Exemplu gratuit, cu comentarii: licenta-disertatie-romania.com/domenii/matematica/

UNIVERSITATEA [NUMELE UNIVERSITĂȚII]
FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ
Programul de studii: Matematică aplicată

LUCRARE DE LICENȚĂ

Metode numerice pentru ecuații diferențiale ordinare

Metodele Euler, Heun și Runge–Kutta de ordinul 4: convergență, cost și stabilitate

EXEMPLU FICTIV – structura, tabelele și figurile unei lucrări de licență. Datele sunt simulate.

Coordonator științific:

Lect. univ. dr. Diana Popovici (nume fictiv)

Absolvent:

Bogdan Lazăr (nume fictiv)

BUCUREȘTI, 2026

Exemplu fictiv cu date simulate · **Ai nevoie de ajutor cu lucrarea ta?** licenta-disertatie-romania.com/contact

DECLARAȚIE DE AUTENTICITATE

Subsemnatul(a) **Bogdan Lazăr**, absolvent(ă) al(a) programului de studii Matematică aplicată, declar pe propria răspundere că lucrarea de licență cu titlul „Metode numerice pentru ecuații diferențiale ordinare” a fost elaborată de mine, sub îndrumarea coordonatorului științific, nu a mai fost prezentată la o altă facultate sau instituție de învățământ superior și că toate sursele folosite, inclusiv cele de pe internet, sunt citate în lucrare, cu respectarea regulilor privind drepturile de autor.

Data: [data]

Semnătura: [semnătura]

Majoritatea facultăților au un model oficial de declarație de autenticitate. Copiază-l exact din ghidul facultății și semnează-l; textul de aici este doar orientativ.
Sfat
20/05/2026 10:00

CUPRINS

Cuprinsul arată din prima pagină logica lucrării: introducere, capitole teoretice, cercetarea proprie, concluzii. În Word îl generezi automat din stilurile de titlu (Referințe → Cuprins), ca să se actualizeze singur.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Lista figurilor și a tabelelor.....4
Rezumat.....5
Introducere.....6
Capitolul 1. Noțiuni preliminare.....7
1.1. Problema Cauchy.....7
1.2. Metode unipas.....7
1.3. Eroare și ordin de convergență.....8
Capitolul 2. Implementare și experimente numerice.....9
2.1. Problema test.....9
2.2. Implementarea.....9
2.3. Precizia metodelor.....10
2.4. Ordinul de convergență.....10
2.5. Costul de calcul.....12
2.6. Stabilitatea.....12
2.7. Analiza autorului.....13
Concluzii.....14
Bibliografie.....15
Anexa 1. Scriptul pentru tabelul erorilor.....16
Bonus: același paragraf, înainte și după corectare.....17

Lista figurilor și a tabelelor

Listele de figuri și tabele sunt cerute de multe facultăți. În Word le generezi automat dacă pui titlurile cu Referințe → Inserare legendă, apoi Referințe → Inserare tabel de figuri.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Lista figurilor

Figura 1. Clasificarea metodelor numerice pentru problema Cauchy.....7
Figura 2. Etapele experimentului numeric.....9
Figura 3. Soluția exactă și aproximațiile obținute cu pasul h = 0,2.....10
Figura 4. Logaritmul zecimal al erorii globale maxime, în funcție de pas.....11
Figura 5. Ordinul de convergență estimat (perechea 0,025 → 0,0125) și ordinul teoretic.....12
Figura 6. Numărul de evaluări ale lui f pentru e(h) < 10^{-6} (logaritm zecimal).....12
Figura 7. Aproximațiile pentru $y' = -15y$ cu pasul h = 0,15 (primii 10 pași).....13

Lista tabelelor

Tabelul 1. Notațiile folosite în lucrare.....6
Tabelul 2. Metodele studiate.....7
Tabelul 3. Tabloul Butcher al metodei Runge–Kutta clasice (RK4).....8
Tabelul 4. Soluția exactă și aproximațiile numerice pentru h = 0,1.....10
Tabelul 5. Eroarea globală maximă e(h) pe intervalul [0, 2].....11
Tabelul 6. Ordinul de convergență estimat, $p \approx \log_2 [e(h) / e(h/2)]$.....11
Tabelul 7. Costul necesar pentru o eroare globală maximă sub 10^{-6}.....12
Tabelul 8. Testul de stabilitate pentru $y' = -15y$, $y(0) = 1$: factorul de amplificare $R(z)$, $z = -15h$.....13

