

Simulasi Orbit Planet Eksentrisitas Tinggi dengan Metode *Leapfrog*

Hasanuddin✉

Program Studi Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Tanjungpura, Jl. Prof. H. Hadari Nawawi, Pontianak, 78124

Info Artikel

Sejarah Artikel:

Diterima:
8 Desember 2022

Disetujui:
21 Juli 2022

Dipublikasikan:
28 Juli 2022

Keywords:
Leapfrog, Planet simulation, Python 3

ABSTRAK

Penelitian ini mengkaji orbit, galat energi, dan besaran aksi dalam simulasi planet mengorbit bintang dengan menggunakan metode *leapfrog* dan langkah waktu konstan. Sebuah modul program dibuat dalam bahasa pemrograman Python 3. Nilai posisi dan kecepatan awal planet diatur sedemikian rupa sehingga eksentrisitas orbit bernilai 0,75. Hasil simulasi menunjukkan galat energi yang memiliki nilai batas tertentu yang cenderung konstan. Secara visual, nilai besaran aksi yang diwakili oleh luas daerah yang dibatasi oleh kurva ruang fasa tidak berubah. Hal ini disebabkan oleh sifat *time-symmetric* dan *symplectic* dari metode *leapfrog*. Akan tetapi, pergeseran fasa orbit tetap terjadi. Sebagai perbandingan, metode Runge-Kutta orde 2 yang tidak *time-symmetric* dan *non-symplectic* menghasilkan galat energi dan besaran aksi yang meningkat secara linier. Selain itu, pergeseran fase metode Runge-Kutta orde 2 terlihat lebih besar dari pergeseran fase metode *leapfrog*. Hal ini menunjukkan metode *leapfrog* sangat cocok dipakai untuk simulasi planet dalam jangka waktu yang sangat lama.

ABSTRACT

This research studied orbit, energy error, and actions in planet simulation orbiting a star with the leapfrog method and constant time-step. A module has been made in Python 3. The initial position and velocity of the planet were set such that the orbit has an eccentricity of 0.75. The result of the simulation shows the energy error has a constant limit. Visually, actions that are represented by area inside phase space curves are constants. This result is due to the time-symmetric and symplectic properties of the leapfrog method. However, the phase precession of orbit still emerged. As a comparison, the second-order Runge-Kutta method which is neither time-symmetric nor symplectic yields linearly increasing energy error and actions. Besides that, phase precession of the second order Runge-Kutta Method is higher than those leapfrog method. This result shows that the leapfrog method is suitable for long time planet simulation.

✉ Alamat korespondensi:
Program Studi Fisika, Fakultas MIPA, Universitas Tanjungpura
E-mail: hasanuddin@physics.untan.ac.id

PENDAHULUAN

Metode *leapfrog* sering digunakan sebagai *integrator* dalam menyelesaikan persamaan Hamilton sistem dinamik (Cuendet & van Gunsteren, 2007; Dehnen & Read, 2011; Hut dkk., 1995), terutama untuk sistem yang disimulasikan dalam waktu yang sangat lama (Leimkuhler & Reich, 2005). Metode ini lebih disukai karena dua hal. Pertama, diperlukan hanya dua kali perhitungan gaya untuk satu langkah. Kedua, galat energi memiliki batas atas. Jadi, tidak ada rambatan galat energi. Hal ini karena sifat *time symmetric* yang dimilikinya. Sifat ini memiliki arti sebagai berikut. Jika integrator ini diaplikasikan pada suatu keadaan awal sistem (misalnya posisi dan kecepatan awal) dua kali berturut-turut, pertama dengan langkah maju/positif dan diikuti langkah mundur/negatif, maka keadaan sistem kembali ke keadaan awal (Hasanuddin, 2020).

Sifat lain yang dimiliki oleh *integrator leapfrog* adalah sifat *symplectic*. Perubahan posisi dan momentum sistem dinamik dalam satu langkah dengan menggunakan *integrator* ini merupakan sebuah transformasi kanonik. Akibatnya, beberapa konstanta seperti integral Jacobi dan volume ruang fasa dijaga tetap.

Metode ini juga dipakai untuk simulasi orbit planet mengelilingi matahari dengan tujuan untuk analisis komputasi dan perbandingan galat energi yang dihasilkan dengan berbagai metode numerik lainnya (Suraina dkk., 2015). Simulasi orbit planet dalam tata surya biasa digunakan karena solusi eksak orbit planet dapat diperoleh dari hukum gravitasi Newton dan hukum Newton II. Selain itu, bentuk orbit elips planet secara empiris dinyatakan oleh hukum Kepler I (Carroll & Ostlie, 2017).

