

Szacowanie błędu lokalnego w metodach jednokrokowych

$$u_n = u_{n-1} + \Delta t \phi(t_{n-1}, u_{n-1}, \Delta t)$$

Po co?

- 1) W rachunkach numerycznych musimy znać oszacowanie błędu
- 2) Gdy oszacowanie jest w miarę dokładne: można poprawić wynik
- 3) Aby ustawić krok czasowy tak, aby błąd był akceptowalny

Oszacowanie błędu lokalnego w metodach jednokrokowych

W każdym kroku generujemy nowy błąd w rachunkach.

Znamy jego rząd.

$$u(t_n) = u(t_{n-1}) + \Delta t f(t_{n-1}, u_{n-1}) + \frac{\Delta t^2}{2} [f'_t + f'_u f]_{(t_{n-1}, u_{n-1})} + \frac{\Delta t^3}{6} u'''(\xi_n)$$

dla RK: wstawialiśmy rozwiązanie dokładne do schematu i je rozwijaliśmy w szereg T.

$$u(t_n) = u(t_{n-1}) + \Delta t [b_1 f + b_2 (f + c\Delta t f'_t + a\Delta t f f'_u)] + O(\Delta t^3)$$

wybór $b_1=0, b_2=1, c=1/2, a=1/2$ dawał RK2 punktu środkowego

$$u_{n+1} = u_n + \Delta t f\left(t_n + \frac{\Delta t}{2}, u_n + \frac{\Delta t}{2} f(t_n, u_n)\right)$$

Rozwijając do jednego rzędu wyżej z Δt uzyskamy oszacowanie błędu lokalnego

$d_n = u(t_n) - u_n$ [przy założeniu, że $u(t_{n-1}) = u_{n-1}$]

$$d_n = \frac{\Delta t^3}{6} [3(f''_{tt} + 2f f''_{tu} + f^2 f''_{yy}) - (f'_t f'_y + f(f'_y)^2)]_{t_{n-1}, u_{n-1}} + O(\Delta t^4)$$

światny wzór choć mało praktyczny

metodą rzędu p z chwili t_{n-1} wykonujemy krok do t_n

$$u_n = u_{n-1} + \Delta t \Phi(t_{n-1}, u_{n-1}, \Delta t)$$

$$u(t_n) - u_n = d_n$$

$$d_n = C_n \Delta t^{p+1} + O(\Delta t^{p+2})$$

może zależeć od t_{n-1} oraz u_{n-1} , ale nie zależy od Δt

$$d_n = \frac{\Delta t^3}{6} [3(f''_{tt} + 2ff''_{tu} + f^2 f''_{yy}) - (f'_t f'_y + f(f'_y)^2)]_{t_{n-1}, u_{n-1}} + O(\Delta t^4)$$

folia wcześniej

szacowanie błędu:

- 1) ekstrapolacja Richardsona (*step doubling*)
- 2) osadzanie (*embedding*)

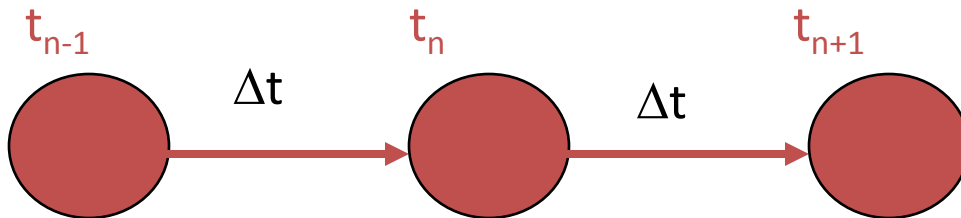
ekstrapolacja
Richardsona

$$u_n = u_{n-1} + \Delta t \Phi(t_{n-1}, u_{n-1}, \Delta t)$$

$$u(t_n) - u_n = d_n$$

$$d_n = C_n \Delta t^{p+1} + O(\Delta t^{p+2})$$

dwa kroki Δt : dostaniemy lepsze oszacowanie $u(t_{n+1})$



jeden krok $2\Delta t$: dostaniemy gorsze oszacowanie $u(t_{n+1})$



szacujemy C_n z porównania obydwu rozwiązań

ekstrapolacja Richardsona

$$u_n = u_{n-1} + \Delta t \Phi(t_{n-1}, u_{n-1}, \Delta t)$$

błąd lokalny $u(t_n) - u_n = d_n$ jest:

$$d_n = C_n \Delta t^{p+1} + O(\Delta t^{p+2})$$

wykonujemy krok następny od t_n do t_{n+1}

$$u_{n+1} = u_n + \Delta t \Phi(t_n, u_n, \Delta t)$$

odchylenie wyniku numerycznego od dokładnego $u(t_{n+1}) - u_{n+1} = \gamma d_n + d_{n+1}$

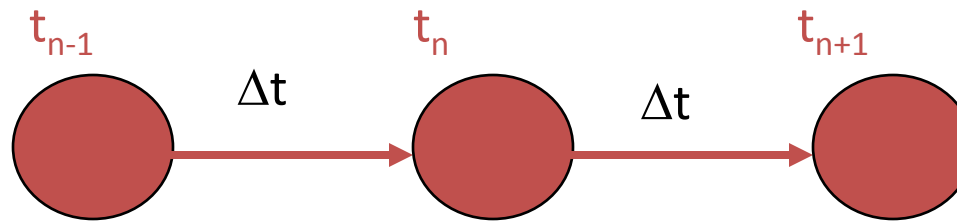
1) zakładamy, że krok jest na tyle mały, że stała błędu się nie zmienia $C_n \approx C_{n+1}$
(lub, że w jednym kroku zmienia się o $O(\Delta t)$]

wtedy błąd lokalny popełniony w chwili t_{n+1} jest $d_{n+1} \approx d_n$.

2) gdy krok mały: współczynnik wzmocnienia błędu $\gamma \approx 1$
(błąd popełniony w kroku pierwszym nie jest istotnie wzmocniany)

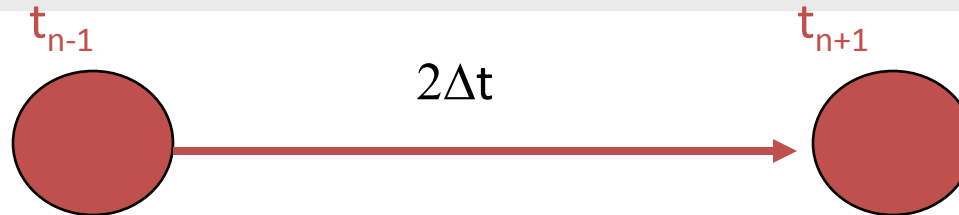
Przy tym założeniu: błąd po drugim kroku – suma błędów $\gamma d_n + d_{n+1} \approx 2d_n$,

$$u(t_{n+1}) - u_{n+1} = 2C_n (\Delta t)^{p+1} + O(\Delta t)^{p+2}$$



$$u(t_{n+1}) - u_{n+1} = 2C_n(\Delta t)^{p+1} + O(\Delta t^{p+2})$$

to chwili t_{n+1} dojdziemy z t_{n-1} w pojedynczym kroku $2\Delta t$



dostaniemy gorsze oszacowanie $u(t_{n+1})$

$$\tilde{u}_{n+1} = u_{n-1} + 2\Delta t \Phi(t_{n-1}, u_{n-1}, 2\Delta t)$$

$$\tilde{d}_n = C_n(2\Delta t)^{p+1} + O((2\Delta t)^{p+2})$$

$$u(t_{n+1}) - \tilde{u}_{n+1} = C_n(2\Delta t)^{p+1} + O(2\Delta t^{p+2})$$

chcemy poznać C_n (to + znajomość p da nam oszacowanie błędu):
odejmujemy niebieskie wzory tak aby wyeliminować rozwiązanie
dokładne (nam niedostępne)

$$u(t_{n+1}) - u_{n+1} = 2C_n(\Delta t)^{p+1} + O(\Delta t^{p+2})$$

$$u(t_{n+1}) - \tilde{u}_{n+1} = C_n(2\Delta t)^{p+1} + O(2\Delta t^{p+2})$$



$$u_{n+1} - \tilde{u}_{n+1} = C_n [(2\Delta t)^{p+1} - 2(\Delta t)^{p+1}]$$

$$C_n = \frac{u_{n+1} - \tilde{u}_{n+1}}{(2\Delta t)^{p+1} - 2(\Delta t)^{p+1}}$$

$$u(t_{n+1}) - u_{n+1} = 2C_n(\Delta t)^{p+1} + O(\Delta t)^{p+2}$$

błąd wykonany po dwóch krokach Δt wynosi więc:

$$d = u(t_{n+1}) - u_{n+1} = \frac{u_{n+1} - \tilde{u}_{n+1}}{2^p - 1} + O(\Delta t^{p+2})$$

pierwszy wniosek: jeśli znamy rząd metody p to potrafimy go podnieść o jeden

$$u(t_{n+1}) = u_{n+1} + \frac{u_{n+1} - \tilde{u}_{n+1}}{2^p - 1} + O(\Delta t^{p+2})$$

$$u(t_{n+1}) = u_{n+1} + \frac{u_{n+1} - \tilde{u}_{n+1}}{2^p - 1} + O(\Delta t^{p+2})$$

podnosimy rząd dokładności metody
„algorytm”

do $n = 1, N$

$$u_n = u_{n-1} + \Delta t \phi(t_{n-1}, u_{n-1}, \Delta t)$$

$$u_{n+1} = u_n + \Delta t \phi(t_n, u_n, \Delta t)$$

$$\tilde{u}_{n+1} = u_{n-1} + 2\Delta t \phi(t_{n-1}, u_{n-1}, 2\Delta t)$$

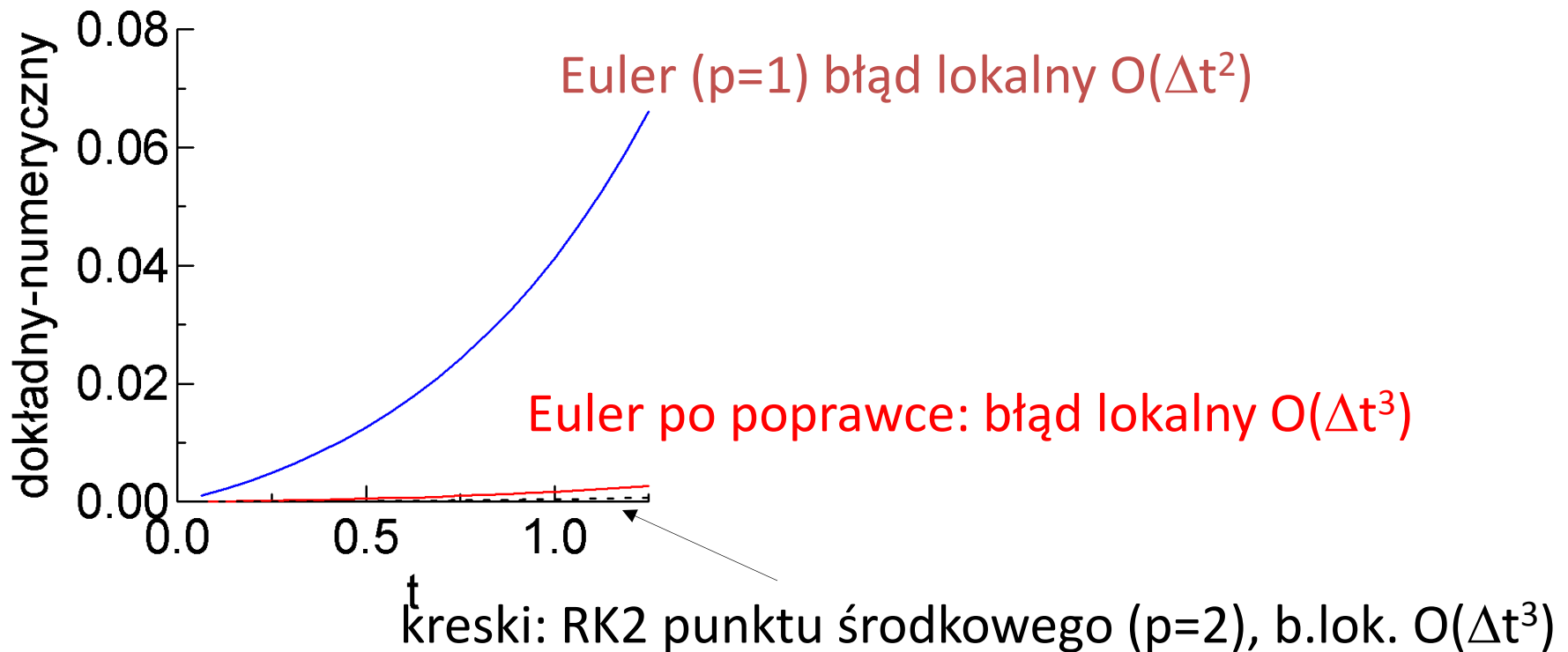
$$u_{n+1} = u_{n+1} + \frac{u_{n+1} - \tilde{u}_{n+1}}{2^p - 1}$$

enddo

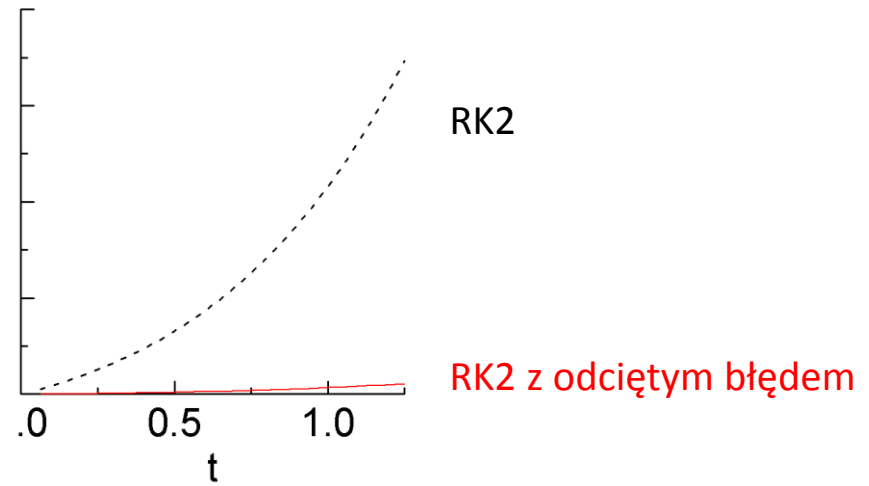
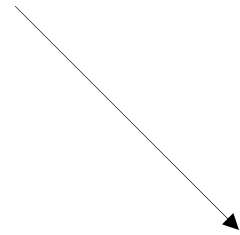
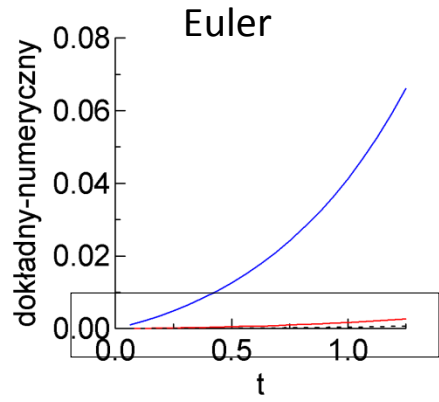
$$\frac{du}{dt} = u \longrightarrow \text{r. dokładne: } u(t) = \exp(t)$$

$$u(0) = 1$$

$$\Delta t = \frac{1}{32}$$



znając rząd dokładności możemy radykalnie poprawić dokładność metody przy natdatku (50 procent) numeryki