Rezumat

Lucrarea compară trei metode numerice unipas explicite pentru problema Cauchy: metoda Euler, metoda Heun și metoda Runge–Kutta clasică de ordinul 4 (RK4). După prezentarea rezultatelor teoretice de existență, unicitate și ordin de convergență, metodele sunt implementate în Python și aplicate problemei test y′ = −2ty, y(0) = 1, cu soluția exactă y = e^(−t²). Erorile globale calculate pentru cinci pași confirmă ordinele teoretice: ordinele estimate sunt 1,01, 2,02 și 4,04. Pentru o eroare sub 10^{-6}, RK4 are nevoie de 132 de evaluări ale funcției, față de 650.596 pentru metoda Euler. Un test pe ecuația y′ = −15y arată limita de stabilitate a metodelor explicite: Euler și Heun devin instabile pentru h > 0,133, iar RK4 pentru h > 0,186. Lucrarea recomandă RK4 pentru problemele nerigide și metodele implicite pentru cele rigide.

Rezumatul are de obicei 150–300 de cuvinte și spune pe scurt tema, obiectivul, metoda, rezultatele și concluzia. Multe facultăți îl cer și în limba engleză, cu 3–5 cuvinte-cheie.
Rolul secțiunii
20/05/2026 10:00

Cuvinte-cheie: ecuații diferențiale ordinare, metoda Euler, metode Runge–Kutta, ordin de convergență, stabilitate numerică

Abstract

This thesis compares three explicit one-step numerical methods for the initial value problem: Euler's method, Heun's method and the classical fourth-order Runge–Kutta method (RK4). After presenting the theoretical results on existence, uniqueness and order of convergence, the methods are implemented in Python and applied to the test problem y′ = −2ty, y(0) = 1, whose exact solution is y = e^(−t²). The global errors computed for five step sizes confirm the theoretical orders, with estimated orders of 1.01, 2.02 and 4.04. For an error below 10^{-6}, RK4 needs 132 function evaluations, compared with 650,596 for Euler's method. A test on the equation y′ = −15y shows the stability limit of explicit methods: Euler and Heun become unstable for h > 0.133 and RK4 for h > 0.186. The thesis recommends RK4 for non-stiff problems and implicit methods for stiff ones.

Keywords: ordinary differential equations, Euler method, Runge–Kutta methods, order of convergence, numerical stability

Introducere

La Matematică introducerea prezintă problema, motivația și ce aduce lucrarea: aici, o comparație riguroasă, verificată numeric, a trei metode. Spune de la început ce este cunoscut din literatură și ce ai făcut tu.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Majoritatea ecuațiilor diferențiale care apar în fizică, biologie sau economie nu pot fi rezolvate analitic. Metodele numerice înlocuiesc soluția exactă cu aproximații calculate pe o rețea de puncte, iar întrebarea esențială este cât de repede scade eroarea când micșorăm pasul [1], [2].

Obiectivul lucrării este compararea a trei metode unipas explicite (Euler, Heun și Runge–Kutta de ordinul 4) din trei puncte de vedere: precizie, cost de calcul și stabilitate. Rezultatele teoretice cunoscute sunt verificate numeric pe o problemă test cu soluție exactă.

Contribuția proprie constă în implementarea metodelor în Python, calculul tabelelor de erori pentru cinci pași, estimarea ordinului de convergență, compararea costului necesar pentru o precizie dată și testul de stabilitate.

Definește notațiile o singură dată (Tabelul 1) și păstrează-le în toată lucrarea. Schimbarea notației de la un capitol la altul este una dintre observațiile frecvente ale comisiei.
Ghid: licenta-disertatie-romania.co
Sfat
20/05/2026 10:00

Tabelul 1. Notațiile folosite în lucrare

Notația	Semnificația
$[t_0, T]$	Intervalul de integrare; în exemple $t_0 = 0$, $T = 2$
h, N	Pasul de integrare și numărul de pași, $N = (T - t_0)/h$
$t_n = t_0 + n \cdot h$	Nodurile rețelei, $n = 0, 1, \ldots, N$
y_n	Aproximația numerică a valorii exacte $y(t_n)$
$e(h) = \max \lvert y_n - y(t_n) \rvert$	Eroarea globală maximă pe rețea
p	Ordinul de convergență: $e(h) \approx C \cdot h^p$

Sursa: Notații proprii, după [1], [2].