Akan tetapi, orbit-orbit planet dalam tata surya memiliki *eksentrisitas* yang sangat kecil, yaitu berkisar 0,0068 sampai 0,2056 (Karttunen dkk., 2016), sehingga bisa diasumsikan orbit berbentuk hampir melingkar. Akibatnya, perbandingan rambatan galat energi energi total dari *integrator leapfrog* dan berbagai metode numerik lainnya tidak mencakup kasus orbit dengan *eksentrisitas* yang tinggi. Kajian orbit yang lebih tinggi sangat menantang dalam komputasi numerik sebab kecepatan planet sangat jauh berbeda ketika planet berada di titik terdekat dan terjauh dari bintang. Sebagai contoh, penelitian Suraina dkk. (2015) hanya mengkaji galat energi akhir simulasi orbit eksentrisitas rendah tanpa membahas rambatan galat energi dan belum menunjukkan sifat *time symmetric* dan *symplectic* yang dimiliki oleh metode *leapfrog*.

Oleh karena itu, penelitian ini mengkaji orbit, galat energi, dan besaran aksi dalam simulasi planet yang mengorbit bintang dengan *eksentrisitas* tinggi. Simulasi ini menggunakan modul program yang dibuat dengan bahasa pemrograman Python 3 yang bersifat *open source* dan sederhana. Penelitian ini menggunakan metode *leapfrog* dan langkah waktu yang tetap. Selain itu, metode Runge-Kutta orde 2 yang tidak bersifat *time symmetric* maupun *symplectic* juga digunakan sebagai pembanding.

METODE

Tipe integrator *leapfrog* yang digunakan adalah tipe *Kick-Drift-Kick* (KDK) yang dinyatakan oleh Persamaan (1), (2), dan (3) (Dehnen & Read, 2011):

$$\mathbf{v}_{1/2} = \mathbf{v}_0 + \mathbf{a}_0 \Delta t/2, \quad (1)$$

$$\mathbf{r}_1 = \mathbf{r}_0 + \mathbf{v}_{1/2} \Delta t, \quad (2)$$

$$\mathbf{v}_1 = \mathbf{v}_{1/2} + \mathbf{a}_1 \Delta t/2, \quad (3)$$

dengan \mathbf{v}_0 , $\mathbf{v}_{1/2}$, dan \mathbf{v}_1 masing-masing kecepatan planet awal, setengah langkah ($\Delta t/2$), dan satu langkah (Δt). Variabel \mathbf{r}_0 dan \mathbf{r}_1 masing-masing adalah posisi planet awal dan satu langkah. Variabel \mathbf{a}_0 dan \mathbf{a}_1 masing-masing adalah percepatan awal dan satu langkah. Potensial gravitasi yang berasal dari bintang diasumsikan sebagai potensial massa titik sehingga diperoleh percepatan planet seperti ditunjukkan pada Persamaan (4).

$$\mathbf{a} = -\frac{GM}{r^3}\mathbf{r}, \quad (4)$$

dengan G adalah tetapan gravitasi universal, M adalah massa bintang, dan r adalah jarak antara planet dan bintang.

Untuk simulasi planet ini, diperlukan sebuah program. Oleh karena itu, dibuat sebuah modul bernama **Hsimulator** yang dapat dijalankan dalam *software* Python 3. Modul ini berisi *class* yang terdiri dari *class* orbit, *class* potential dan *class* snapshot. *Class* orbit memiliki atribut posisi, kecepatan, potensial, dan percepatan planet. *Class* orbit juga memiliki metode berupa **set_gravity** untuk memberi nilai potensial dan percepatan planet jika diberikan posisi planet. Selain itu, *class* ini juga memiliki metode **drift** dan **kick** untuk memperbaharui nilai posisi dan kecepatan pada selang waktu berikutnya. *Class* snapshot merupakan rekaman kumpulan orbit-orbit planet pada waktu tertentu. *Class* ini digunakan untuk menampung orbit-orbit planet yang atributnya terdiri dari waktu snapshot dan *object* orbit. *Class* potential digunakan untuk merepresentasikan potensial massa titik. Dalam *class* ini, terdapat metode **pot** dan **acc** yang memberikan nilai potensial dan percepatan pada suatu titik.

