

## Równania różniczkowe i komputery

Jerzy Karczmarczuk

Zakład Informatyki, Uniwersytet w Caen, Francja

### 1. Wstęp

Niniejszy tekst stanowi kolejny rozdział z cyklu: jak efektywnie wykorzystywać komputery w nauczaniu fizyki. Przebiegi czasowe w fizyce klasycznej: trajektorie obiektów mechanicznych (np. oscylacje, ruchy planet, żyroskopy itp.), dynamika wzbudzeń obwodów elektrycznych, dążenie do równowagi (albo nie) układów termodynamicznych itp. – wszystko to wymaga rozwiązywania równań różniczkowych.

W fizyce ogólnej sporo czasu poświęca się prostym równaniom, których rozwiązania analityczne są znane, ale większość interesujących przypadków wychodzi poza ramy szkolne i nawet jeśli student (czy nauczyciel) jest w stanie sformułować równanie, jego rozwiązanie wymaga zastosowania metod numerycznych. W przypadku programów symulacyjnych stosowanie metod numerycznych jest regułą. Szczególnie dotyczy to zagadnień nieliniowych.

Zbyt często korzysta się z gotowych pakietów lub bibliotek procedur rozwiązujących równania, argumenty za tym są oczywiste: w nauczaniu fizyki należy zajmować się fizyką, a nie technikami rachunkowymi. Procedury biblioteczne są (miejmy nadzieję...) napisane przez specjalistów, i są dobre (tj. szybkie, dokładne itp.), więc po co tracić czas na własne? Nie jest to nasza specjalność... Dlaczego więc „zbyt często”? Po prostu dlatego, gdyż *w praktyce pedagogicznej rzadko oplaca się robić coś na ślepo.*

1. Na ogół z góry nie wiemy, jaka dokładność, a także stosunek dokładności do prędkości obliczeń są nam potrzebne. Użycie gotowych procedur może stanowić marnotrawstwo lub dostarczyć źle kontrolowanych błędów.
2. Opanowanie bibliotek numerycznych, zrozumienie jak należy stosować tę, czy inną procedurę, jak ocenić jej jakość, może być bardziej skomplikowane, niż zaprogramowanie samemu czegoś prostego, a wystarczającego. I tu również możemy zmarnować sporo czasu, a niczego się nie nauczymy...
3. W wielu szczególnych przypadkach, znajomość fizyki zagadnienia pozwala na przeformułowanie równania tak, że rozwiązanie stanie się prostsze i dokładniejsze. Nawet jeśli ogólnie wstawienie np. zachowania energii do równania jest trudne, *kontrola* tego prawa zachowania i ewentualne poprawki mogą być pożyteczne, a czasami niezbędne. Gotowe pakiety nie mają tej „mądrości”.
4. Właśnie dlatego, iż w centrum naszych zainteresowań winna być fizyka, a nie metody numeryczne, winniśmy zrozumieć jakie artefakty wprowadza-

my do naszych programów, w przeciwnym wypadku nie możemy mieć zaufania do naszych obliczeń, symulacji itp., i możemy ogóle traktować komputery jedynie jako zabawkę do wizualizacji (tak funkcjonuje, niestety, wiele pakietów do symulacji dostępnych w internecie...).

Metody przybliżone wymagają szczegółowej analizy jakości rozwiązania. Jak dokładność zależy od długości kroku czasowego? Czy gdy zmierza ona do zera, metoda jest zbieżna do prawdziwej wartości, czy pojawiają się kłopoty? Jak błędy metody rozwijają się w czasie? Dla – typowych dla fizyki – problemów nieliniowych, nie ma prostych metod szacowania jakości rozwiązania... Ten ostatni problem *stabilności rozwiązań* numerycznych jest uniwersalny i wszyscy korzystający z metod komputerowych powinni o nim mieć pojęcie. Czasami pomaga „zanurzenie” rozwiązywanego problemu w szerszym zagadnieniu, którego własności są znane.

Główną motywacją do napisania tego artykułiku było spostrzeżenie, że moi koledzy wykładający fizykę na pierwszych latach studiów często ograniczają się do schematycznego przedstawienia najprostszej metody Eulera i to wszystko. Wynik jest taki, że jedyną metodą znaną przez studentów jest najgorszy możliwy algorytm, którego nie należy używać! Oczywiście przedyskutujemy jedynie metody intuicyjnie, proste i łatwe do zaprogramowania.