Capitolul 1. Noțiuni preliminare

Capitolul de preliminarii face lucrarea autonomă: definiții, teoreme clasice (cu trimitere la sursă) și o demonstrație scurtă, scrisă de tine. Teoremele și propozițiile se numerotează, iar demonstrațiile se încheie cu ■.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

1.1. Problema Cauchy

Fie $f : [t_0, T] \times \mathbb{R} \to \mathbb{R}$ o funcție continuă. Problema Cauchy constă în determinarea unei funcții y derivabile care satisface:

$$y'(t) = f(t, y(t)), \quad t \in [t_0, T], \quad y(t_0) = y_0 \qquad (1.1)$$

Teorema 1.1 (de existență și unicitate). *Dacă f este continuă pe $[t_0, T] \times \mathbb{R}$ și lipschitziană în raport cu y, adică există $L > 0$ astfel încât $|f(t, u) - f(t, v)| \le L \cdot |u - v|$ pentru orice t, u, v, atunci problema (1.1) are o soluție unică pe $[t_0, T]$.* Demonstrația se găsește în [2].

1.2. Metode unipas

O metodă unipas calculează y_{n+1} numai din y_n. Figura 1 arată locul metodelor studiate printre metodele numerice pentru EDO, iar Tabelul 2 le prezintă formulele.

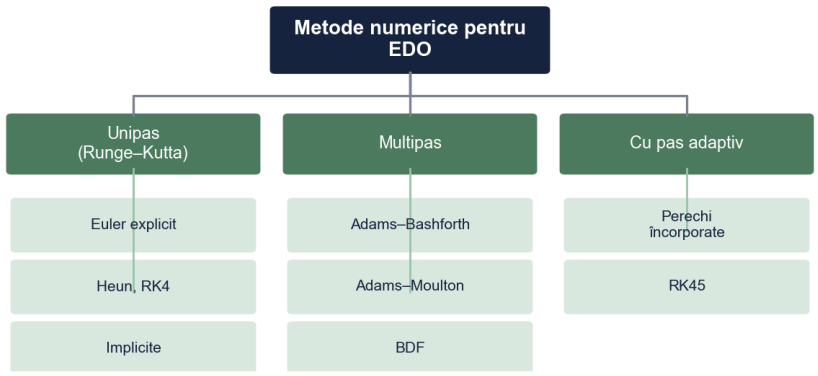


Figura 1. Clasificarea metodelor numerice pentru problema Cauchy
Sursa: Prelucrare proprie.

Tabelul 2. Metodele studiate

Metoda	Formula de avans	Ordinul p	Evaluări ale lui f pe pas
Euler	$y_{n+1} = y_n + h \cdot f(t_n, y_n)$	1	1
Heun	$k_1 = f(t_n, y_n)$, $k_2 = f(t_n + h, y_n + h \cdot k_1)$, $y_{n+1} = y_n + (h/2) \cdot (k_1 + k_2)$	2	2
RK4	$y_{n+1} = y_n + (h/6) \cdot (k_1 + 2k_2 + 2k_3 + k_4)$, cu k_i din Tabelul 3	4	4

Sursa: După [1], [3].

Tabelul 3. Tabloul Butcher al metodei Runge–Kutta clasice (RK4)

c_i	a_{i1}	a_{i2}	a_{i3}	a_{i4}
0				
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
b_j	1/6	1/3	1/3	1/6

Notă. $k_i = f(t_n + c_i \cdot h,\ y_n + h \cdot \Sigma_j\, a_{ij} \cdot k_j)$, $i = 1, \ldots, 4$; ultimul rând conține ponderile b_j din formula de avans.
Sursa: După [1].

1.3. Eroare și ordin de convergență

O metodă are ordinul p dacă eroarea globală satisface $e(h) \le C \cdot h^p$, cu C independent de h. Pentru metoda Euler, ordinul rezultă din eroarea locală de trunchiere.