Oszacowanie błędu lokalnego w metodach jednokrokowych

- 1) ekstrapolacja Richardsona (*step doubling*)
- 2) osadzanie (*embedding*)

cel: szacujemy błąd lokalny metody rzędu p
przy pomocy lepszej metody, np. rzędu $p+1$
obydwie metody szacują rozwiązanie w tych samych chwilach czasowych

$$u_n^p = u_{n-1} + \Delta\phi_p(t_{n-1}, y_{n-1}, \Delta t) \qquad d_n^p = C_n^p (\Delta t)^p$$

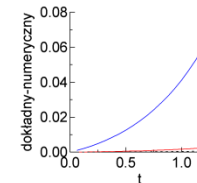
$$u_n^{p+1} = u_{n-1} + \Delta\phi_{p+1}(t_{n-1}, y_{n-1}, \Delta t) \qquad d_n^{p+1} = C_n^{p+1} (\Delta t)^{p+1}$$

$$d_n^p = u(t_n) - u_n^p = u(t_n) - u_n^{p+1} + u_n^{p+1} - u_n^p = d_n^{p+1} + u_n^{p+1} - u_n^p$$

$$\underline{|d_n^p|} \leq \underline{|\cancel{d_n^{p+1}}|} + |u_n^{p+1} - u_n^p|$$

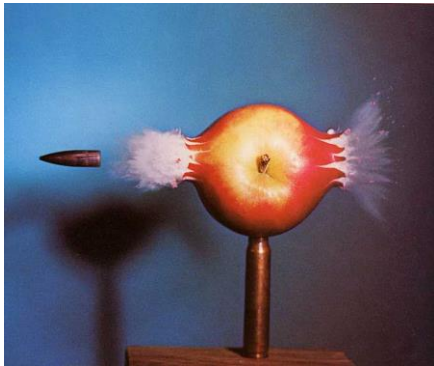
co daje oszacowanie błędu gorszej metody

$$|d_n^p| \simeq |u_n^{p+1} - u_n^p|$$

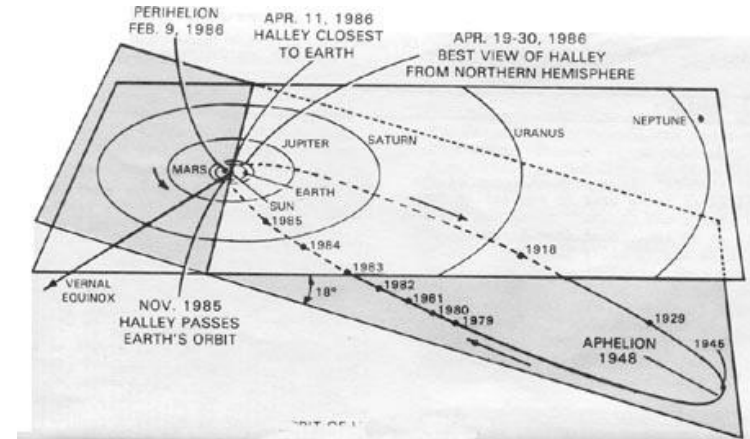


nie nadaje się do poprawiania schematu p po cóż zresztą poprawiać gdy mamy $p+1$

celem szacowania błędu nie jest poprawa wyniku,
(dla poprawy zawsze można Δt zmienić)
lecz adaptacja Δt :
stały krok zawsze może okazać się zbyt wielki albo zbyt mały.



JAKI KROK CZASOWY SYMULACJI USTAWIĆ
gdy coś ciekawego zdarza się tylko czasem ?



Automatyczna kontrola kroku czasowego dla metod jednokrokowych

Program może sam dobierać krok czasowy w zależności od tego co dzieje się w symulacji.

$$u_n = u_{n-1} + \Delta t \Phi(t_{n-1}, u_{n-1}, \Delta t)$$

Chcemy utrzymać błąd na poziomie zbliżonym do parametru tol .
*nie większy aby zachować wymaganą dokładność,
nie mniejszy aby nie tracić czasu na rachunki zbyt dokładne*

Szacujemy błąd lokalny E (ekstrapolacja Richardsona lub metody *embedding*)

$$E = C[\Delta t]^{p+1}$$

chcemy zmienić krok odpowiednio do naszych wymagań z Δt do $\Delta t(\text{nowy})$

$$tol = C[\Delta t(\text{nowy})]^{p+1} \longrightarrow \Delta t(\text{nowy}) = (tol/E)^{1/(p+1)} \Delta t$$

$$\Delta t(\text{nowy}) = (S \, tol / E)^{1/(p+1)} \Delta t \quad \text{dla bezpieczeństwa } S < 1$$

wzór zwiększy zbyt mały krok i vice versa

uwaga: błąd jest szacowany, zawsze warto dorzucić sztywne ograniczenia na Δt

symulacja ustawiająca krok czasowy może wyglądać np. tak:

u_0 = warunek początkowy

$t_0 = 0$

$n = 1$

do {

$$u_n = u_{n-1} + \Delta t \Phi(t_{n-1}, u_{n-1}, \Delta t)$$

jeśli $E < \text{tol}$

{ $t_n := t_n + \Delta t$

$n := n + 1$ (oznacza akceptację wyniku) }

$$\Delta t := (S \text{ tol} / E)^{1/(p+1)} \Delta t$$

} while ($t < T$)

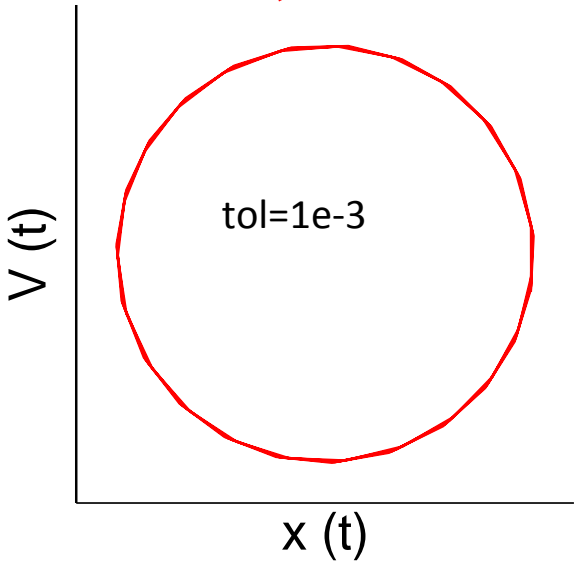
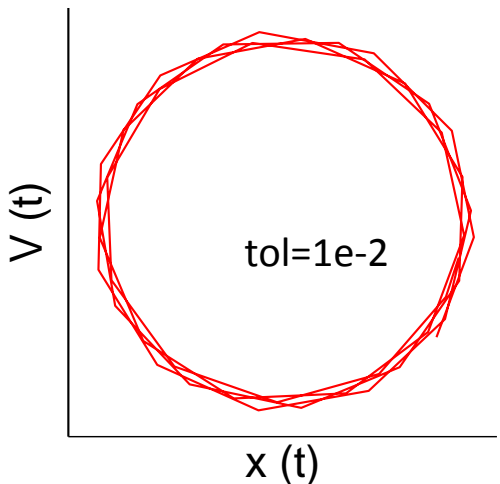
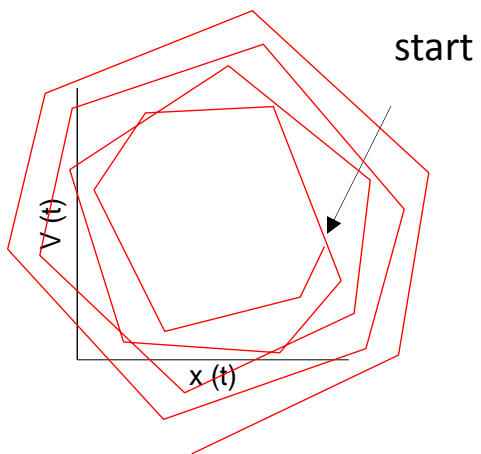
Przykład: oscylator harmoniczny
 używane oszacowanie błędu z RK2

Uwaga: tutaj rozwiązania nie poprawiamy
 przez ekstrapolacje

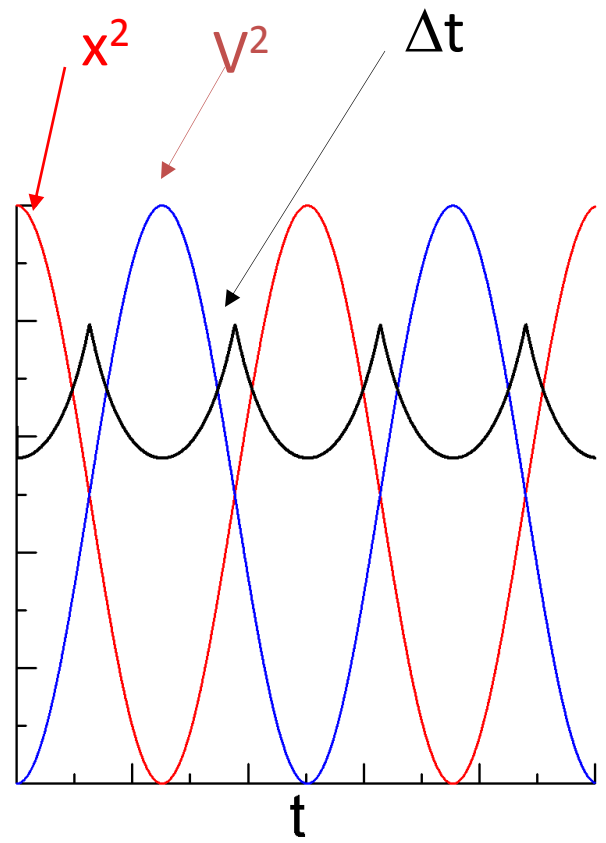
$$\frac{dx}{dt} = V(t) \quad x(0) = 1$$

$$\frac{dV}{dt} = -x(t) \quad V(0) = 0$$

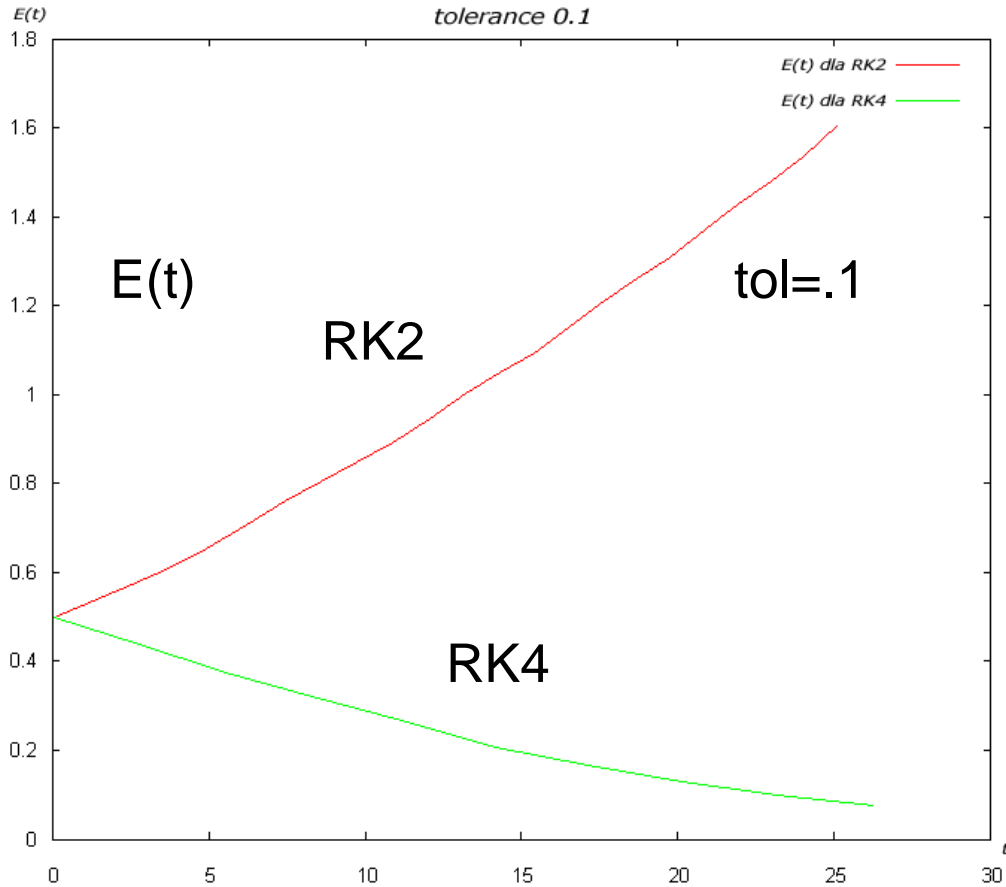
tolerancja błędu obciążenia
 tol=0.1 **RK2**