Modul ini juga berisi sebuah fungsi utama bernama **run** yang digunakan untuk menjalankan simulasi. Untuk menggunakan modul ini, digunakan perintah *import* di terminal Python seperti yang ditunjukkan oleh potongan kode berikut:

```
>>>import Hsimulator as H
>>>H.run(snap=snapshot)
```

Kode program ini dapat dilihat di website <https://github.com/hasanastro4/Hsimulator>. Orbit planet disimulasikan dalam waktu 200 periode dan besar langkah waktu $3,68 \times 10^{-3}$ periode. Dalam penelitian ini, digunakan satuan jarak dan kecepatan sedemikian rupa sehingga nilai $GM = 1$. Langkah-langkah simulasi planet dapat dilihat pada Gambar 1. Sebagai perbandingan, metode yang bukan *time symmetric* tetapi memiliki orde akurasi yang sama juga digunakan. Untuk tujuan ini, dipakai metode Runge-Kutta orde-2 (selanjutnya dinamakan RK2). Metode ini dapat dituliskan oleh Persamaan (5) sampai dengan Persamaan (10) (Süli & Mayers, 2003):

$$\mathbf{k}_1 = \mathbf{a}(\mathbf{r}_0) \Delta t, \quad (5)$$

$$\mathbf{l}_1 = \mathbf{v}_0 \Delta t, \quad (6)$$

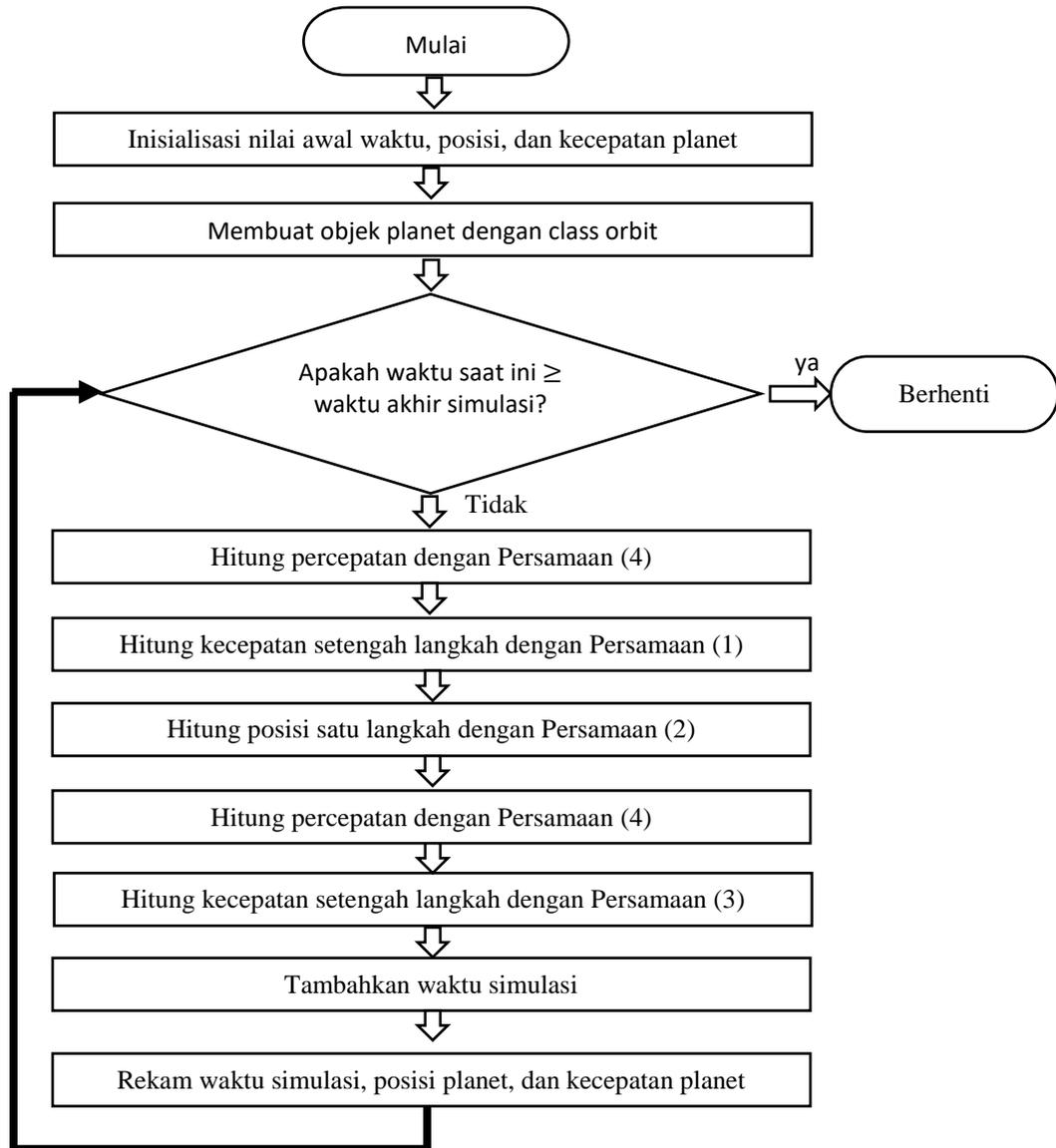
$$\mathbf{k}_2 = \mathbf{a}(\mathbf{r}_0 + \mathbf{l}_1) \Delta t, \quad (7)$$