Propoziția 1.2. *Dacă $y \in C^2[t_0, T]$, eroarea locală de trunchiere a metodei Euler, $\tau_n = y(t_{n+1}) - y(t_n) - h \cdot f(t_n, y(t_n))$, satisface $|\tau_n| \le (h^2/2) \cdot M_2$, unde $M_2 = \max |y''(t)|$.*

Demonstrație. Din formula lui Taylor cu restul de tip Lagrange, $y(t_{n+1}) = y(t_n) + h \cdot y'(t_n) + (h^2/2) \cdot y''(\xi_n)$, cu $\xi_n \in (t_n, t_{n+1})$. Deoarece $y'(t_n) = f(t_n, y(t_n))$, rezultă $\tau_n = (h^2/2) \cdot y''(\xi_n)$, deci $|\tau_n| \le (h^2/2) \cdot M_2$. ■

Însumând cele $N = (T - t_0)/h$ erori locale, propagate cu factorul $e^{L(T - t_0)}$, se obține $e(h) = O(h)$, adică ordinul 1 [2]. În mod analog, metoda Heun are ordinul 2, iar RK4 ordinul 4 [1].

Nu prezenta o teoremă cunoscută ca rezultat propriu. Teoremele clasice se enunță cu trimitere la sursă; demonstrațiile scrise de tine (aici, Propoziția 1.2) trebuie să fie complete și cu notațiile din lucrare.
Greșeală frecventă
20/05/2026 10:00

Capitolul 2. Implementare și experimente numerice

Acesta este capitolul cu contribuția ta: implementarea, tabelele de erori, verificarea ordinului teoretic, costul și stabilitatea. Fiecare tabel este urmat de interpretare, iar capitolul se încheie cu analiza ta.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

2.1. Problema test

Am ales o problemă cu soluție exactă cunoscută, ca eroarea să poată fi calculată direct:

$$y' = -2t\cdot y,\quad y(0) = 1,\quad t \in [0, 2],\quad \text{cu soluția exactă } y(t) = e^{\wedge}(-t^2) \qquad (2.1)$$

Funcția f(t, y) = −2t·y este lipschitziană în y pe [0, 2], cu L = 4, deci problema are soluție unică (Teorema 1.1). Experimentul a urmat etapele din Figura 2.

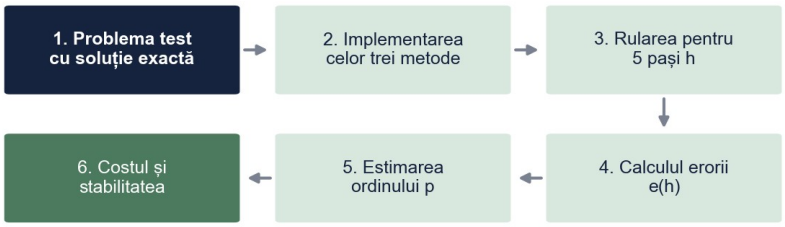


Figura 2. Etapele experimentului numeric
Sursa: Prelucrare proprie.

2.2. Implementarea

Metodele au fost implementate în Python, cu bibliotecile standard și NumPy [4]. Fiecare metodă este scrisă ca o funcție care face un singur pas, iar funcția `rezolva` le aplică pe toată rețeaua; scriptul complet de calcul al erorilor este în Anexa 1.

```
import numpy as np

def pas_euler(f, t, y, h):
    return y + h * f(t, y)

def pas_heun(f, t, y, h):
    k1 = f(t, y)
    k2 = f(t + h, y + h * k1)
    return y + h / 2 * (k1 + k2)

def pas_rk4(f, t, y, h):
    k1 = f(t, y)
    k2 = f(t + h / 2, y + h / 2 * k1)
    k3 = f(t + h / 2, y + h / 2 * k2)
    k4 = f(t + h, y + h * k3)
    return y + h / 6 * (k1 + 2 * k2 + 2 * k3 + k4)

def rezolva(pas, f, t0, y0, h, N):
    t = t0 + h * np.arange(N + 1)   # nodurile t_n
    y = np.empty(N + 1)
    y[0] = y0
    for n in range(N):
        y[n + 1] = pas(f, t[n], y[n], h)
    return t, y
```

Nu pune în lucrare capturi de ecran cu cod. Codul se scrie ca text, cu font monospațiat, iar în capitol rămân doar funcțiile pe care le explici; restul merge în anexă.
Sfat
20/05/2026 10:00

2.3. Precizia metodelor

Pentru pasul h = 0,2, aproximația RK4 nu se distinge vizual de soluția exactă, în timp ce metoda Euler supraestimează soluția pe porțiunea descendentă (Figura 3). Tabelul 4 dă valorile în patru noduri pentru h = 0,1: la t = 2, eroarea este de $6{,}29 \cdot 10^{-3}$ pentru Euler și de $6{,}81 \cdot 10^{-6}$ pentru RK4.