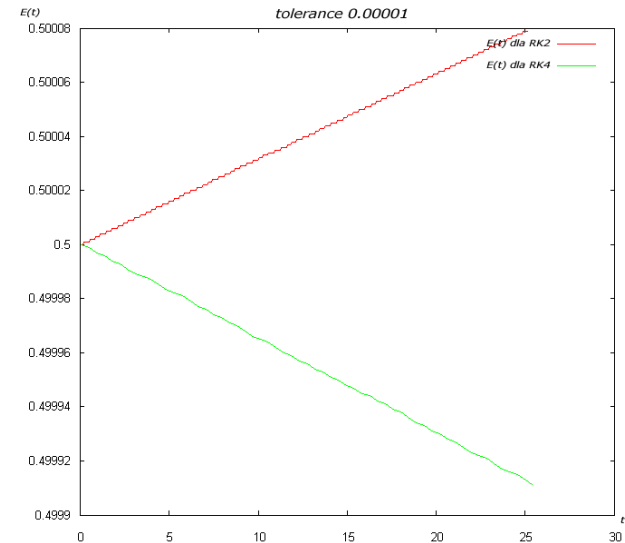
algorytm ustawia
 minimalny
 krok czasowy
 gdy zmiany prędkości
 lub położenia są maksymalne



wyniki Konrada Rekiecia



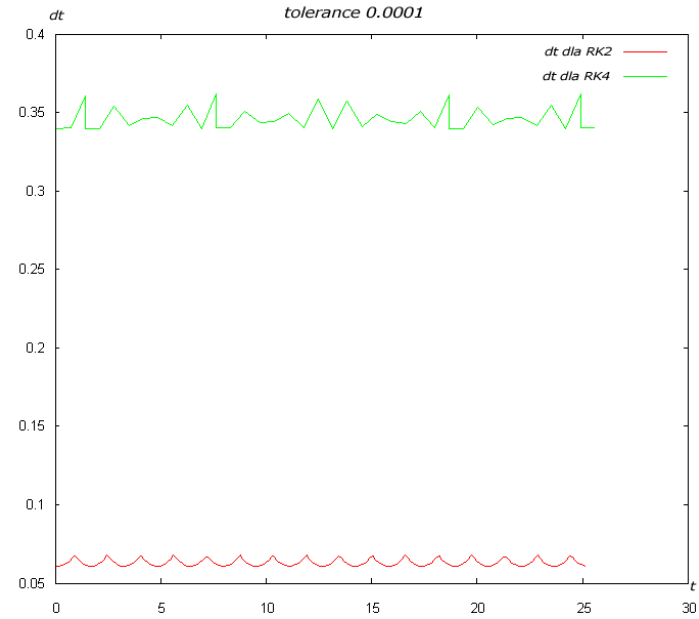
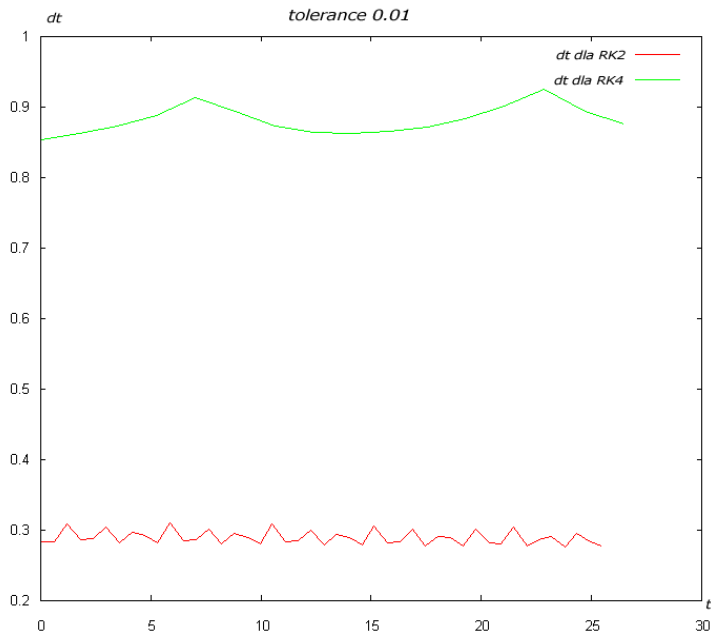
RK4 – spirala się skręca zamiast rozkręcać



przy założonej tolerancji
RK4 wcale nie jest
dokładniejsze od RK2

... tylko pozwala stawiać dłuższe kroki

dt

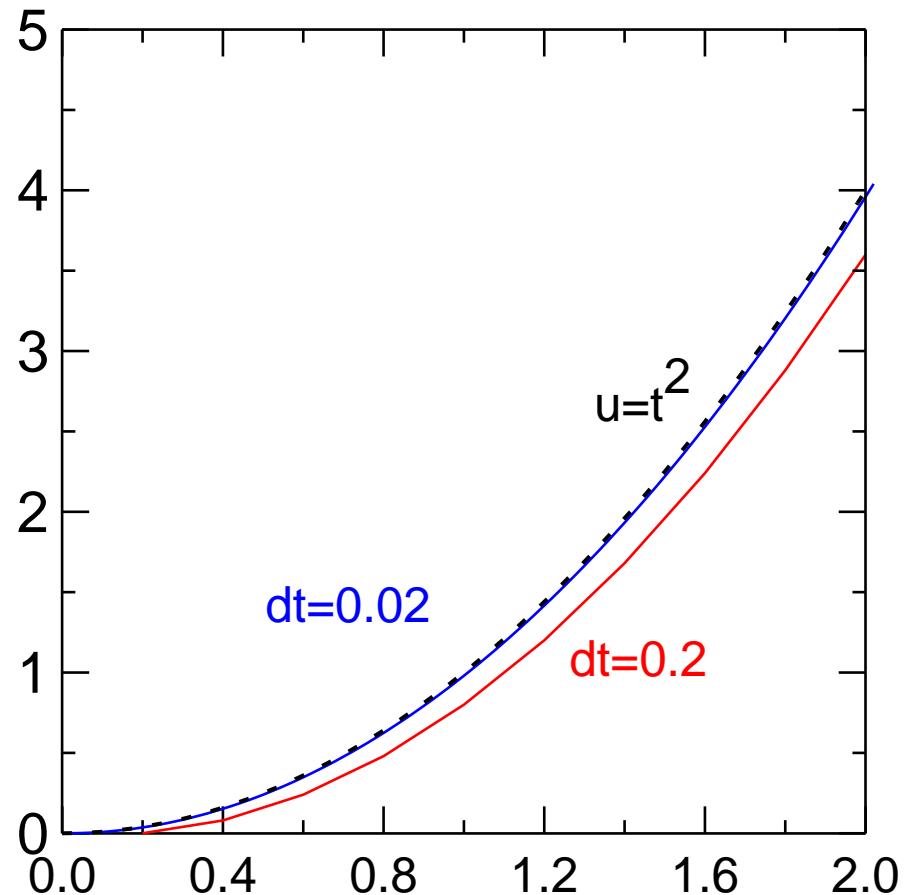


Problemy sztywne

$$\frac{du}{dt} = 2t \quad u(0)=0$$

$$u(t + dt) = u(t) + dt \times 2t$$

proste równanie
traktowane jawnym
schematem Eulera



prosty problem
nieco komplikujemy

$$u' = -\alpha(u - t^2) + 2t$$

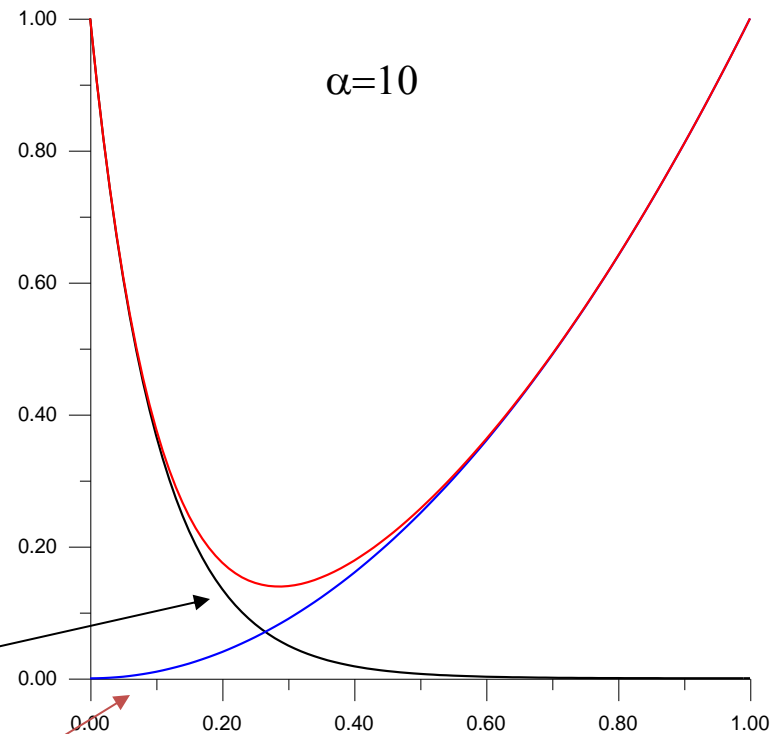
niech $\alpha \gg 0$

$$u(0) = u_0$$

$$u(t) = u_0 \exp(-\alpha t) + t^2$$

szybkozmienna
składowa

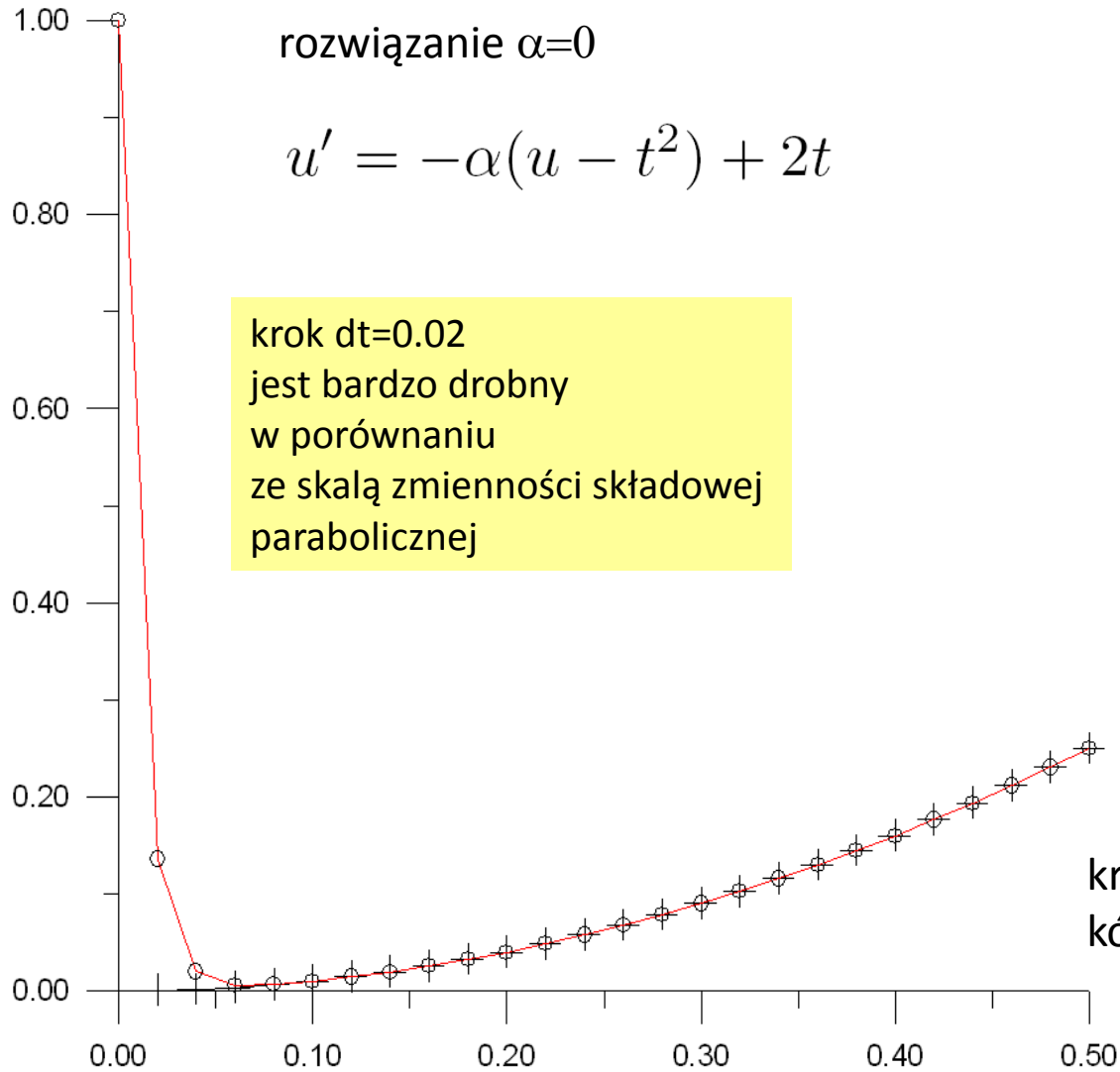
składowa
wolnozmienna



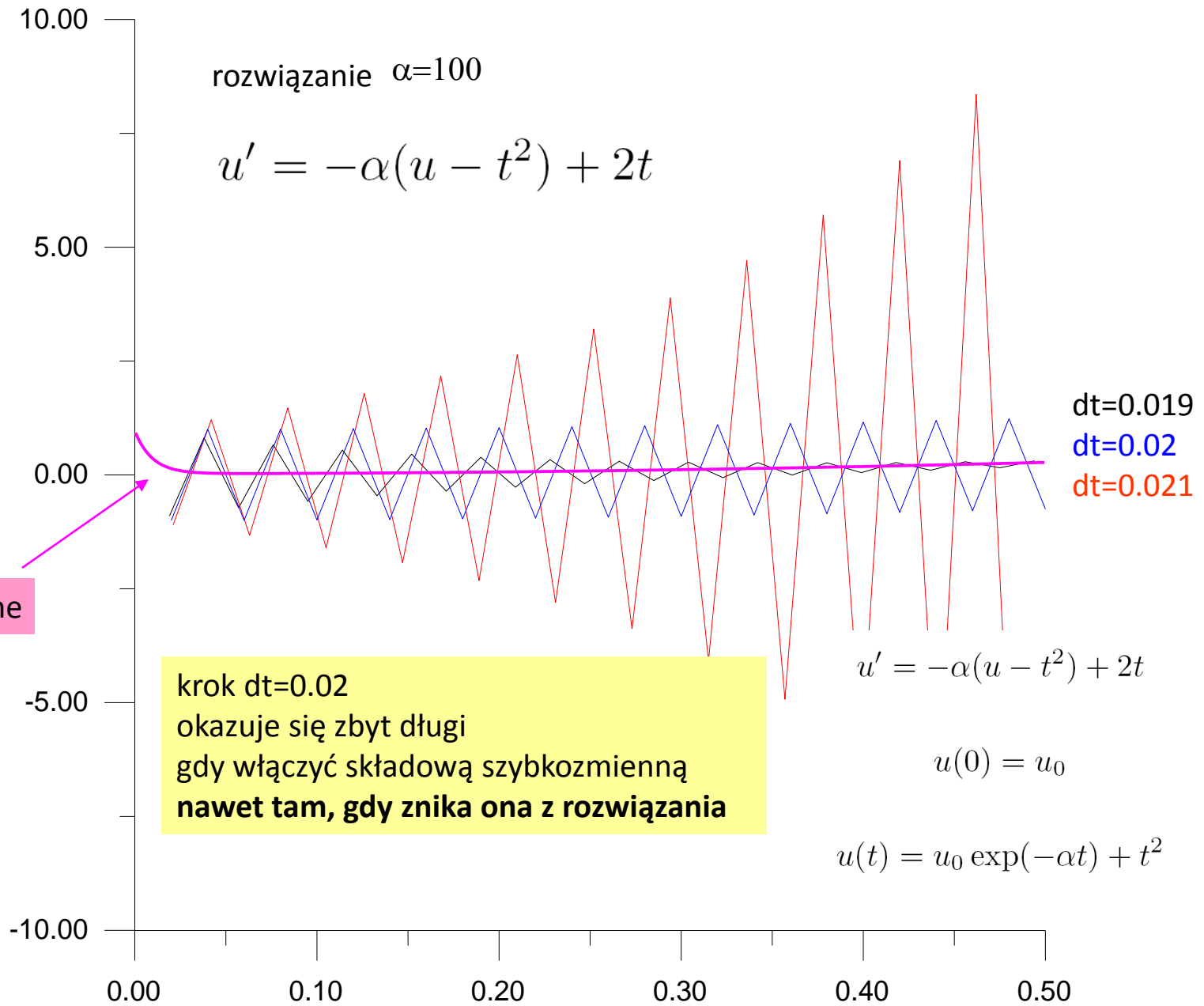
rozwiązanie $\alpha=0$

$$u' = -\alpha(u - t^2) + 2t$$

krok $dt=0.02$
jest bardzo drobny
w porównaniu
ze skalą zmienności składowej
parabolicznej