$$\mathbf{l}_2 = (\mathbf{v}_0 + \mathbf{k}_1) \Delta t, \quad (8)$$

$$\mathbf{r}_1 = \mathbf{r}_0 + (\mathbf{l}_1 + \mathbf{l}_2)/2, \quad (9)$$

$$\mathbf{v}_1 = \mathbf{v}_0 + (\mathbf{k}_1 + \mathbf{k}_2)/2. \quad (10)$$

dengan \mathbf{k} dan \mathbf{l} masing-masing adalah besaran vektor yang menunjukkan perubahan kecepatan dan posisi. Langkah-langkah simulasi planet dengan menggunakan metode RK2 hampir sama seperti diagram alir pada Gambar 1, kecuali persamaan untuk menghitung posisi dan kecepatan diganti dengan Persamaan (9) dan Persamaan (10).

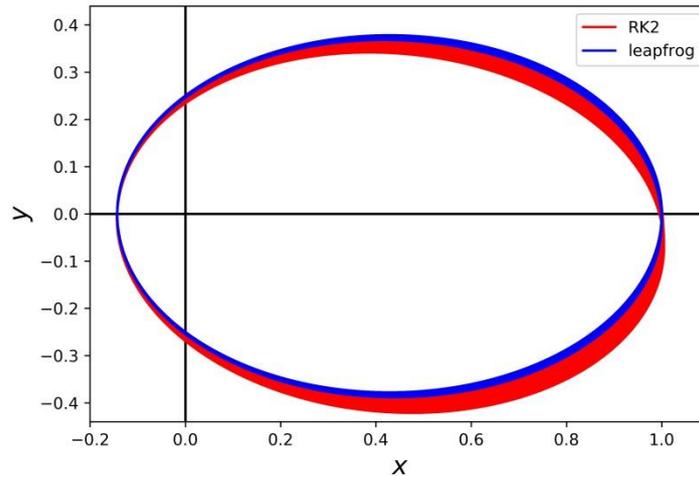


Gambar 1. Diagram Alir Simulasi Planet dengan menggunakan metode leapfrog.

HASIL DAN PEMBAHASAN

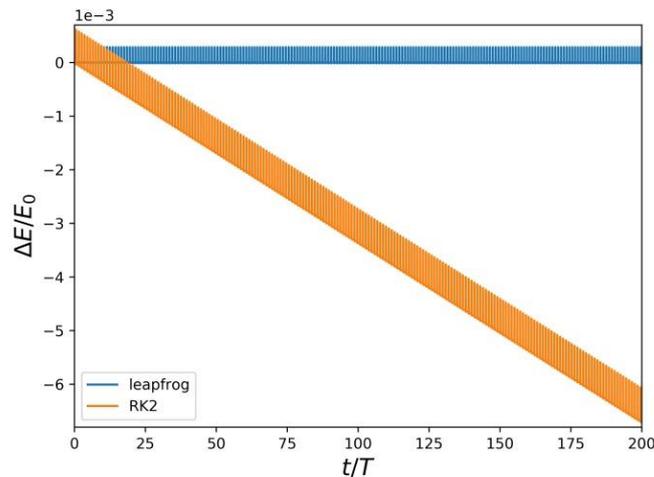
Dalam simulasi ini, digunakan nilai awal posisi $\mathbf{r} = \hat{x}$ dan nilai awal kecepatan $\mathbf{v} = 0,5 \hat{y}$. Di kondisi awal ini, planet berada di titik terjauh dari bintang (*apocenter*) dan planet tidak memiliki kecepatan radial. Orbit yang dihasilkan memiliki *eksentrisitas* 0,75 dan titik terdekat planet dari bintang (*pericenter*) berjarak sekitar 0,14. Bentuk orbit dari hasil simulasi selama 200 periode dapat dilihat pada Gambar 2.