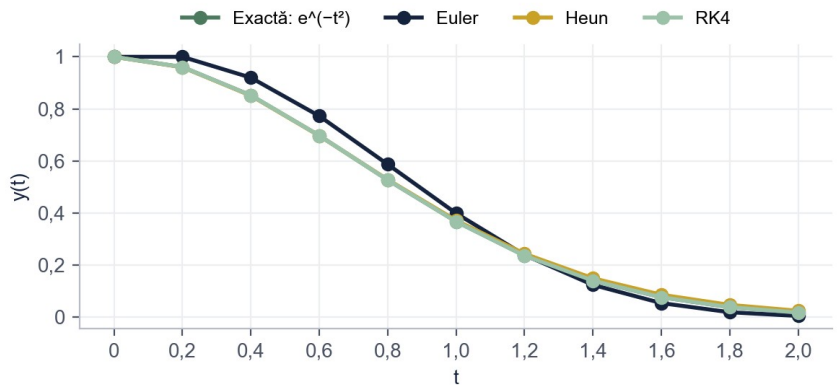


Figura 3. Soluția exactă și aproximațiile obținute cu pasul h = 0,2
Sursa: Calcul propriu în Python 3.12 (scriptul din Anexa 1).

Tabelul 4. Soluția exactă și aproximațiile numerice pentru h = 0,1

t_n	$y(t_n)$ exact	Euler	Heun	RK4
0,5	0,778801	0,813604	0,778765	0,778801
1,0	0,367879	0,381707	0,369053	0,367881
1,5	0,105399	0,096448	0,107451	0,105406
2,0	0,018316	0,012023	0,019573	0,018322

Notă. Valori rotunjite la șase zecimale.
Sursa: Calcul propriu în Python 3.12 (scriptul din Anexa 1).

2.4. Ordinul de convergență

Tabelul 5 dă eroarea globală maximă pentru cinci pași, fiecare fiind jumătate din precedentul. La fiecare înjumătățire a pasului, eroarea scade de aproximativ 2 ori la Euler, de 4 ori la Heun și de 16 ori la RK4. În scară logaritmică, punctele fiecărei metode sunt aproape coliniare (Figura 4).

Tabelul 5. Eroarea globală maximă e(h) pe intervalul [0, 2]

h	N	Euler	Heun	RK4
0,2	10	$7{,}51 \cdot 10^{-2}$	$9{,}58 \cdot 10^{-3}$	$1{,}50 \cdot 10^{-4}$
0,1	20	$3{,}48 \cdot 10^{-2}$	$2{,}05 \cdot 10^{-3}$	$7{,}47 \cdot 10^{-6}$
0,05	40	$1{,}69 \cdot 10^{-2}$	$4{,}80 \cdot 10^{-4}$	$4{,}18 \cdot 10^{-7}$
0,025	80	$8{,}28 \cdot 10^{-3}$	$1{,}16 \cdot 10^{-4}$	$2{,}47 \cdot 10^{-8}$
0,0125	160	$4{,}10 \cdot 10^{-3}$	$2{,}86 \cdot 10^{-5}$	$1{,}50 \cdot 10^{-9}$

Sursa: Calcul propriu în Python 3.12 (scriptul din Anexa 1).