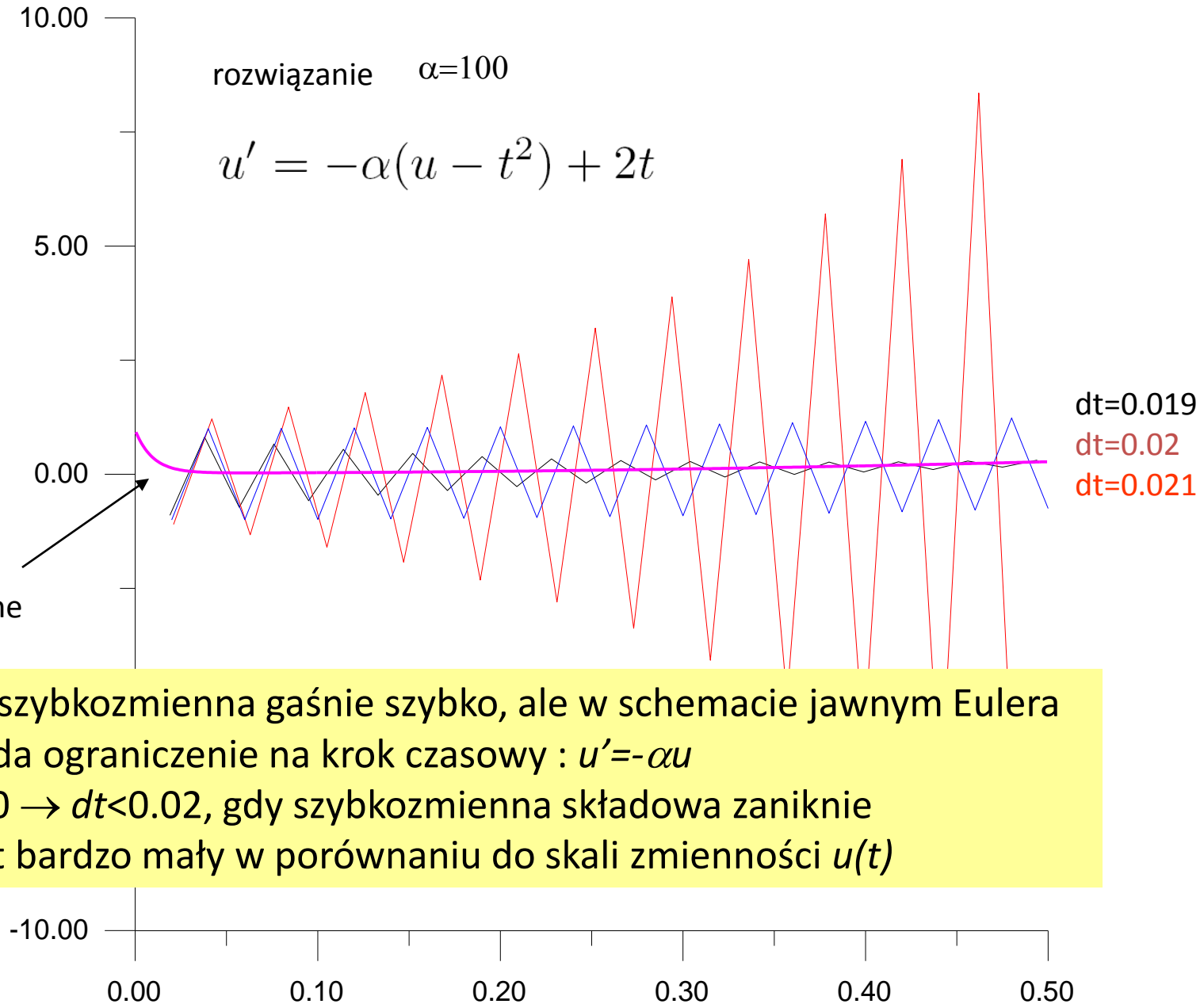
krzyżyki : t^2
kółka : $t^2 + \exp(-100t)$



rozwiązanie $\alpha=100$

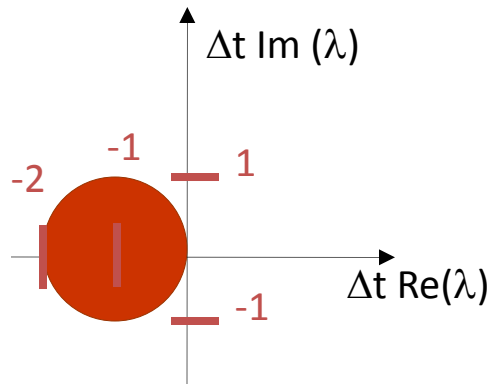
$$u' = -\alpha(u - t^2) + 2t$$

dokładne

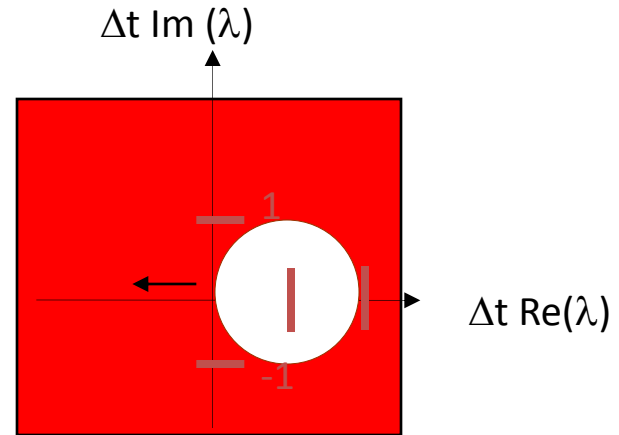


część szybkozmienna gaśnie szybko, ale w schemacie jawnym Eulera nakłada ograniczenie na krok czasowy : $u' = -\alpha u$
 $\alpha=100 \rightarrow dt < 0.02$, gdy szybkozmienna składowa zaniknie
 dt jest bardzo mały w porównaniu do skali zmienności $u(t)$

regiony stabilności metod Eulera



metoda Eulera jawna



niejawna metoda Eulera

w metodzie niejawniej problemu ze stabilnością bezwzględną nie ma ...

niejawna metoda Eulera:
zastosowanie do problemu sztywnego

$$u' = -\alpha(u - t^2) + 2t$$

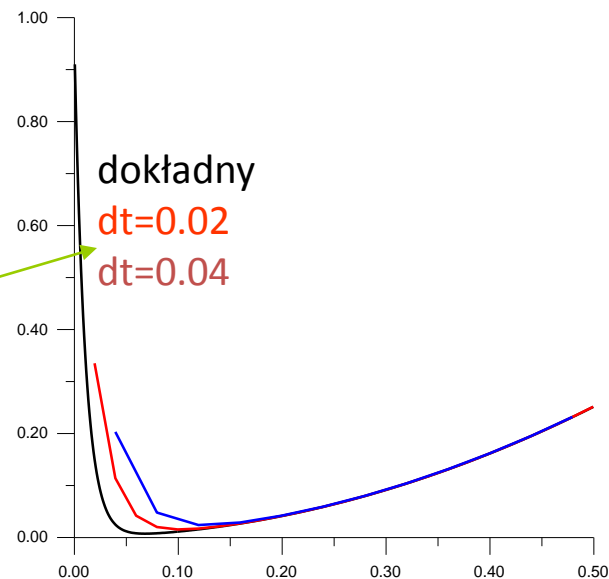
$$\frac{du}{dt} = f(t, u)$$

$$u(0) = u_0$$

$$u_n = u_{n-1} + f(t_n, u_n) \Delta t$$

$$u_n = \frac{u_{n-1} + 2\Delta t t_n + \alpha \Delta t t_n^2}{1 + \alpha \Delta t}$$

rozwiązania są stabilne
i dokładne dla dużych t nawet gdy Δt duże
dla małych t można wstawić mniejsze Δt ,
potem krok zwiększyć



Problemy sztywne (drętwe) (*stiff, stiffness*)

Problem jest praktyczny i ścisłej definicji, która byłaby użyteczna, nie ma .
Jedną z możliwych: problem jest sztywny, gdy stosując schemat jawny musimy przyjąć krok czasowy bardzo mały w porównaniu ze skalą zmienności funkcji.

RRZ jest problemem sztywnym gdy:

1. Problem jest charakteryzowany bardzo różnymi skalami czasowymi
2. Stabilność bwwz nakłada silniejsze ograniczenia na krok czasowy niż dokładność.
3. Metody jawne się nie sprawdzają.

$$u' = -\alpha(u - t^2) + 2t$$

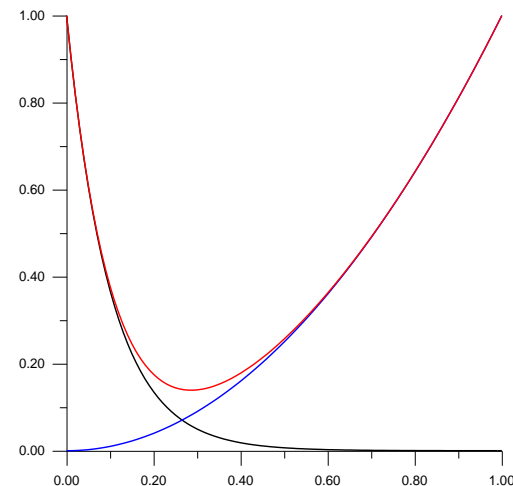
$$u(0) = u_0$$

$$u(t) = u_0 \exp(-\alpha t) + t^2$$

niech $\alpha \gg 0$

szybkozmienna
składowa

składowa
wolnozmienna



Problemy sztywne (drętwe) (*stiff*)

problem najczęściej spotykany dla układ równań różniczkowych opisujących sprzężone procesy o bardzo różnych skalach czasowych

Ogólna postać układu równań pierwszego rzędu

$$\frac{d\mathbf{y}}{dt} = \mathbf{F}(t, \mathbf{y})$$

fcja $\mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ wektor \mathbb{R}^n

Tylko niekiedy można podać rozwiązanie w zamkniętej formie analitycznej.
Można, np. dla jednorodnego problemu liniowego

$$\frac{d\mathbf{y}}{dt} = \mathbf{A}\mathbf{y}(t)$$

$$\frac{d\mathbf{y}}{dt} = \mathbf{A}\mathbf{y}(t) \quad \text{gdzie} \quad \mathbf{A}\mathbf{v}_j = \lambda_j\mathbf{v}_j$$

$$\mathbf{y}(t) = \sum_{j=1}^n c_j \exp(\lambda_j t) \mathbf{v}_j \quad \leftarrow \begin{array}{l} \text{dla niezdegenerowanych wartości własnych} \\ c_j \text{ liczone z warunku początkowego} \end{array}$$

np. problem rozpadu promieniotwórczego

Izotop 2 o stałej rozpadu λ_2 rozpada się promieniotwórczo na inny izotop 1 o stałej rozpadu λ_1

$$\begin{aligned} \frac{dy_1}{dt} &= -\lambda_1 y_1(t) + \lambda_2 y_2(t), & y_1(0) &= 0 \\ \frac{dy_2}{dt} &= -\lambda_2 y_2(t), & y_2(0) &= 1 \end{aligned}$$

$$\mathbf{A} = \begin{pmatrix} -\lambda_1 & \lambda_2 \\ 0 & -\lambda_2 \end{pmatrix} \rightarrow \begin{array}{l} \text{wartości własne } -\lambda_1, -\lambda_2 \\ \text{rozłożyć warunek początkowy} \\ \text{na wektory własne} \end{array}$$

$$\mathbf{y}(t) = \frac{\lambda_2}{\lambda_1 - \lambda_2} \exp(-\lambda_2 t) \begin{bmatrix} 1 \\ \frac{\lambda_1 - \lambda_2}{\lambda_2} \end{bmatrix} - \frac{\lambda_2}{\lambda_1 - \lambda_2} \exp(-\lambda_1 t) \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\mathbf{y}(t) = \frac{\lambda_2}{\lambda_1 - \lambda_2} \exp(-\lambda_2 t) \begin{bmatrix} 1 \\ \frac{\lambda_1 - \lambda_2}{\lambda_2} \end{bmatrix} - \frac{\lambda_2}{\lambda_1 - \lambda_2} \exp(-\lambda_1 t) \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

gdy duża rozpiętość między minimalną
a maksymalną wartością własną $|\lambda_{\max}/\lambda_{\min}| \gg 1$:

duże różnice skal czasowych

wektor własny który odpowiada największej wartości własnej wygaśnie najprędzej,
ale (dla metod jawnych) pozostawi najsilniejsze ograniczenie dla kroku czasowego
(np. Euler, RK2 $dt < 2/|\lambda_{\max}|$)

jesteśmy zmuszeni przyjąć mały krok w porównaniu z przebiegiem rozwiązania
(w przeciwnym wypadku eksplozja)

następny przykład:

podobny do poprzedniego

problem sztywny z liniowego równania drugiego rzędu o bliskich współczynnikach

$$u'' + 1001u' + 1000u = 0$$

$$\begin{array}{l} w_1 = u \\ w_2 = u' \end{array} \quad \longrightarrow \quad \begin{array}{l} w_2' + 1001w_2 + 1000w_1 = 0 \\ w_1' = w_2 \end{array}$$

$$\begin{pmatrix} w_1' \\ w_2' \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1000 & -1001 \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

wartości / wektory własne: $-1 / [-1, 1]^T$
 $-1000 / [1, -1000]^T$

bardzo różne
skale czasowe

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = c \exp(-t) \begin{pmatrix} -1 \\ 1 \end{pmatrix} + d \exp(-1000t) \begin{pmatrix} 1 \\ -1000 \end{pmatrix}$$

szczególnie dotkliwy przypadek: równanie niejednorodne (bez rozwiązania analitycznego)

$$\frac{d\mathbf{y}}{dt} = \mathbf{A}\mathbf{y}(t) + \phi(t) \quad \text{załóżmy, że wartości własne } \mathbf{A} \text{ są ujemne}$$

Rozwiązanie będzie miało postać:

$$\mathbf{y}(t) = \underbrace{\sum_{j=1}^n c_j \exp(\lambda_j t) \mathbf{v}_j}_{\text{stan przejściowy (wszystkie zgasną)}} + \Psi(t)$$

stan ustalony wolnozmienny

Na czym polega problem? :