Orbit planet hasil simulasi dengan menggunakan metode *leapfrog* (ditunjukkan oleh kurva berwarna biru dalam Gambar 2 bersifat stabil dan berbentuk ellips tetapi terjadi pergeseran fasa yang kecil setelah 1 periode. Hal ini ditunjukkan oleh akumulasi pergeseran fasa setelah 200 periode seperti yang terlihat pada Gambar 2. Akan tetapi, pergeseran fasa ini jauh lebih kecil jika dibandingkan dengan metode RK2. Selain itu, setengah jarak antara *apocenter* dan *pericenter* atau panjang sumbu semi-mayor (a) orbit dengan metode RK2 mengalami peningkatan. Lain halnya dengan metode *leapfrog*, nilai a orbit cenderung tetap.



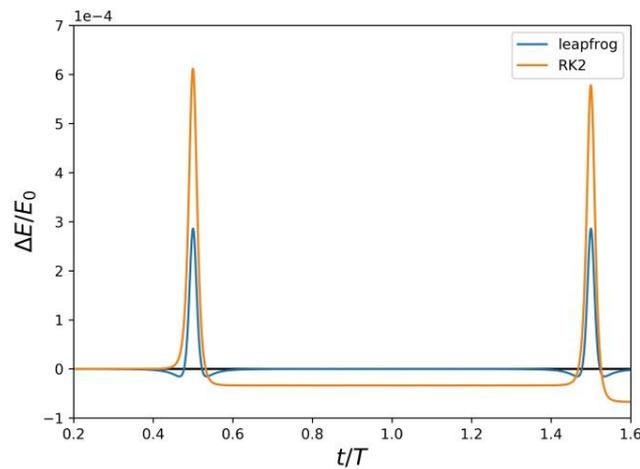
Gambar 2. Trajektori planet selama 200 periode dengan eksentrisitas 0,75

Perubahan a menunjukkan terjadinya perubahan energi karena besar energi berbanding terbalik terhadap a . Rambatan galat energi relatif terhadap energi mula-mula dapat dilihat pada Gambar 3. Simulasi dengan menggunakan metode *leapfrog* menunjukkan tidak adanya rambatan error energi sedangkan metode RK2 menunjukkan rambatan energi error yang linier. Hal ini disebabkan oleh sifat *time-symmetric* yang dimiliki oleh metode *leapfrog*.



Gambar 3. Galat energi relatif metode leapfrog dan RK2 selama 200 periode.

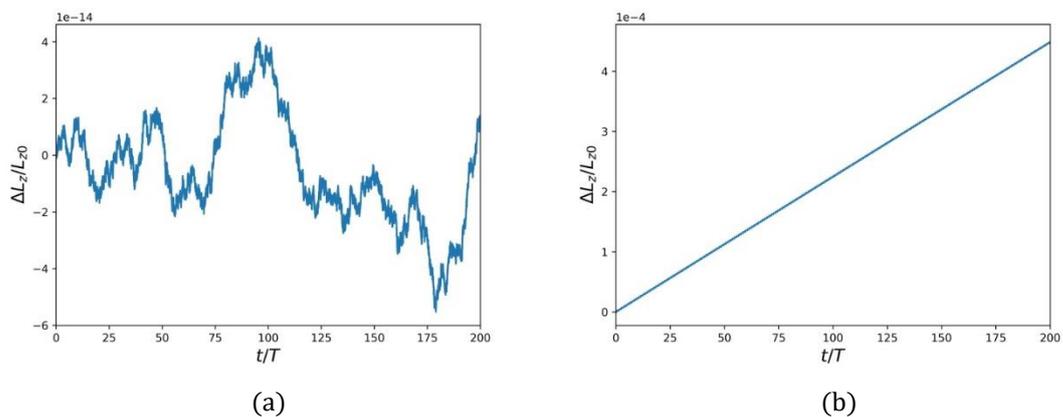
Energi error metode *leapfrog* meningkat sebelum setengah periode orbit dan mencapai puncak pada tepat setengah periode (pada saat planet mencapai *pericenter*). Setelah itu, galat energi menurun dan kembali ke nol ketika mencapai *apocenter*. Ketika planet bergerak dari *pericenter* ke *apocenter*, perhitungan posisi dan kecepatan dalam metode *leapfrog* persis sama dengan perhitungan posisi dan kecepatan pada saat planet bergerak dari *apocenter* ke *pericenter* kecuali dengan langkah waktu yang berbeda tanda. Akibatnya, galat energi saling meniadakan sehingga galat energi kembali nol pada saat planet berada dalam posisi *apocenter* berikutnya seperti yang terlihat pada Gambar 4.