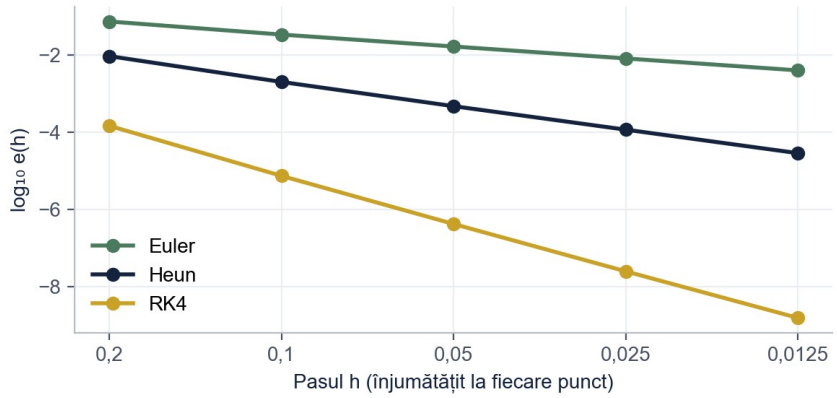


Figura 4. Logaritmul zecimal al erorii globale maxime, în funcție de pas

Sursa: Prelucrare proprie după Tabelul 5.

Estimarea $p \approx \log_2 [e(h)/e(h/2)]$ tinde spre ordinul teoretic pe măsură ce h scade: 1,01 pentru Euler, 2,02 pentru Heun și 4,04 pentru RK4 la ultima pereche de pași (Tabelul 6, Figura 5).

Tabelul 6. Ordinul de convergență estimat, $p \approx \log_2 [e(h) / e(h/2)]$

Perechea de pași	Euler	Heun	RK4
0,2 → 0,1	1,11	2,22	4,32
0,1 → 0,05	1,05	2,10	4,16
0,05 → 0,025	1,03	2,05	4,08
0,025 → 0,0125	1,01	2,02	4,04
Ordinul teoretic	1	2	4

Sursa: Calcul propriu după Tabelul 5.

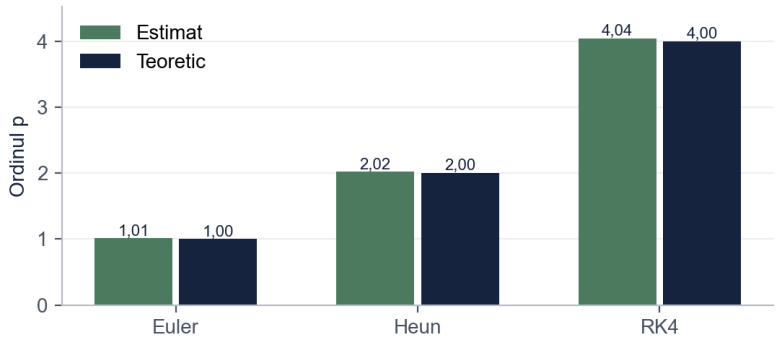


Figura 5. Ordinul de convergență estimat (perechea 0,025 → 0,0125) și ordinul teoretic
Sursa: Prelucrare proprie după Tabelul 6.

2.5. Costul de calcul

Pentru o eroare sub 10^{-6}, metoda Euler are nevoie de 650.596 de evaluări ale funcției f, metoda Heun de 1.702, iar RK4 de numai 132 (Tabelul 7, Figura 6). Deși RK4 face patru evaluări pe pas, ordinul mai mare compensează de multe ori acest cost.

Tabelul 7. Costul necesar pentru o eroare globală maximă sub 10^{-6}

Metoda	Număr minim de pași N	Evaluări ale lui f	Eroarea obținută
Euler	650.596	650.596	$1{,}00 \cdot 10^{-6}$
Heun	851	1.702	$9{,}99 \cdot 10^{-7}$
RK4	33	132	$9{,}23 \cdot 10^{-7}$

Notă. N a fost determinat prin căutare binară, ca cel mai mic număr de pași pentru care $e(h) < 10^{-6}$, cu $h = 2/N$.
Sursa: Calcul propriu în Python 3.12 (scriptul din Anexa 1).

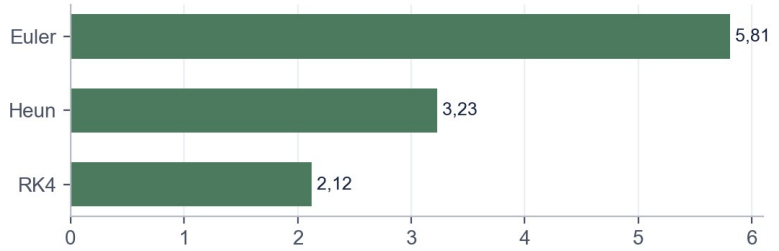


Figura 6. Numărul de evaluări ale lui f pentru $e(h) < 10^{-6}$ (logaritm zecimal)
Sursa: Prelucrare proprie după Tabelul 7.