## 2. Dyskretyzacja czasu i metody Eulera

Równanie od którego zaczniemy dyskusję to równanie pierwszego rzędu:

$$\dot{x} = f(x, t), \quad (2.1)$$

gdzie  $x(t)$  jest zmienną zależną od czasu, na ogół wielowymiarową. (Przyjmijmy, że kropka oznacza różniczkowanie po czasie:  $\dot{x} = \frac{dx}{dt}$ ). Równania ruchu i wiele innych równań w fizyce są drugiego rzędu, równanie Newtona to  $m\ddot{x} = F(x, t)$ , gdzie  $F$  jest wyrażeniem określającym siłę, zależną od położenia, np. równanie oscylatora harmonicznego, to  $m\ddot{x} = -kx$ . Jednak łatwo to równanie przedstawić w postaci pierwszego rzędu, tylko wektorowej:

$$\frac{d}{dt} \begin{pmatrix} x \\ v \end{pmatrix} = \begin{pmatrix} v \\ F(x, t)/m \end{pmatrix}, \quad (2.2)$$

gdzie  $v$  jest prędkością. Tak więc, w dyskusji podstawowych własności równań, posłużymy się ogólną formą (2.1), niekoniecznie precyzując czy  $x$  jest skalarem, czy wektorem.

Techniki numeryczne wymagają tutaj dyskretyzacji czasu, będziemy operować ciągami  $(t_0, t_1, \dots, t_n, \dots)$ , które generują trajektorie  $(x_0, x_1, \dots)$ , i  $(v_0, v_1, \dots)$ , gdzie  $x_n = x(t_n)$ . Odstępów czasowych  $\Delta t$  między  $t_n$  a  $t_{n+1}$  mogą być zmienne, ale tutaj potraktujemy je jako jedną stałą, zasadniczą dla całej

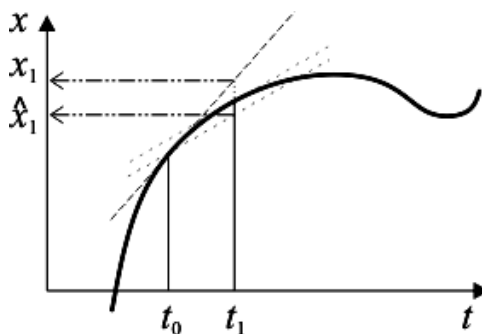
techniki. Pochodną czasową możemy więc w przybliżeniu zastąpić ilorazem różnicowym:  $\dot{x} \cong (x_{n+1} - x_n)/\Delta t$ , gdy  $\Delta t$  jest małe, błąd przybliżenia będzie mały. Tu jednak uwidacznia się **podstawowy** problem numerycznych technik rozwiązywania: dla skończonego, określonego  $t$ , jeśli  $\Delta t$  będzie się zmniejszać, liczba kroków rozwiązania wzrośnie, wzrośnie więc czas procesu, co zmniejszy opłacalność metody. W dodatku, jeśli nawet bardzo małe błędy się akumulują, zmniejszanie  $\Delta t$  może być przejściem od Scylli do Charybdy, po pewnej liczbie kroków rozwiązanie może stracić sens. Wzór na iloraz różnicowy jest niestabilny numerycznie, gdy  $\Delta t$  dąży do zera: odejmuje się od siebie zbliżone liczby, wynik zależy od „dalszych miejsc po przecinku”, więc względny błąd może być spory, na skutek błędów zaokrągleń. Różnicę dzieli się przez bardzo małą liczbę, więc ostateczny wynik może być bardzo zły... Zachowanie umiaru jest tutaj krytyczne. Wreszcie, iloraz różnicowy „w przód” nie jest jedynym możliwym, wyrażenie „w tył”,  $\dot{x} \cong (x_n - x_{n-1})/\Delta t$  jest również możliwe, jak i formuły pośrednie (np. symetryczna przód-tył,  $\dot{x} \cong (x_{n+1} - x_{n-1})/2\Delta t$ ).

W tekście poniżej spotkamy się z wyrażeniami dostosowanymi do równań ruchu, w których oddzieliliśmy położenia  $x$  i prędkości  $v$ , ale także z technikami uniwersalnymi, gdzie występuje jedynie  $x$ . Wtedy należy pamiętać o wektoryzacji, o podwojeniu liczby zmiennych.

