

# Az információtechnika és a bionika fizikai alapjai I.

## Bolygómozgás szimulálása explicit numerikus módszerekkel

Jánoska Dóra Katalin | VHR5VP, Erdélyi Áron János | JBO61Q  
Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar  
2019.03.04.

janoska.dora.katalin@itk.ppke.hu, erdelyi.aron.janos@itk.ppke.hu

**Kivonat**—A Föld Nap körüli keringését szimuláljuk Matlabban. Pár numerikus módszer megvalósítása után látható, hogy a Föld annyira nem látványos példa, hiszen a pályája közelítőleg kör alakú.

Ezáltal kapunk egy differenciálegyenlet-rendszert, amelyikben két ismeretlenünk van:

$$\dot{v} = -G \frac{M}{r^3} \cdot r \quad (5)$$

### I. FIZIKAI HÁTTÉR

A dinamika alaptörvénye azt mondja ki:

$$F = \dot{p} = m \cdot a = m \cdot \ddot{r} \quad (1)$$

Newton gravitációs törvénye szerint, ha a Nap tömege  $M$ , és a Nap tömegközéppontjából az  $r$  vektor mutat az  $m$  tömegű bolygó tömegközéppontjába, akkor a bolygó és a Nap közötti gravitációs vonzóerő:

$$F = -\frac{GmM}{|r|^2} \cdot \frac{r}{|r|} \quad (2)$$

ahol a  $G$  szám az univerzális gravitációs állandó ( $Nm^2 \frac{1}{kg^2}$ -ban mérve).

Ez a gravitációs erő a tömegközéppontokat összekötő egyenes mentén hat. Nap körül keringő bolygót tehát a gravitáció tartja körpályán. Ha nem lenne a kör közepe felé mutató erő, akkor a tárgy nem maradna körpályán, kirepülne.

A test mozgás egyenlete ezek alapján így alakul:

$$\ddot{r} = -G \frac{M}{r^3} \cdot r \quad (3)$$

Mivel a Nap tömege legalább 1000-szerese bármely bolygó tömegénél, és feltehetjük, hogy a Nap mozdulatlan, akkor elhanyagolhatjuk a bolygók tömegét, jelen esetben a Föld ( $m$ ) tömegét

Bevezetjük a

$$v = \dot{r} \quad (4)$$

sebességet.

### II. EULER CRAMER MODELL

Eredetileg lenne 2 másodrendű differenciál egyenletünk, amit át lehet alakítani 4 elsőrendűvé.

$$\dot{x} = v_x \quad (7)$$

$$\dot{v}_x = -G \frac{M}{r^3} \cdot x \quad (8)$$

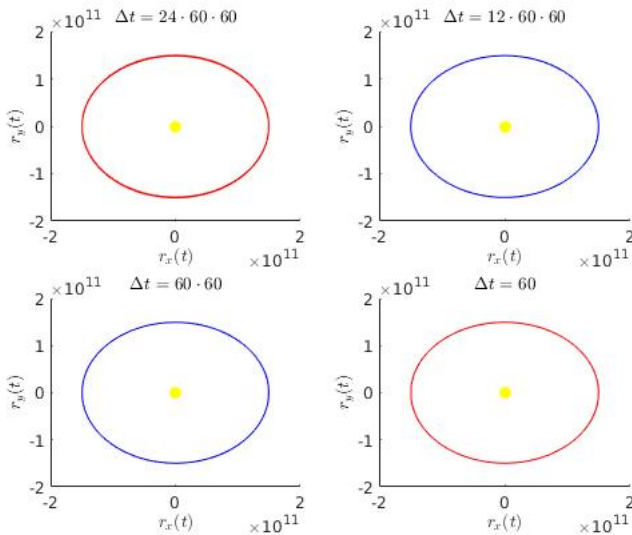
$$\dot{y} = v_y \quad (9)$$

$$\dot{v}_y = -G \frac{M}{r^3} \cdot y \quad (10)$$

Ahol az  $r$ :

$$r = \sqrt{x^2 + y^2} \quad (11)$$

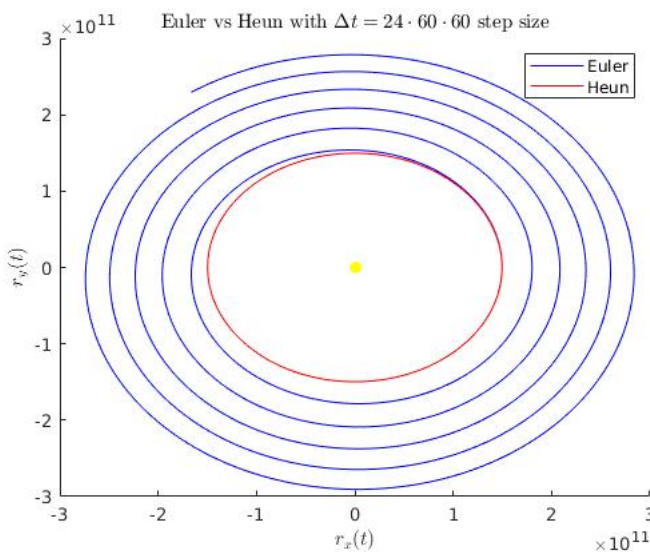
A Matlab-ban ezen összefüggéseket alkalmazva, különböző időintervallumokat nézve adódtak az 1. ábrán láthatóak.



Ezek látszólag hasonlóak, azonban belenagyítva látszódik a különbség. A baj az, hogy a legjobb megoldás csak kis lépésköznél valósul meg, ami pedig nem túl kedvező.