Nu compara metodele doar la același pas h: o metodă cu mai multe evaluări pe pas pare favorizată. Comparația corectă se face la aceeași precizie sau la același număr de evaluări ale funcției, ca în Tabelul 7.
Greșeală frecventă
20/05/2026 10:00

2.6. Stabilitatea

Pentru ecuația test $y' = -15y$, soluția exactă scade rapid spre zero. Cu $h = 0{,}15$, metodele Euler și Heun au $|R(z)| > 1$, iar aproximațiile cresc în modul de la un pas la altul; RK4 rămâne

stabilă (Tabelul 8, Figura 7). După 10 pași, Euler dă 9,31 și Heun 11,92, în timp ce valoarea exactă y(1,5) = e^(−22,5) este de ordinul 10^{-10}.

Tabelul 8. Testul de stabilitate pentru y′ = −15y, y(0) = 1: factorul de amplificare R(z), z = −15h

Metoda	R(z)	h = 0,1 (z = −1,5)	h = 0,15 (z = −2,25)	Pas maxim stabil
Euler	1 + z	−0,500	−1,250 (instabil)	h < 0,133
Heun	1 + z + z²/2	0,625	1,281 (instabil)	h < 0,133
RK4	1 + z + z²/2 + z³/6 + z⁴/24	0,273	0,451	h < 0,186

Notă. Metoda este stabilă pentru acest test dacă |R(z)| < 1. Limitele: z > −2 pentru Euler și Heun, z > −2,785 pentru RK4.
Sursa: Calcul propriu după [2].

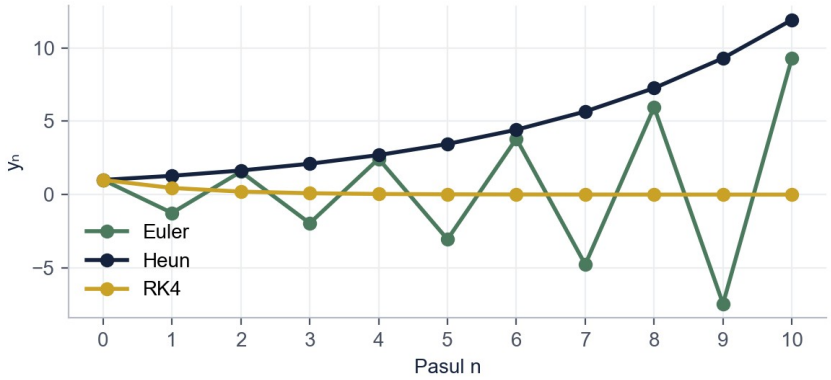


Figura 7. Aproximațiile pentru y′ = −15y cu pasul h = 0,15 (primii 10 pași)
Sursa: Calcul propriu în Python 3.12 (scriptul din Anexa 1).

2.7. Analiza autorului

Experimentele confirmă ordinele teoretice 1, 2 și 4. Considerăm că, pentru probleme nerigide cu soluție netedă, RK4 este alegerea potrivită: la o precizie de 10^{-6} este de aproape 5.000 de ori mai ieftină decât metoda Euler. Metoda Euler rămâne utilă didactic și ca punct de plecare pentru metodele implicite.

Testul de stabilitate arată însă o limită comună: toate metodele explicite cer un pas mic când problema are o componentă care scade rapid. Pentru astfel de probleme (rigide), propunem ca extindere studiul metodelor implicite, de exemplu Euler implicit sau formulele BDF.

Concluzii

Concluziile rezumă rezultatele, spun limitele abordării şi propun direcţii de continuare. La Matematică, leagă rezultatele numerice de teoria din primul capitol.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

- Ordinele de convergență estimate numeric (1,01; 2,02; 4,04) confirmă ordinele teoretice ale metodelor Euler, Heun și RK4.
- Pentru o eroare sub 10^{-6}, RK4 necesită 132 de evaluări ale lui f, față de 1.702 pentru Heun și 650.596 pentru Euler.
- Pe ecuația test $y' = -15y$, metodele Euler și Heun devin instabile pentru $h > 0{,}133$, iar RK4 pentru $h > 0{,}186$.