Gambar 4. Galat energi relatif metode leapfrog dan RK2 selama 1,6 periode.

Galat energi metode RK2 mengalami puncaknya di setengah periode orbit dan puncak galat berikutnya terjadi sekitar 1,5 periode. Tetapi, nilai puncak galat berikutnya berbeda dengan sebelumnya karena metode RK2 tidak bersifat *time-symmetric*. Inilah penyebab rambatan galat energi linier seperti yang terlihat pada Gambar 3.

Sifat *symplectic* metode *leapfrog* dapat ditunjukkan dengan besaran aksi dari sistem dinamik yang tetap. Salah satu besaran aksi yang mudah dihitung adalah aksi yang sebanding dengan momentum sudut planet. Momentum sudut planet memiliki galat momentum sudut relatif yang sangat kecil yaitu dalam orde 10^{-14} seperti yang ditunjukkan oleh Gambar 5a. Fluktuasi dalam Gambar 5a disebabkan oleh galat pembulatan karena dalam simulasi ini nilai numerik posisi dan kecepatan direpresentasikan sebagai bilangan titik kambang berketelitian tunggal (*single precision floating point*). Sementara itu, metode RK2 menghasilkan momentum sudut cenderung meningkat secara linier seperti yang ditunjukkan oleh Gambar 5b.



Gambar 5. Grafik galat momentum sudut relatif terhadap waktu simulasi dengan metode (a) leapfrog dan (b) RK2

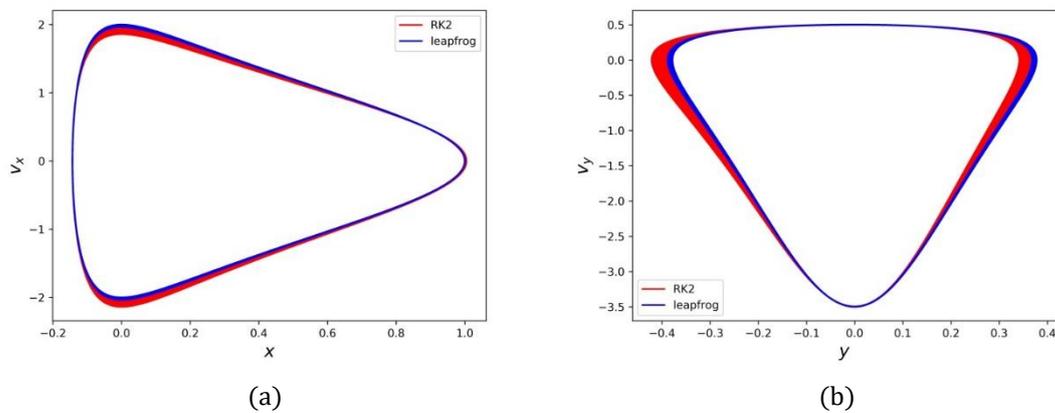
Besaran aksi yang lain, J_x dan J_y masing-masing diberikan oleh Persamaan (11) dan (12) (Goldstein dkk., 2000):