### III. HEUN MÓDSZER

A két módszert (Euler és Heun) láthatjuk a kettő közti különbséget. A Heun-módszer sokkal kisebb lépésköz mellett is kielégítő megoldást nyújt.



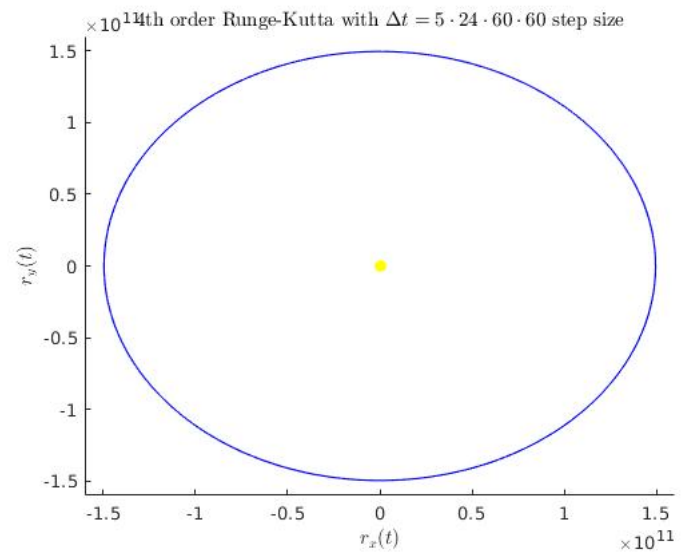
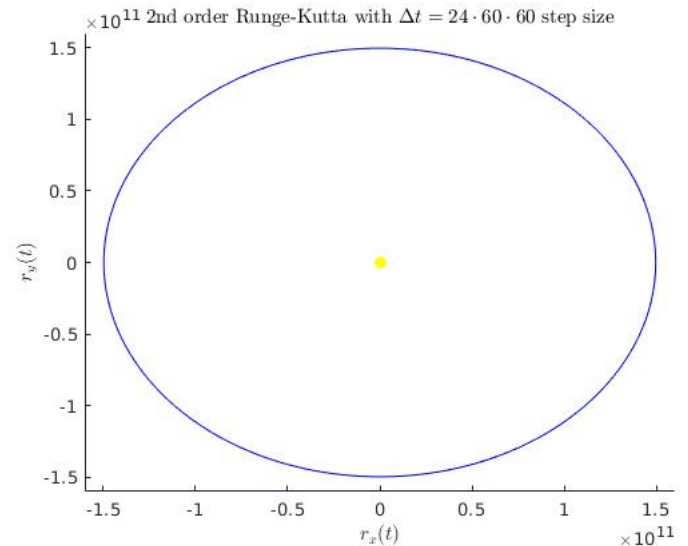
### IV. MÁSOD- ÉS NEGYEDFOKÚ RUNGE-KUTTA

A következő két ábrán láthatjuk a megoldást a másod-, illetve negyed fokú Runge-Kutta módszerrel.

A Runge-Kutta-módszercsalád közönséges negyedrendű tagja annyira elterjedten használatos, hogy egyszerűen csak „a Runge-Kutta-módszer”-ként hivatkoznak rá. E módszer az alábbi kezdetiérték-probléma egy negyedrendű közelítő megoldását adja.

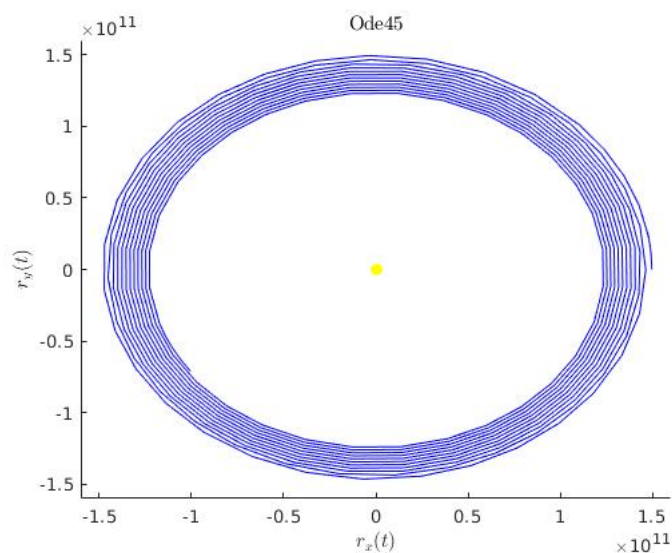
$$y'(t) = f(t, y(t)) \quad y(t_0) = y_0.$$

Azaz tetszőlegesen rögzített pozitív valós, tipikus esetben kicsiny  $\Delta t$  lépésköz esetén az  $n$ -edik lépésben a kezdetiérték-probléma  $y(t)$  megoldásának egy olyan  $y(t_n) \approx y_n$  közelítését adja a  $t_n = t_0 + n \cdot \Delta t$  helyen, amely közelítés hibája negyedrendű. E negyedrendűség azt jelenti, hogy a választott lépésköz zsugorításakor annak negyedik hatványával zsugorodik a hibára adott felső becslés.



### V. ODE45

Végül a Matlab beépített Differenciálegyenlet megoldó algoritmusával próbálkoztunk:



Látható, hogy az ode45 hasonlóan rossz megoldást ad, mint az Euler-módszer. Ez annak köszönhető, hogy az ode45 túl van optimaizálva.

## VI. ÖSSZEGZÉS

A numerikusan megoldó algoritmusok összehasonlításával arra jutottunk, hogy erre a feladatra a Runge-Kutta módszer egyértelműen a legjobb megoldás.