet Jupiter, maka berikut adalah koordinat posisi nya dari waktu ke waktu:

TABLE V. KOORDINAT JUPITER

Waktu (detik)	Koordinat	
	X (m)	Y (m)
0	7.4e11	0
86400	7.39998e11	1.131e9
172800	7.39991e11	2.26e9
259200	7.3998e11	3.39e9
345600	7.3997e11	4.53e9

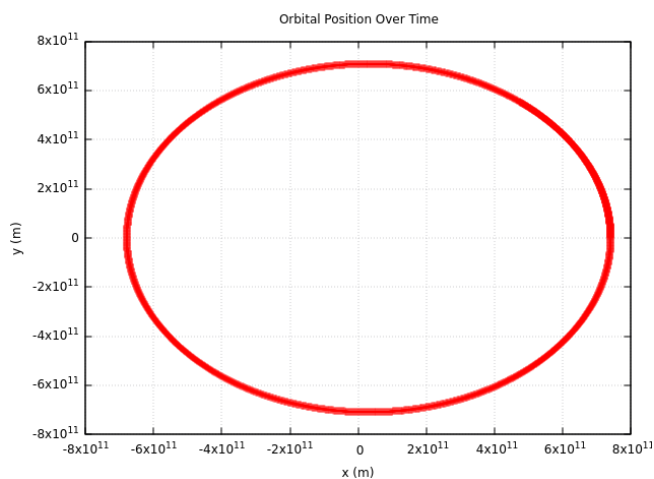


Fig. 4. Orbit Jupiter dalam Satu Periode

Dari data table dan gambar, dapat dilihat bahwa Jupiter memiliki orbit yang berbentuk elips. Selain itu, dapat dilihat Jupiter memiliki lintasan orbit yang sangat tebal dibanding planet-planet lainnya. Hal ini karena data koordinat diambil setiap satu hari dan periode Jupiter mencapai 11 tahun sehingga banyak sekali data titik yang ditempatkan dalam plot.

Dari hasil-hasil di atas, dapat ditarik tiga hal penting. Pertama, metode Runge-Kutta orde empat memiliki akurasi yang cukup tinggi dan waktu komputasi yang cepat sehingga

mampu menghasilkan data CSV dan plot yang baik. Kedua, orbit Bumi memiliki tingkat elips yang paling rendah dibanding planet-planet lainnya. Ketiga, ternyata berangkat dari Hukum Gravitasi Newton akan menyebabkan orbit elips dari planet-planet. Hal ini berarti orbit elips merupakan bentuk orbit yang stabil dan yang mengikuti hukum gravitasi. Hal ini juga berarti bahwa Hukum Kepler 1 berhasil dipenuhi. Hukum Kepler 1 menyatakan “Semua planet bergerak pada lintasan elips mengitari Matahari, dengan Matahari terletak pada salah satu fokus elips tersebut.”

VI. KESIMPULAN

Simulasi gerak orbit planet menggunakan metode Runge-Kutta orde keempat (RK4) menunjukkan hasil yang akurat dan efisien secara komputasi. Data yang dihasilkan dalam format CSV dan visualisasi orbit memperlihatkan bahwa bentuk lintasan elips konsisten dengan hukum gravitasi Newton. Selain itu, ditemukan orbit Bumi memiliki keellipsan paling rendah dibanding planet lain yang dianalisis. Hasil ini mendukung kebenaran Hukum Kepler pertama, bahwa planet bergerak dalam lintasan elips dengan Matahari sebagai salah satu fokusnya. Dengan demikian, pendekatan numerik seperti RK4 terbukti efektif dalam memahami dan merepresentasikan dinamika benda langit.

REFERENCES

- [1] K. L. Lerner, “Newton’s law of universal gravitation,” *ResearchGate*, Jan. 2001, doi: 10.13140/RG.2.2.17902.08009.
- [2] P. K. Seidelmann, “Orbital motion of the planets, theoretical and observational,” *Celestial Mechanics*, vol. 26, no. 2, pp. 149–160, Feb. 1982, doi: 10.1007/bf01230878.
- [3] J. R. Dormand and P. J. Prince, “New Runge-Kutta algorithms for numerical simulation in dynamical astronomy,” *Celestial Mechanics*, vol. 18, no. 3, pp. 223–232, Oct. 1978, doi: 10.1007/bf01230162.
- [4] Md. A. B. Masud and S. A. Doly, “A Study of Stellar Model with Kramer’s Opacity by Using Runge Kutta Method with Programming C,” *International Journal of Astronomy and Astrophysics*, vol. 07, no. 03, pp. 185–201, Jan. 2017, doi: 10.4236/ijaa.2017.73015.
- [5] P. L. Sharma and A. Kumar, “Demonstration study on Runge-Kutta Fourth Order Method by using MATLAB programming,” *ResearchGate*, Aug. 2021, doi: 10.9790/5728-1704030109.
- [6] H. Ramos and J. Vigo-Aguiar, “A fourth-order Runge–Kutta method based on BDF-type Chebyshev approximations,” *Journal of Computational and Applied Mathematics*, vol. 204, no. 1, pp. 124–136, Jun. 2006, doi: 10.1016/j.cam.2006.04.033.
- [7] S. A. Medjahed, “Introduction of the Runge-Kutta method in GPS orbit computation,” *International Journal of Engineering and Geosciences*, vol. 9, no. 2, pp. 256–263, Jul. 2024, doi: 10.26833/ijeg.1403435.
- [8] “View of numerical solution analysis of planetary motion models using the Runge-Kutta Method.” <https://cahaya-ic.com/index.php/IJoME/article/view/1359/926>