Rozwiązując problem numerycznie metodą jawną (Euler, RK2)

musimy przyjąć krok czasowy $\Delta t < 2/|\lambda_{\max}|$ aby uniknąć eksplozji rozwiązań
nawet gdy wszystkie wyrazy z powyższej sumy w rozwiązaniu znikają

y_2 – izotop matka wolno rozpadająca się na y_1

y_1 – izotop szybko rozpadający się, **niejednorodność: dodatkowo pewna ilość jest w stałym tempie doprowadzana z zewnątrz**

$$\frac{dy_1}{dt} = -\lambda_1 y_1 + \lambda_2 y_2 + \lambda_1/2$$

$$\frac{dy_2}{dt} = -\lambda_2 y_2,$$

$$y_2(0)=1$$

$$y_1(0)=0$$

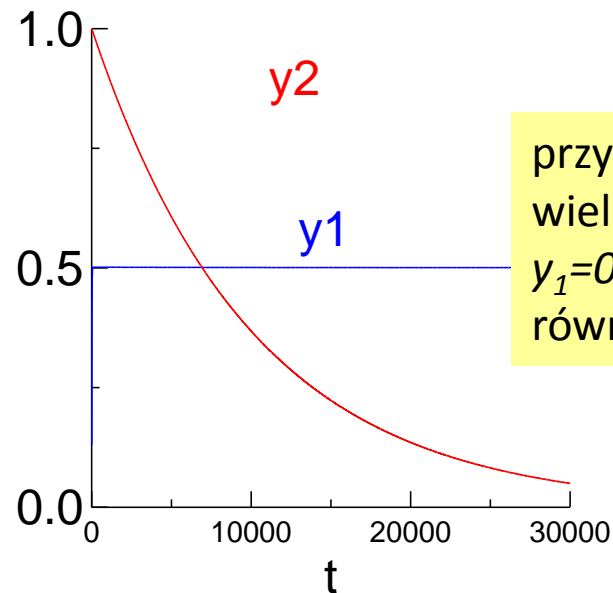
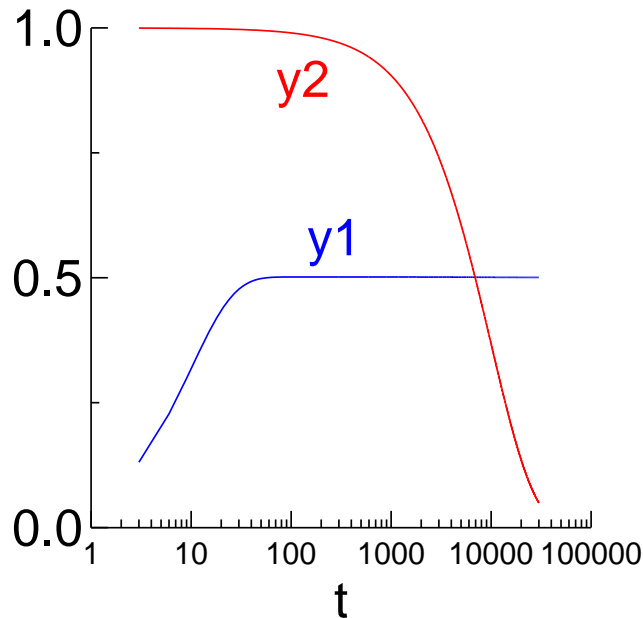
$$\lambda_1=1/10$$

$$\lambda_2=1/10\ 000$$

← bardzo wolno się rozpada [taka i większa rozpiętość lambda typowa

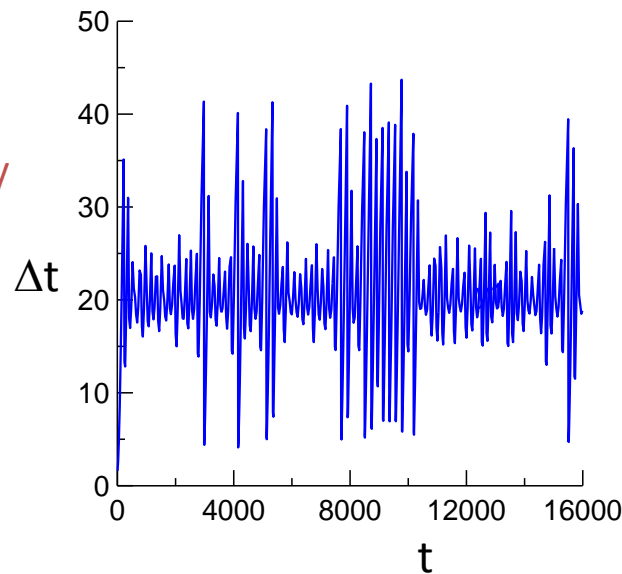
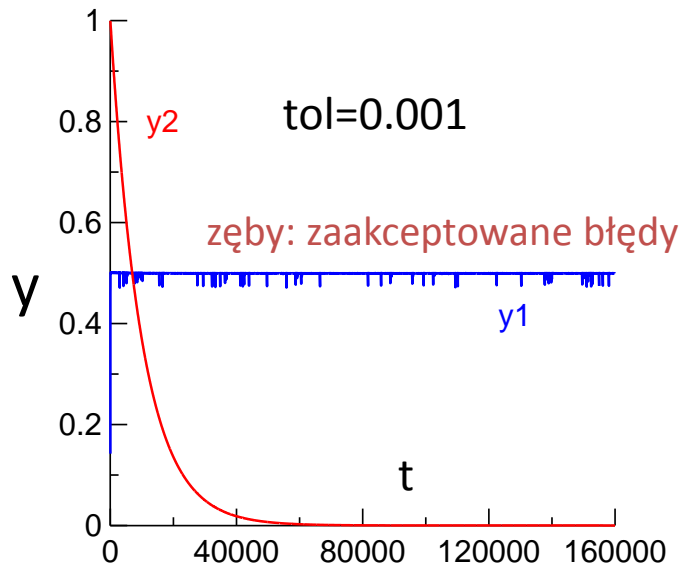
również dla reakcji chemicznych

spotykana również dla układów elektrycznych]

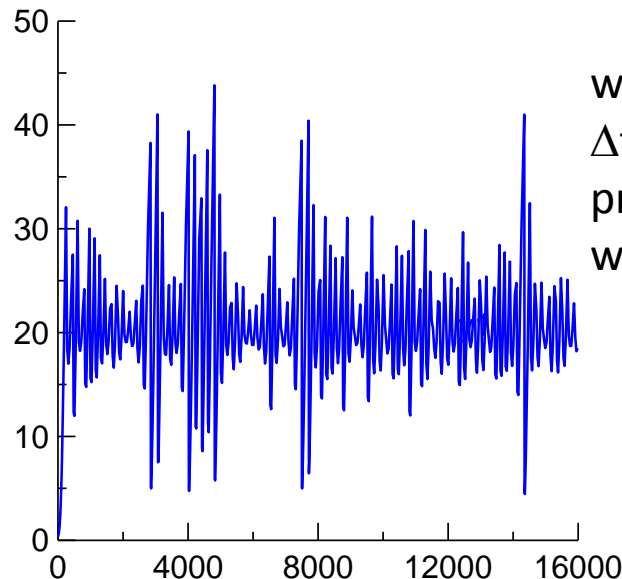
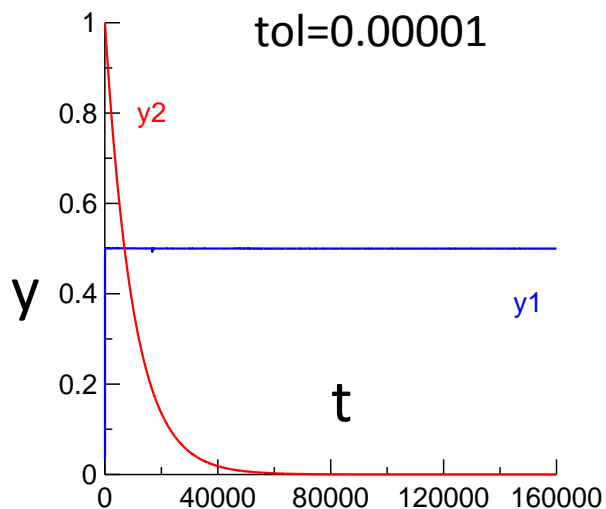


przy zaniedbywalnej wielkości λ_2
 $y_1=0.5$ spełnia pierwsze równanie

automatyczna kontrola kroku czasowego dla jawnego RK2
z krokiem czasowym ustawianym przez ekstrapolację Richardsona

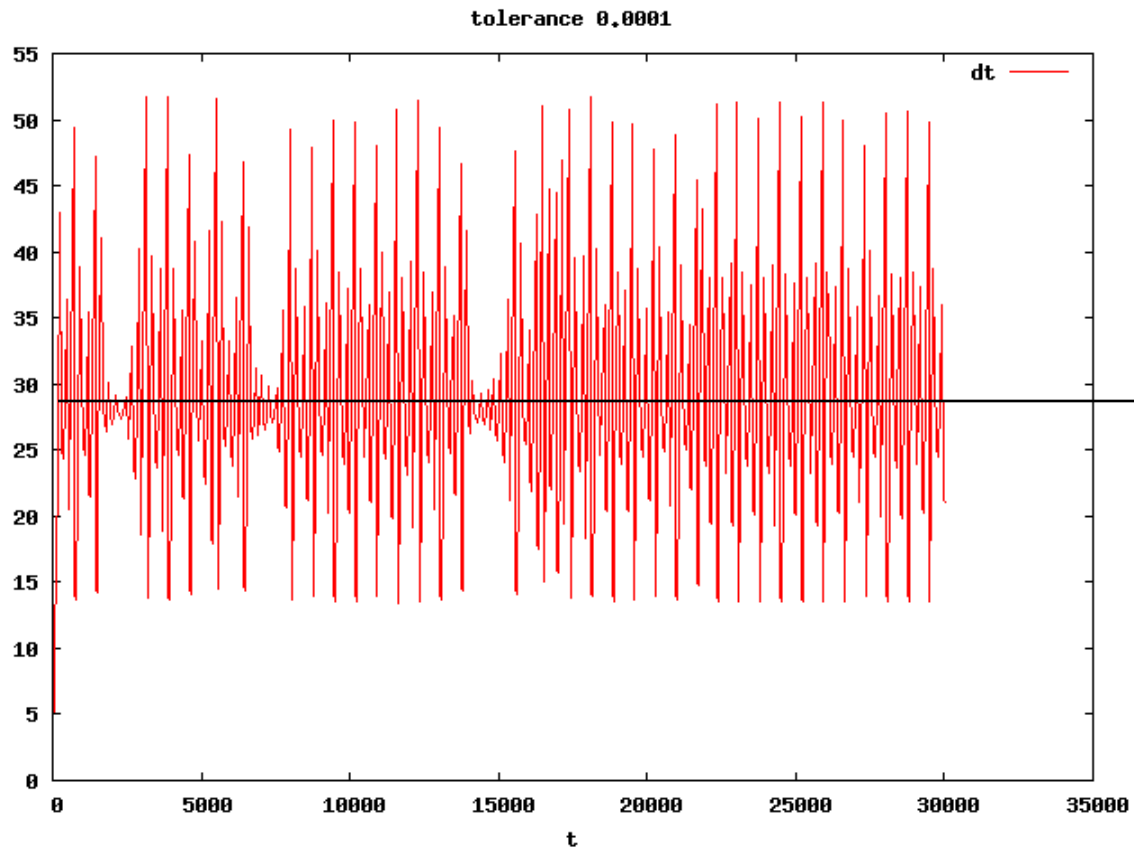


$$\lambda_1 = 1/10$$
$$\lambda_2 = 1/10\ 000$$



w obydwu przypadkach
 Δt tylko chwilowo
przekracza krytyczną
wartość $2/(1/10)=20$

RK4



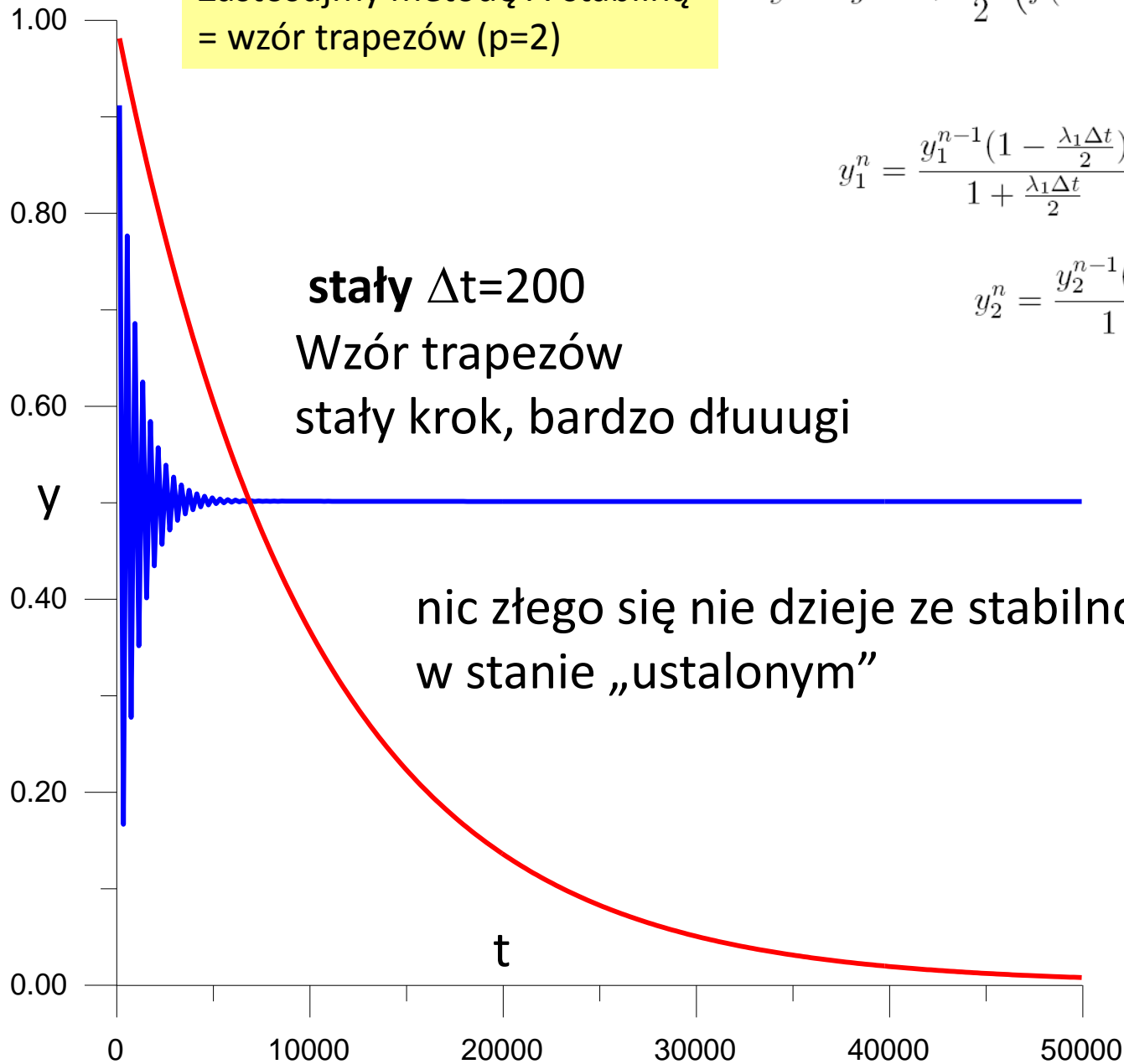
$2.78 / \lambda_1$

Zastosujemy metodę A-stabilną
= wzór trapezów (p=2)