### 2.1. Podstawowa metoda Eulera

Jeśli zastąpimy we wzorze (2.1) pochodną przez iloraz różnicowy w przód, otrzymamy wzór:  $x_{n+1} = x_n + \Delta t \cdot f(x, t)$ . Jest to „prawie algorytm”, tylko nie wiemy co to jest  $x$  po prawej stronie formuły, nie wiemy także jaką wartość przyjąć dla  $t$ . Ponieważ punktem wyjścia rozwiązania są nasze warunki początkowe, znamy  $x_0 = x(t_0)$ , aby móc posunąć się naprzód, wykorzystujemy znane dane. Podstawowy algorytm ekstrapolacyjny Eulera, to

$$x_{n+1} = x_n + \Delta t \cdot f(x_n, t_n). \quad (2.3)$$



Rys. 1. Metoda Eulera

Często w kursach fizyki wspominając o metodach rozwiązywania równań, ogranicza się do tego algorytmu. W interesujących przypadkach ten algorytm jest niestabilny. Bywa jednak pomocny jako element innych algorytmów, doskonalszych.

Popatrzmy na rysunek 1. Chcemy ekstrapolować szukaną krzywą  $x(t)$  od  $x(t_0)$  do  $x(t_1)$ . Linia kreskowana styczna do krzywej w  $t_0$  (najbardziej stroma) określa wartość  $x = x_1$ , górną. (Pozostałe kreski posłużą nam za chwilę). Ponieważ krzywa odchyła się w dół, wartość numeryczna jest nadmiarowa.

Ponieważ w fizyce operujemy równaniami ruchu, więc musimy operować wielkościami wektorowymi; trzeba będzie przejść od wartości początkowych  $(x_0; v_0)$  do  $(x_1 = x_0 + v_0 \Delta t; v_1 = v_0 + a_0 \Delta)$ , gdzie  $a$  oznacza przyspieszenie, siłę podzieloną przez masę. Ta wartość zależy od położenia  $x$ , rzadziej od prędkości (gdy np. mamy tarcie, lub pola magnetyczne działające na ładunki elektryczne). Metoda Eulera nieco „uszkodzona” (warianty metody Eulera-Cromera):  $(v_{n+1} = v_n + a_n \Delta t; x_{n+1} = x_n + v_{n+1} \Delta t)$ , lub  $(x_{n+1} = x_n + v_n \Delta t; v_{n+1} = v_n + a_{n+1} \Delta t)$ , gdzie  $a_n = a(x_n)$  dają często numerycznie lepsze wyniki, zwłaszcza gdy ruch jest globalnie periodyczny lub prawie periodyczny. Nie jest to żaden „przypadek”, powyższe techniki należą do grupy tzw. integratorów symplektycznych, które znacznie lepiej zachowują energię niż zwykle metody ekstrapolacyjne. Do nich należy także metoda Verleta omówiona poniżej.

## 2.2. Metoda Eulera niejawna

Zastępując w prawej stronie wzoru iteracyjnego  $x$  przez  $x_n$  dokonaliśmy najłatwiejszego wyboru, ale teoretycznie moglibyśmy określić także

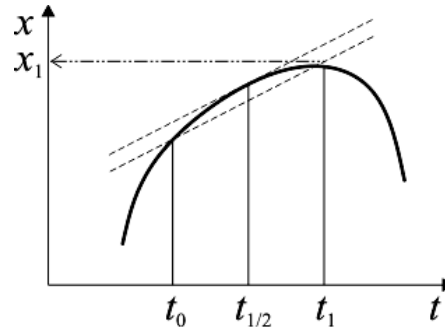
$$x_{n+1} = x_n + \Delta t \cdot f(x_{n+1}, t_{n+1}), \quad (2.4)$$

co jest równoważne wzięciu ilorazu różnicowego „w tył”, lub wziąć jakieś inne wartości  $(t, x)$  z przedziału między indeksami  $n$  i  $(n + 1)$ . Ale jak to zaprogramować? Wzór (2.4) wyraża  $x_{n+1}$  poprzez tę samą wartość. Jest to jednak po prostu równanie algebraiczne (nie różniczkowe) na  $x_{n+1}$  i możemy je rozwiązać różnymi metodami, zależnie od funkcji  $f$ . Nie będziemy opisywać tutaj wymyślnych metod algebraicznych, mimo iż są czasami niezbędne, zauważymy tylko, że jeśli udałoby nam się to równanie rozwiązać – co również zaproponował Euler – wtedy dostalibyśmy dolną wartość  $\hat{x}_1$ . Kreska styczna do krzywej w  $t_1$  została przeniesiona równoległe do punktu wyjścia.