$$J_x = \oint v_x dx \tag{11}$$

dan

$$J_y = \oint v_y dy. \tag{12}$$

Secara visual, perubahan J_x dan J_y dari metode *leapfrog* dan RK2 dapat diamati pada Gambar 6. Nilai kedua aksi ini diwakili oleh luas daerah yang dibatasi oleh kurva ruang-fasa. Terlihat bahwa kedua aksi tidak banyak berubah pada metode *leapfrog* dibandingkan dengan metode RK2. Kedua metode menampilkan pergeseran fasa dalam ruang fasa (hasil serupa ditunjukkan dalam orbit pada Gambar 2). Tetapi, metode RK2 memberikan pergeseran fasa yang cukup signifikan dibandingkan dengan metode *leapfrog*. Untuk mengatasi pergeseran fasa, diperlukan simulasi dengan langkah waktu variabel (Hasanuddin, 2020). Tetapi, simulasi seperti ini di luar cakupan penelitian ini. Jika langkah waktu dapat berubah, maka posisi dan kecepatan planet tidak akan kembali ke posisi dan kecepatan planet semula setelah satu langkah maju dan satu langkah mundur dalam simulasi sehingga sifat *time-symmetric* tidak akan berlaku.



Gambar 6. Kurva ruang fasa (a) v_x vs x dan (b) v_y vs y .

Jika *eksentrisitas* orbit sangat rendah (bentuk orbit mendekati lingkaran), maka galat energi hampir sama di setiap langkah (perbedaan galat energi di sekitar *apocenter* dan *pericenter* sangat kecil). Jadi, perbedaan rambatan galat energi metode *leapfrog* dan RK2 tidak begitu signifikan. Metode RK2 tetap menghasilkan rambatan energi galat meningkat secara linier tetapi dengan kemiringan yang lebih kecil.

Karena sifat *time-symmetric* dan *symplectic* ini, banyak simulasi *N-body* yang menggunakan metode *leapfrog*. Metode ini digunakan mulai dari simulasi dinamika molekul (Cuendet & van Gunsteren, 2007) sampai dinamika galaksi (Binney & Tremaine, 2008; Dehnen & Read, 2011). Keuntungan lain yang didapat dari metode *leapfrog* adalah perhitungan gaya dilakukan cukup dua kali dalam satu langkah waktu sehingga dapat mengurangi biaya komputasi jika diaplikasikan pada sistem *N-body*.

SIMPULAN

Simulasi orbit planet *eksentrisitas* tinggi dengan menggunakan metode *leapfrog* dan langkah konstan menunjukkan pergeseran fase yang lebih kecil dari metode Runge-Kutta Orde 2. Selain itu, galat energi yang dihasilkan tidak mengalami hanyutan (*drift*) karena metode *leapfrog* bersifat *time-symmetric*. Selain itu, secara visual, momentum sudut dan besaran aksi juga tidak meningkat secara linier tetapi cenderung relatif konstan. Jadi, metode *leapfrog* sangat penting dalam simulasi yang memerlukan waktu simulasi yang sangat panjang.

REFERENSI

- Binney, J., & Tremaine, S. (2008). *Galactic Dynamics: Second Edition*. Princeton University Press.
- Carroll, B. W., & Ostlie, D. A. (2017). *An Introduction to Modern Astrophysics (2nd ed.)*. Cambridge University Press.
- Cuendet, M. A., & van Gunsteren, W. F. (2007). On the calculation of velocity-dependent properties in molecular dynamics simulations using the leapfrog integration algorithm. *The Journal of Chemical Physics*, 127(18), 184102.
- Dehnen, W., & Read, J. I. (2011). N-body simulations of gravitational dynamics. *European Physical Journal Plus*, 126, 55.
- Goldstein, H., Poole, C. P. Jr., & Safko, J. (2000). *Classical mechanics (3rd ed.)*. Addison-Wesley.
- Hasanuddin, H. (2020). Analisis Galat Energi dan Galat Fase Metode Forward 4th Order Symplectic Chin-Chen untuk Kasus Sistem Osilator Harmonik Sederhana. *POSITRON*, 10(2), 88–92.
- Hut, P., Makino, J., & McMillan, S. (1995). Building a better leapfrog. *Astrophysical Journal*, 443, L93–L96.
- Karttunen, H., Kröger, P., Oja, H., Poutanen, M., & Donner, K. J. (2016). *Fundamental Astronomy*. Springer Berlin Heidelberg.
- Leimkuhler, B., & Reich, S. (2005). *Simulating Hamiltonian Dynamics (1st ed.)*. Cambridge University Press.
- Süli, E., & Mayers, D. F. (2003). *An Introduction to Numerical Analysis*. Cambridge University Press.
- Suraina, S., Arman, Y., & Lapanporo, B. (2015). Simulasi Orbit Planet dalam Tata Surya dengan Metode Euler, Leapfrog dan Runge-Kutta. *PRISMA FISIKA*, 3(3), 69.