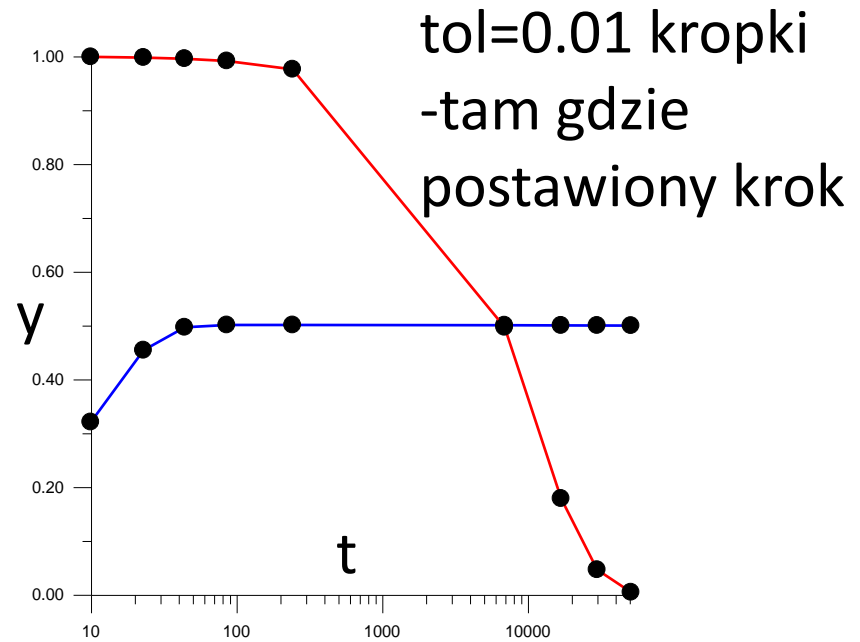
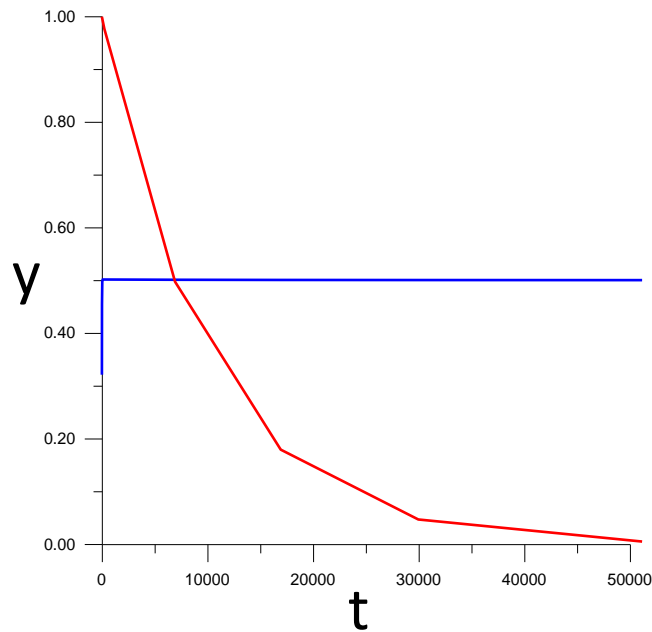
$$y^n = y^{n-1} + \frac{\Delta t}{2} (f(x^{n-1}, y^{n-1}) + f(x^n, y^n)),$$

$$y_1^n = \frac{y_1^{n-1}(1 - \frac{\lambda_1 \Delta t}{2})}{1 + \frac{\lambda_1 \Delta t}{2}} + \frac{\Delta t}{2} \frac{2\lambda_2 y_2^{n-1} + \lambda_1}{1 + \frac{\lambda_1 \Delta t}{2}}$$

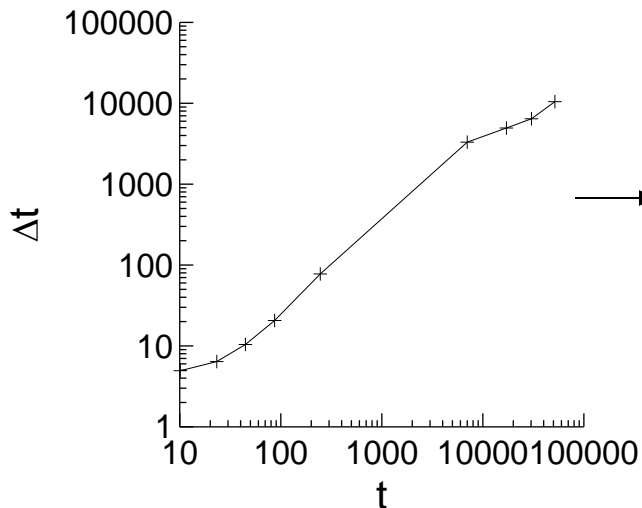
$$y_2^n = \frac{y_2^{n-1}(1 - \frac{\lambda_2 \Delta t}{2})}{1 + \frac{\lambda_2 \Delta t}{2}}.$$



Wzór trapezów i **krok automatycznie dobierany** przez ekstrapolację Richardsona

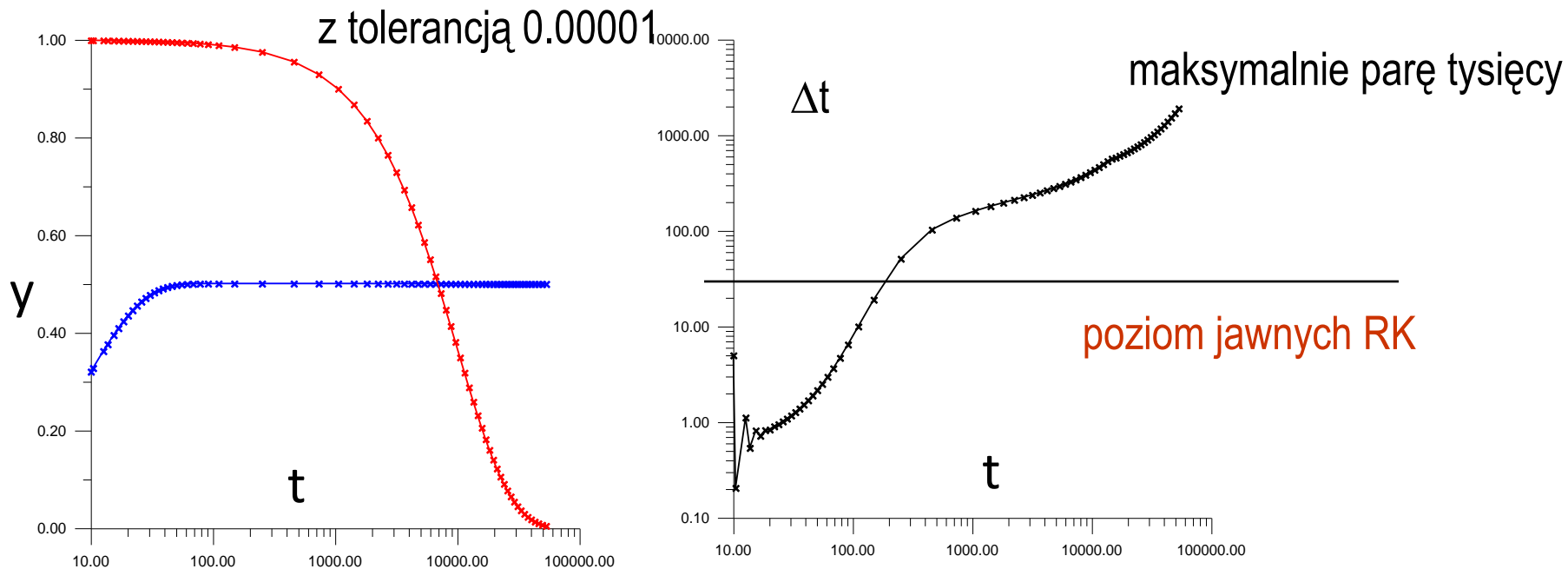


raptem 10 kroków i załatwione!
zamiast 10^4 kroków RK4



Krok czasowy – zmienność 4 rzędów wielkości.

trapezy (najdokładniejsza metoda A-stabilna spośród wielokrokowych)

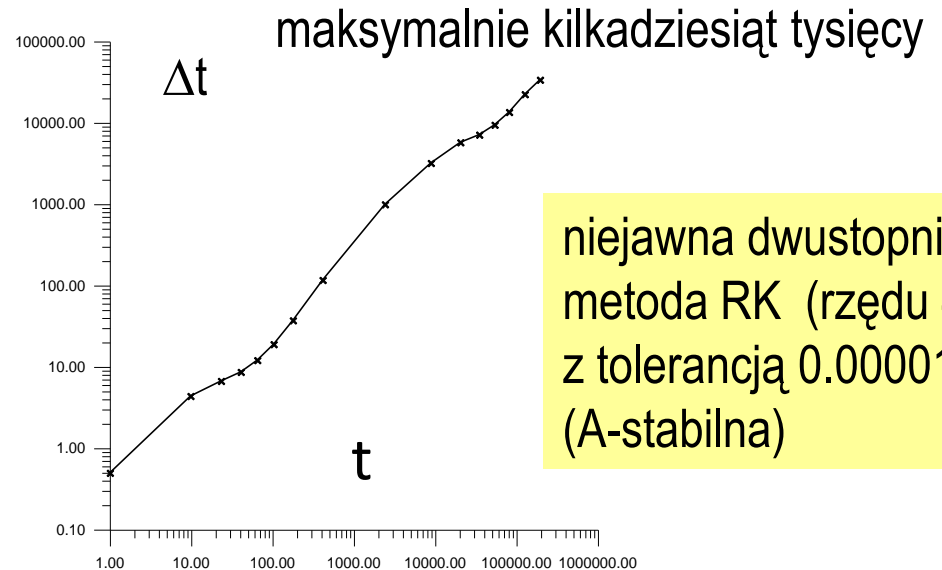
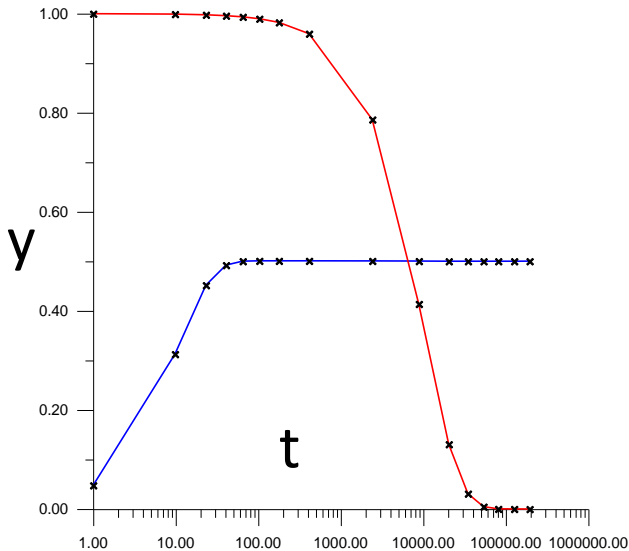
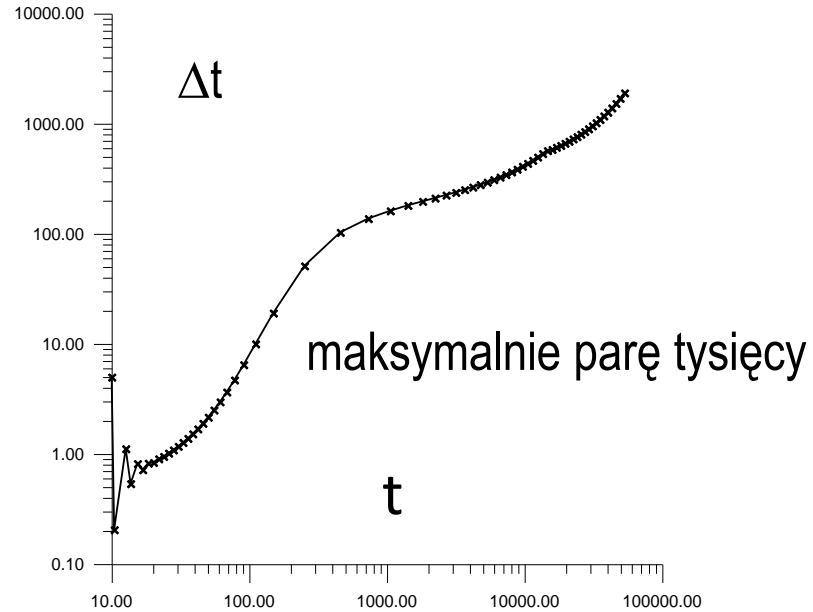
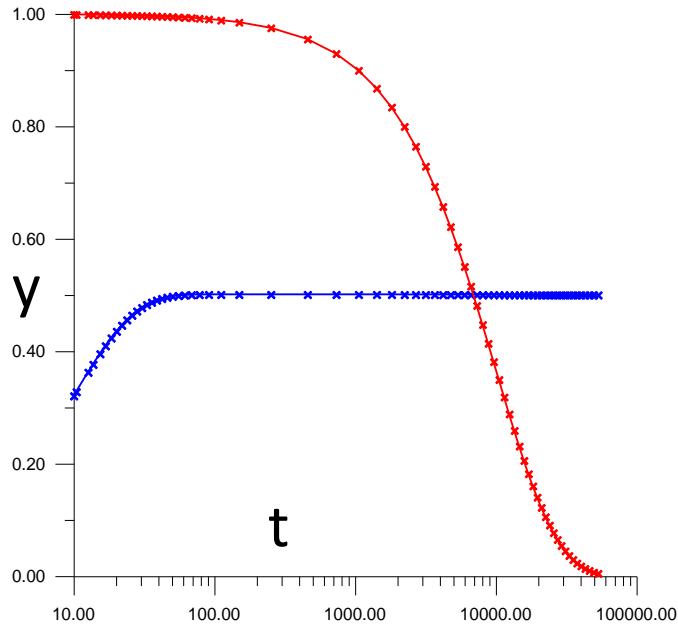


metoda trapezów: jako A-stabilna radzi sobie nieźle z doбором kroku czasowego w problemach sztywnych – ale jest stosunkowo mało dokładna
dokładniejsza A-stabilna pozwoliłaby stawiać jeszcze dłuższe kroki

niestety = dokładniejszej A-stabilnej tej w klasie metod (liniowe wielokrokowe) nie ma

dlatego : niejawne metody RK (jednokrokowe, nieliniowe)

trapezy z tolerancją 0.00001 (najdokładniejsza metoda A-stabilna spośród wielokrokowych)



niejawna dwustopniowa
metoda RK (rzędu 4)
z tolerancją 0.00001
(A-stabilna)

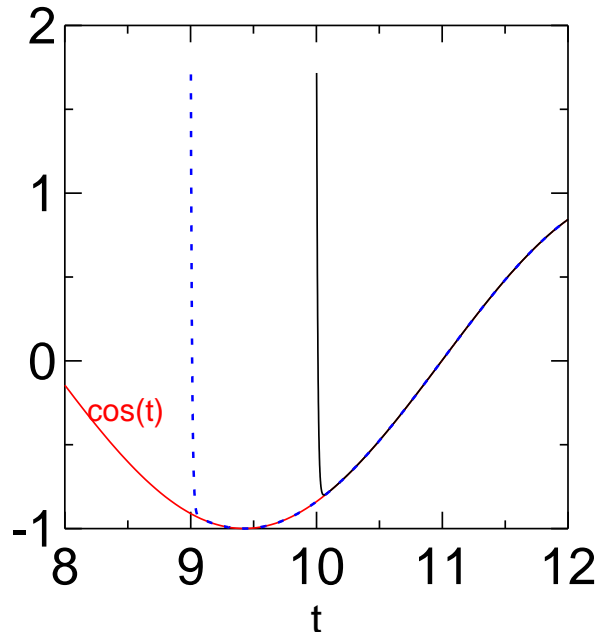
Mówimy, że RRZ jest problemem sztywnym gdy:

1. Problem jest charakteryzowany różnymi skalami czasowymi.
2. Stabilność bwwz nakłada silniejsze ograniczenia na krok czasowy niż dokładność.
3. Metody jawne się nie sprawdzają.