Błąd poszedł w przeciwnym kierunku. Dla naszej funkcji otrzymaliśmy wartość z niedomiarem. Gdyby „skombinować” te dwie metody, podstawową i niejawną (pośrednią), otrzymany błąd może być *znacznie* mniejszy. Co więcej, możemy uniknąć destabilizacji, tj. wybuchowego narastania błędu w kolejnych krokach (w jednym kierunku lub rosnących oscylacji wokół prawdziwej trajektorii). W następnym rozdziale omówimy te metody kombinowane, ale wypada omówić najprostszą metodę Eulera, która jest stosowana praktycznie.

### 2.3. Metoda Eulera centrowana (schemat punktu środkowego)

Twierdzenie o wartości średniej w analizie mówi, że między  $t_n$  a  $t_{n+1}$  istnieje taki punkt, że styczna do krzywej będzie równoległa do siecznej między  $x_n$  a  $x_{n+1}$ . Nie znamy go, ale dla dość regularnych krzywych wzięcie punktu środkowego,  $t_{1/2} = 0.5(t_0 + t_1)$ , daje niezły wynik. Jest to równoważne przybliżeniu pochodnej przez iloraz różnicowy symetryczny:  $\dot{x} \cong (x_{n+1} - x_{n-1})/2\Delta t$  (i przepołowieniu kroku, tj. użyciu indeksów połówkowych).



Rys. 2. Metoda punktu środkowego

Metoda oparta o to spostrzeżenie, również zaproponowana przez Eulera, okazuje się przydatna do modelowania prostych ruchów periodycznych, do oscylatora, problemu dwóch ciał grawitujących itp. W prawej stronie wzoru iteracyjnego zastępujemy nieznaną  $x$  wartością „połówkową”, i także użyjemy połówkowej wartości dla kroku czasowego (co już jest uwidocznione wyżej):  $t_{n+1/2} = t_n + \Delta t/2$ .

$$x_{n+1} = x_n + \Delta t \cdot f(x_{n+1/2}, t_{n+1/2}). \quad (2.5)$$

Następnie, dysponując  $x_{n+1/2}$  oraz  $x_{n+1}$ , obliczamy  $x_{n+3/2}$  itd. Ta metoda nie „wystartuje” sama, gdyż standardowe warunki początkowe dostarczają tylko  $x_0$ , a pierwszą obliczaną wartością jest  $x_1$ . Musimy otrzymać  $x_{1/2}$  skądinąd, np. używając metody Eulera ekstrapolacyjnej albo lepszej. Naprawdę, *musi* to być lepsza metoda; początkowy błąd winien być bardzo mały, znacznie mniejszy od samego błędu metody, w przeciwnym wypadku metoda „się rozłązi”. Można więc zapytać o sens stosowania metody centrowanej, skoro wymaga ona innej i to dobrej? Odpowiedź jest oczywista: metoda centrowana jest prosta, a więc szybka (przy określonej liczbie kroków; o ile jej dokładność nam wystarczy. Metoda pomocnicza jest stosowana tylko raz).

Trzeba jednak powiedzieć, że przy danym kroku jest ona mniej dokładna od np. metody Verleta. Jak zawsze, wybór między większą dokładnością a szybkością obliczeń zależy od problemu i od wymagań programisty.

### 3. Problem stabilności

Podstawowym, kluczowym problemem w stosowaniu iteracyjnych metod numerycznych, w których przybliżone wyrażenie wykorzystuje się w wielu następnych krokach, jest charakter narastania błędów w kolejnych krokach. Dla równań różniczkowych jest to często ważniejsze niż sam błąd pojedynczego kroku. Aby oszacować zachowanie się błędu zastosujemy metodę „wirtual-

nych” zmian obliczanej wartości; przypuścimy, że program w pewnym momencie zamiast  $x$  oblicza  $x + \delta x$  (albo, dla uproszczenia notacji, po prostu  $\delta$  zamiast  $\delta x$ ). *Założenia*  $\delta$  jest uważane za małe.

Czytelnik niezainteresowany tzw. „kuchnią” metod numerycznych może opuścić resztę tego rozdziału przy pierwszym czytaniu, ale winien zdać sobie sprawę jakie kłopoty mogą go spotkać, jeśli zlekceważy pewne kryteria przedyskutowane poniżej.