Limite şi direcţii ulterioare: studiul folosește o singură problemă test, scalară și nerigidă. Lucrarea poate fi extinsă la sisteme de ecuații, la metode cu pas adaptiv (perechi Runge–Kutta încorporate) și la metode implicite pentru probleme rigide.

Bibliografie

La Matematică se folosește cel mai des citarea numerică [1], cu sursele în ordinea primei citări. Lucrările [1], [2], [3] și [4] sunt reale; [5] este o sursă generică, inventată pentru acest exemplu.
Rolul secțiunii
20/05/2026 10:00

[1] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, ed. a 3-a. Chichester: Wiley, 2016.

[2] E. Hairer, S. P. Nørsett și G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*, ed. a 2-a. Berlin: Springer, 1993.

[3] R. L. Burden și J. D. Faires, *Numerical Analysis*, ed. a 9-a. Boston: Brooks/Cole, 2011.

[4] C. R. Harris et al., „Array programming with NumPy”, *Nature*, vol. 585, pp. 357–362, 2020.

[5] V. Neagu, *Analiză numerică. Curs și aplicații în Python*. București: Editura Exemplu, 2021.

Anexa 1. Scriptul pentru tabelul erorilor

În anexă pui codul complet care produce tabelele din lucrare, ca rezultatele să poată fi reproduse. Menționează versiunea limbajului și a bibliotecilor.
Rolul secțiunii
20/05/2026 10:00

```
import numpy as np
from metode import rezolva, pas_euler, pas_heun, pas_rk4   # Capitolul 2

f = lambda t, y: -2 * t * y
exact = lambda t: np.exp(-t**2)
T, pasi = 2.0, [0.2, 0.1, 0.05, 0.025, 0.0125]

for nume, pas in [('Euler', pas_euler), ('Heun', pas_heun), ('RK4', pas_rk4)]:
    erori = []
    for h in pasi:
        N = round(T / h)
        t, y = rezolva(pas, f, 0.0, 1.0, h, N)
        erori.append(np.max(np.abs(y - exact(t))))
    ordine = [np.log2(erori[i] / erori[i + 1]) for i in range(len(pasi) - 1)]
    print(nume, ['%.2e' % e for e in erori], ['%.2f' % p for p in ordine])
```

Rulat cu Python 3.12 și NumPy 2.1, scriptul afișează valorile din Tabelele 5 și 6.

Bonus: același paragraf, înainte și după corectare

Această pagină nu face parte din lucrare. Arată cum se schimbă un paragraf la o corectare atentă: aceeași idee, scrisă clar, cu sursă și în stil academic.
Ghid: licenta-disertatie-romania.co
Rolul secțiunii
20/05/2026 10:00

Fragmentul de mai jos este scris intenționat cu greșelile pe care le întâlnim des în lucrările de Matematică aplicată. Dedesubt este aceeași idee după corectare.

VARIANTA INIȚIALĂ

> Am rulat programul si se vede clar ca RK4 e cea mai buna metoda, eroarea e practic 0 (0.0000000015). Euler are erori mari deci nu e buna. Se stie ca RK4 are ordinul 4 deci rezultatele sunt corecte.

VARIANTA CORECTATĂ

> Pentru h = 0,0125, eroarea globală maximă a metodei RK4 este de $1{,}50 \cdot 10^{-9}$, față de $4{,}10 \cdot 10^{-3}$ pentru metoda Euler (Tabelul 5). Ordinul de convergență estimat din erorile pentru pași succesivi, 4,04 pentru RK4, confirmă ordinul teoretic 4 [1]. Metoda Euler are o precizie mai mică la același pas, dar costul ei pe pas este de patru ori mai mic; comparația corectă se face la aceeași precizie (Tabelul 7).

CE AM CORECTAT

1. Am pus diacriticele (ș, ț cu virgulă), obligatorii în lucrare.
2. „Practic 0” a devenit o valoare scrisă în notație științifică, cu pasul pentru care a fost obținută.
3. „Se vede clar” a devenit o trimitere la tabelul de erori.
4. Raționamentul circular („se știe că are ordinul 4, deci rezultatele sunt corecte”) a devenit o verificare: ordinul estimat numeric este comparat cu cel teoretic.
5. „Euler nu e bună” a devenit o comparație corectă, la aceeași precizie, ținând cont de costul pe pas.

Vrei aceeași corectare pentru lucrarea ta? licenta-disertatie-romania.com/contact