Następny przykład: sztywny problem w pojedynczym równaniu:

$$\frac{du}{dt} = -100(u - \cos(t)) - \sin(t)$$

dla dużych t – rozwiązanie ustalone $u(t) = \cos(t)$



dwie bardzo różne skale czasowe

1) rozwiązania ustalonego okres 2π

2) skala czasowa tłumienia

„odchylenia od stanu ustalonego”

$\exp(-100 t)$ – czasowa stała zaniku 0.01

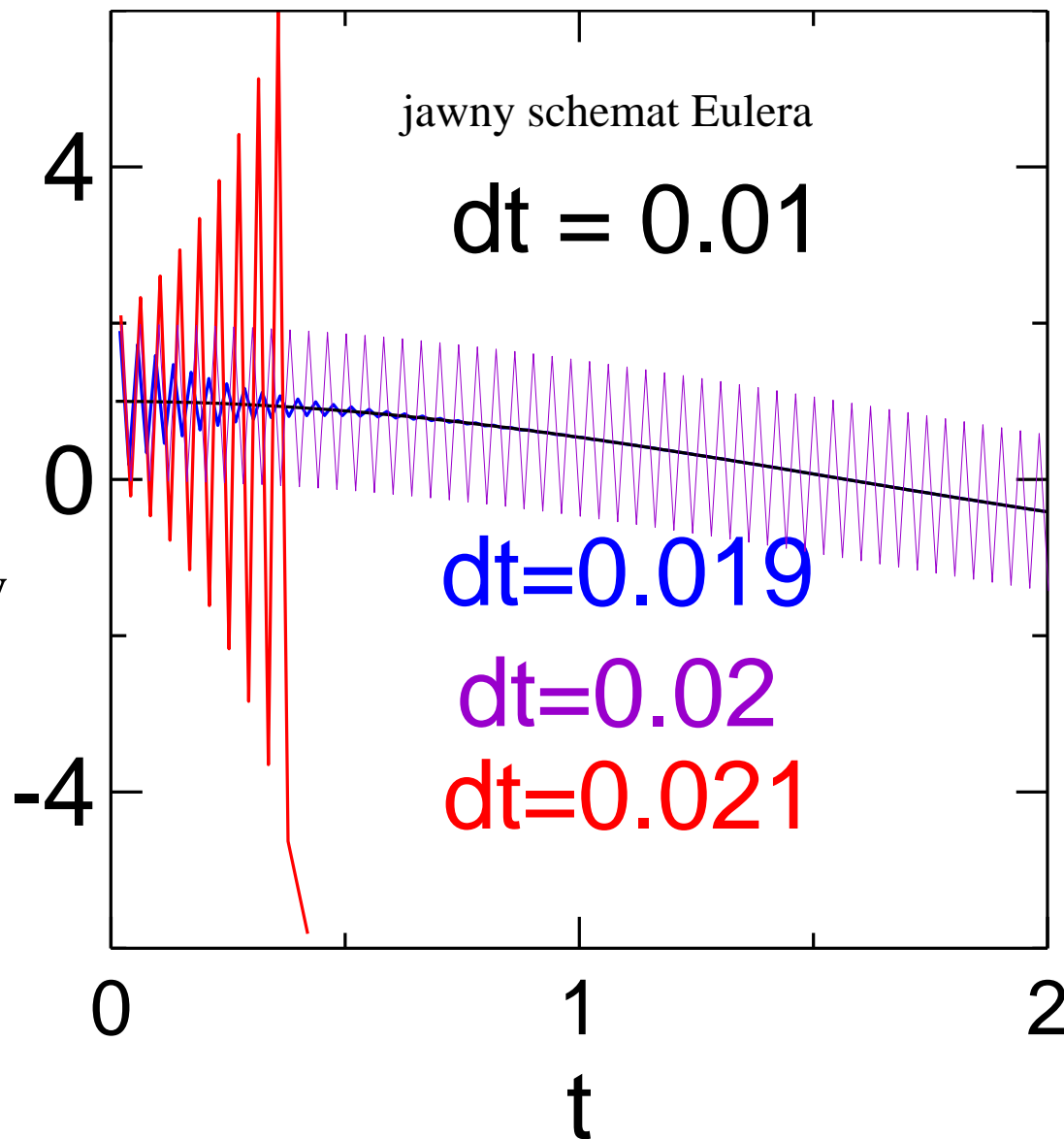
z $u(0)=2$

rozwiązanie: „stacjonarne” $u(t)=\cos(t)$

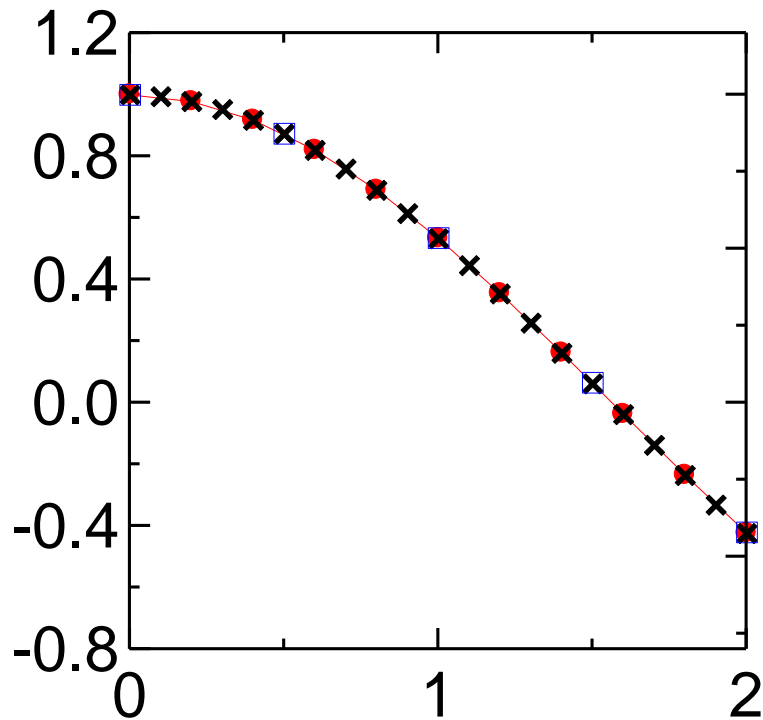
Stały krok
czasowy:

rozpoznajemy
ograniczenie:

$$\Delta t < 2/|100|$$



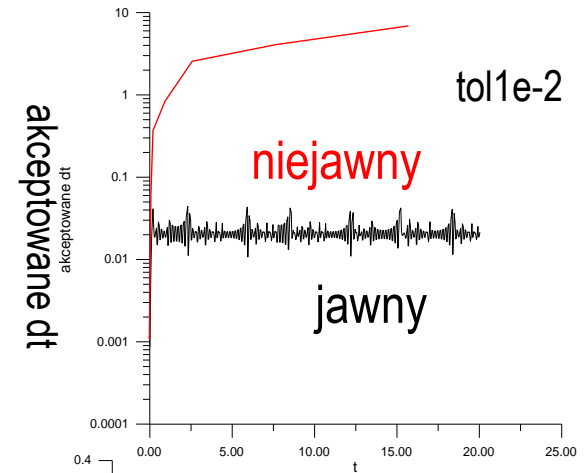
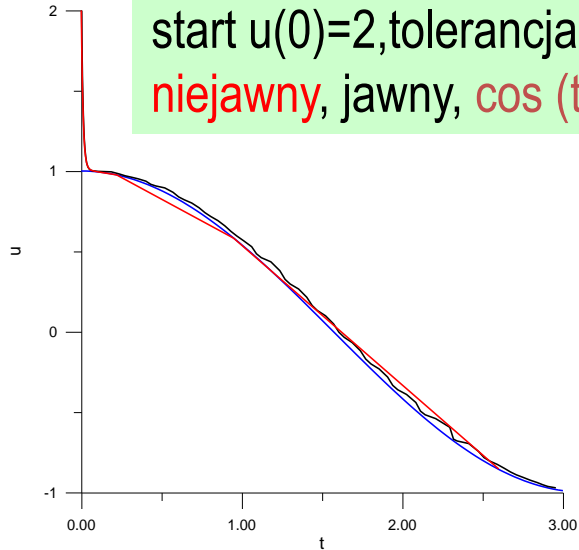
niejawny schemat Eulera – krok stały



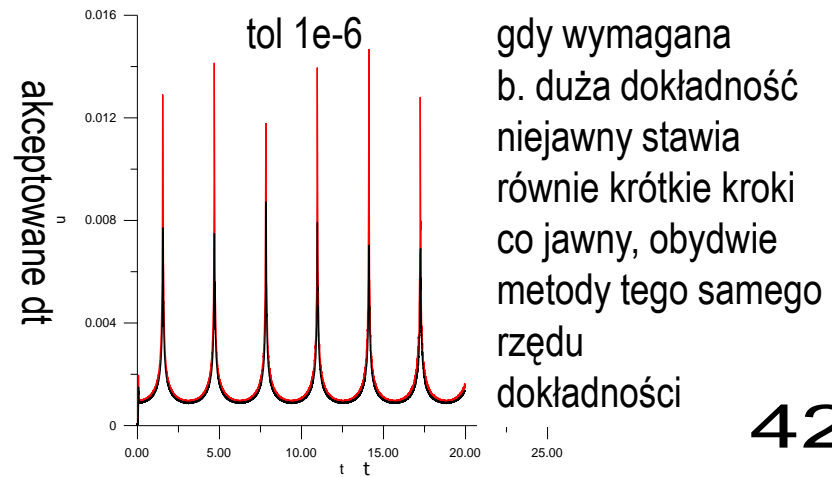
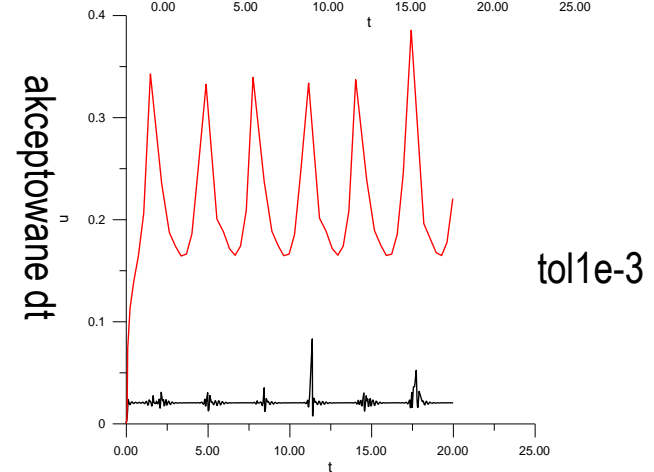
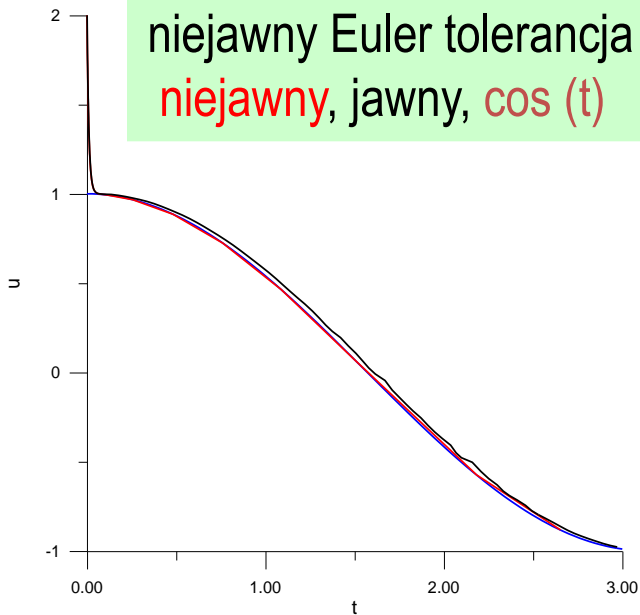
$dt=0.1$
 $dt=0.2$
 $dt=0.5$

tutaj: startowane od warunku $u(0)=1$

wyniki do uzyskania na laboratorium
 start $u(0)=2$, tolerancja $1e-2$
 niejawnym, jawnym, $\cos(t)$



niejawnym Euler tolerancja $1e-3$
 niejawnym, jawnym, $\cos(t)$



gdy wymagana
 b. duża dokładność
 niejawnym stawia
 równie krótkie kroki
 co jawnym, obydwie
 metody tego samego
 rzędu
 dokładności

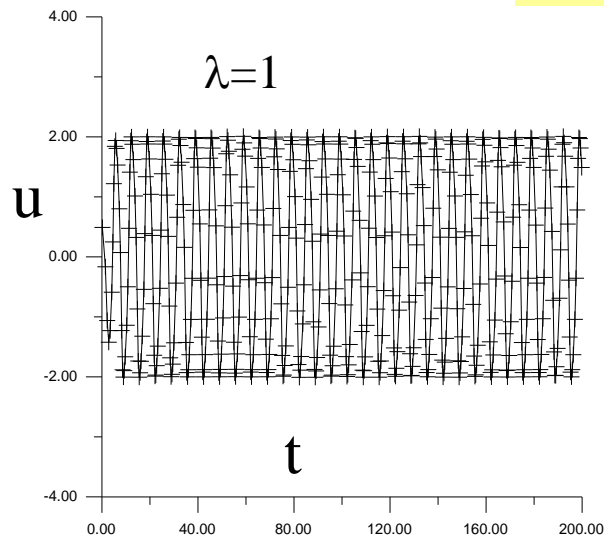
następny przykład: równanie swobodnego oscylatora van der Pola
[historycznie = odkrycie deterministycznego chaosu w lampach firmy Philips
aperiodyczne oscylacje przy periodycznym wymuszeniu]

$$u'' - \lambda(1 - u^2)u' + u = 0 \quad (\lambda=0 = \text{zwykły o. harmoniczny})$$

$$u' = v$$

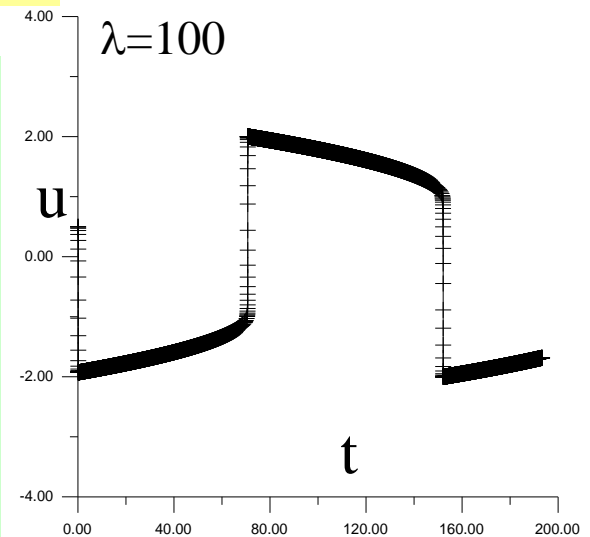
$$v' = \lambda(1 - u^2)v - u$$

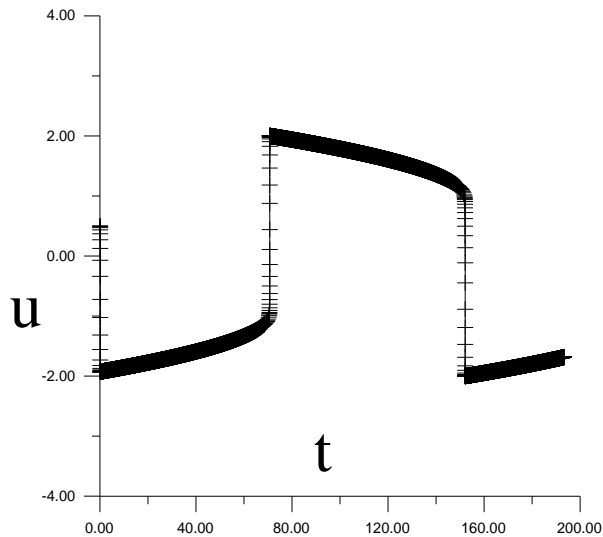
jawny RK4 = zmienny krok czasowy



punkt $u(t)$ policzony = krzyż
po lewej: krzyże położone rozsądnie w porównaniu ze zmiennością rozwiązania

po prawej: problem sztywny
gładkie rozwiązanie
a krzyże się zlewają





równanie: czasem sztywne czasem nie

przydałoby się narzędzie do wykrywania sztywności
np. dla podjęcia decyzji:

tam gdzie sztywność = schemat niejawny

tam gdzie nie = schemat jawny (tańszy)

Detekcja sztywności dla problemu nieliniowego

(dla liniowego = wystarczy rozwiązać problem własny macierzy układu równań)

$$\frac{du}{dt} = f(t, u) \quad \text{układ } N \text{ równań } (u, f\text{-wektory})$$

w chwili t rozwiązanie $u^*(t)$

rozwiązanie chwilę później opisane przez odchylenie $\delta u(t)$ od u^*

$$u(t) = u^*(t) + \delta u(t)$$

linearyzacja: zakładamy, że odchylenie małe, rozwijamy $f(t, u)$ względem u wokół $f(t, u^*)$: [Taylor dla wektora]

$$f(t, u) = B + J(t)u(t) + \dots$$

$$[J(t)]_{ij} = \frac{\partial f^i}{\partial u_j}(t, u^*(t)) \quad \leftarrow \text{ macierz Jakobiego } [N \text{ na } N]$$

$$\frac{du}{dt} = f(t, u)$$

$$f(t, u) = B + J(t)u(t) + \dots$$

problem zlinearyzowany

$$[J(t)]_{ij} = \frac{\partial f^i}{\partial u_j}(t, u^*(t))$$

$$\frac{du}{dt} = Au + B$$

B bez znaczenia dla stabilności

rozwiązać problem własny A : dostaniemy wartości własne λ_i :

Aby rachunek się powiódł: $\Delta t \lambda_i$ musi leżeć w regionie stabilności używanej metody dla wszystkich i .

Jeśli duża rozpiętość λ : problem będzie sztywny.

Przykład: nieliniowy układ równań z warunkowo występującą sztywnością

$$\begin{pmatrix} \frac{du^{(1)}}{dt} \\ \frac{du^{(2)}}{dt} \end{pmatrix} = \begin{pmatrix} -u^{(1)}u^{(2)} \\ \cos(u^{(1)}) - \exp(u^{(2)}) \end{pmatrix}$$

$$[J(t)]_{ij} = \frac{\partial f^i}{\partial u_j}(t, u^*(t))$$

$$J = \begin{pmatrix} -u^{(2)} & -u^{(1)} \\ -\sin(u^{(1)}) & -\exp(u^{(2)}) \end{pmatrix}$$

jeśli druga składowa \mathbf{u} urośnie – macierz prawie diagonalna
z szerokim zakresem wartości własnych - sztywność

Przykład detekcja sztywności dla: oscylatora van der Pola

$$u' = v$$

$$v' = \lambda(1 - u^2)v - u$$

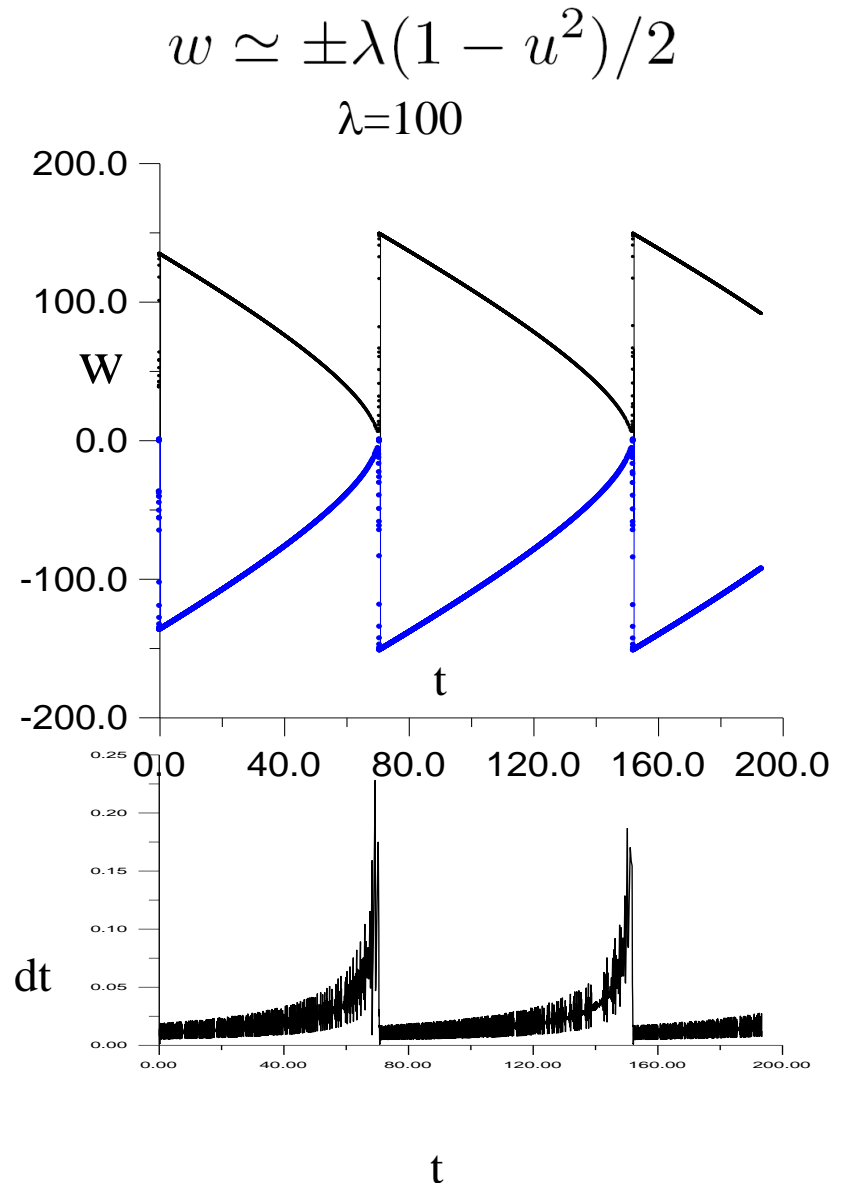
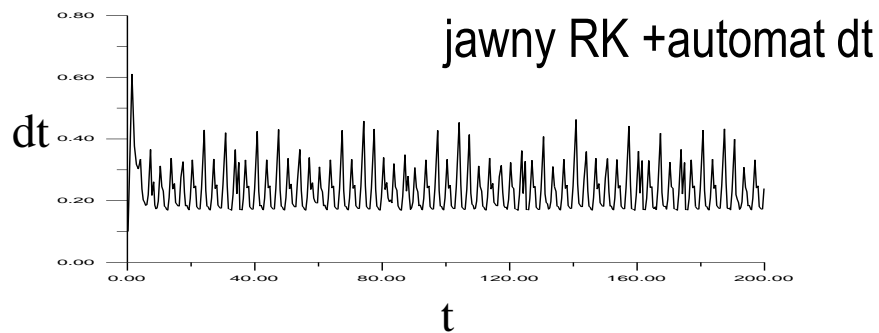
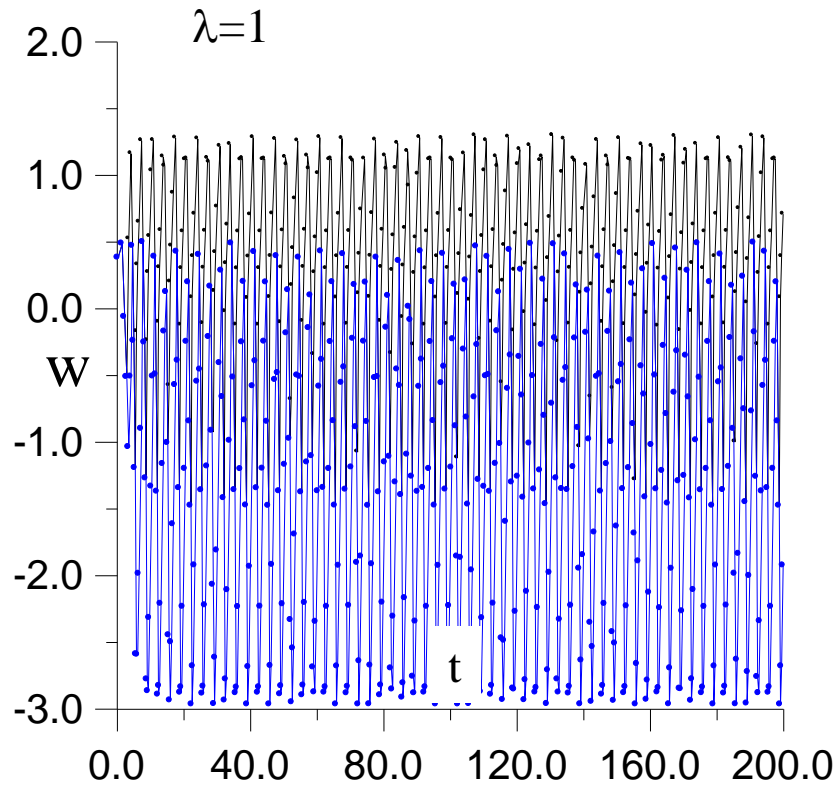
$$J = \begin{pmatrix} 0 & 1 \\ -2\lambda v u - 1 & \lambda(1 - u^2) \end{pmatrix}$$

wartości własne:

$$w(t) = \frac{1}{2} (1 - u(t)^2) \pm \frac{1}{2} \sqrt{(1 - u(t)^2)^2 - 8\lambda v(t)u(t) - 4}$$

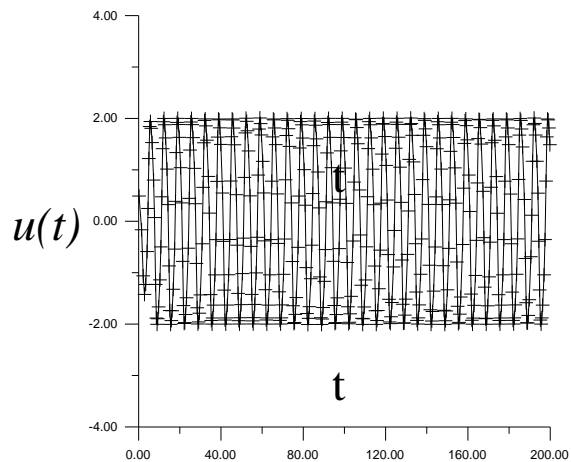
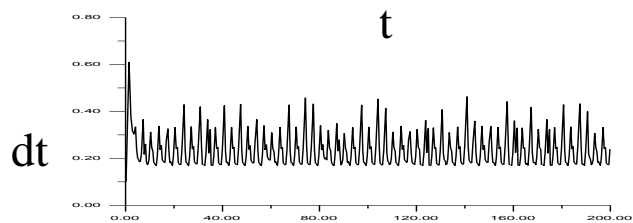
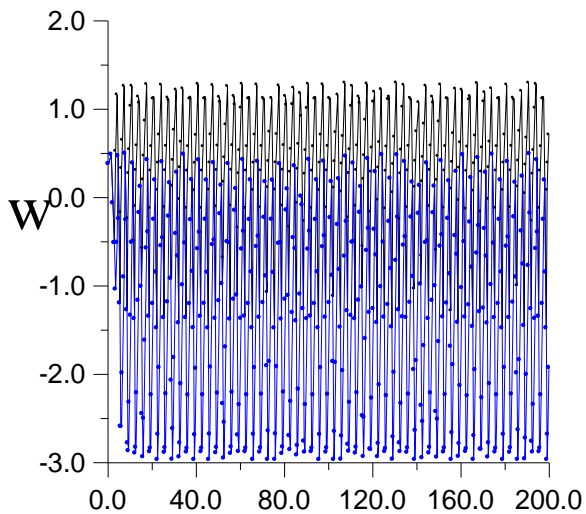
$$w = \frac{1}{2} (1 - u^2) \pm \frac{1}{2} \sqrt{\lambda^2(1 - u^2)^2 - 8\lambda vu - 4}$$

niebieskie i czarne: części rzeczywiste wartości własnych



$\lambda=1$

jawny RK + automat dt



$$w \simeq \pm \lambda(1 - u^2)/2$$

 $\lambda=100$ 