Przypuszczamy, że w pewnym kroku obliczeń już mamy pewną niedokładność, nasz program obliczył  $x_n + \delta_n$ . Problem stabilności sprowadzi się do odpowiedzi na pytanie, jaka jest relacja między  $\delta_n$ , a wartością  $\delta_{n+1}$ , która dotyczy następnego kroku, w którym obliczymy  $x_{n+1} + \delta_{n+1}$ .

Jeżeli wirtualna niedokładność jest mała, możemy użyć rachunku różniczkowego do oszacowania jej zmian. Teraz wszystko będzie zależeć od użytego algorytmu! Dla podstawowej metody Eulera dostaniemy

$$x_{n+1} + \delta_{n+1} = x_n + \delta_n + \Delta t \cdot f(x_n + \delta_n, t_n). \quad (3.1)$$

Ale  $f(x_n + \delta_n, t_n)$  można rozwinąć i w pierwszym rzędzie przybliżenia różniczkowego względem  $\delta$ , dostaniemy

$$x_{n+1} + \delta_{n+1} = x_n + \delta_n + \Delta t \cdot (f(x_n, t_n) + \delta_n \cdot f'(x_n, t_n)), \quad (3.2)$$

co można uprościć. Odejmując równanie (2.3) od (3.2) otrzymamy:

$$\delta_{n+1} = \delta_n \cdot (1 + \Delta t \cdot f'(x_n, t_n)), \quad (3.3)$$

gdzie pochodna  $f'$  jest wzięta po  $x$ . Widzimy, że aby stabilność nie została pogwałcona,  $f'$  musi być ujemne i nie za duże co do wartości bezwzględnej. Jeśli  $f' < -2/\Delta t$ , metoda się rozbiegnie na skutek wzrastających oscylacji. Tak więc w przypadku równania liniowego:  $\dot{x} = A x$ , aby nie pogwałcić stabilności algorytmu,  $A$  musi być ujemne,  $x$  musi spadać do zera. Dokładnym rozwiązaniem jest, oczywiście,  $x = x_0 e^{At}$ , dla dowolnej wartości  $x_0$ , dodatniej lub ujemnej. Jeśli  $x$  będzie rosła (co do wartości bezwzględnej), metoda się natychmiast destabilizuje. Problem jest jednak bardziej skomplikowany, gdyż skoro (z  $A$  dodatnim) wzrastają również wartości  $x_n$ , więc błąd *względny* może być akceptowalny... Na ogół nie należy na to liczyć.

W podobny sposób możemy oszacować stabilność dla metody niejawnej, np. dla równania liniowego  $\dot{x} = Ax$ , otrzymamy  $\delta_{n+1} = \delta_n / (1 - A\Delta t)$ , a także dla metody centrowanej i metod przedstawionych w następnym rozdziale. Jest to jednak trudniejsze, a w ogólnym przypadku „ręcznie” niewiele da się powiedzieć. Wtedy trzeba zaprogramować rozwiązanie stowarzyszając algorytm z obliczaniem równoległym „odchylenia wirtualnego” i monitorowaniem stabilności. W przypadku, gdy grozi rozbiegnięcie się algorytmu, czyli jego wynik nie zmierza do prawdziwej wartości rozwiązania, należy zmienić krok albo przejść do innego algorytmu.

Interesującym testem jest sprawdzenie stabilności metody Eulera dla ruchu periodycznego, np. dla oscylatora harmonicznego. Wprowadzając sztuczną zmienną zespoloną  $z_n = x_n + i v_n$ , dostaniemy równanie liniowe pierwszego rzędu:  $\dot{z} = A z$ , ze współczynnikiem  $A$  czysto urojonym. Ponieważ wartość bezwzględna wyrażenia  $|1 + A\Delta t|$  jest wtedy na pewno większa od 1, metoda Eulera dla oscylatora jest bezwzględnie niestabilna dla dowolnych parametrów i dowolnego  $\Delta t$ . Ponieważ każde równanie o zbliżonym charakterze (trajektorie periodyczne) lokalnie linearyzuje się do równania oscylatora, *wszelkie* ruchy periodyczne, np. problem Keplera, będą miały tę własność.

#### 4. Metoda Heuna i metoda Rungego-Kutty (2)

Wróćmy do metody Eulera pośredniej (2.4) i do naszych rozważań o celowości skombinowania jej z metodą wprost (2.3). Otóż metoda bezpośrednia, podstawowa, pozwoli nam w przybliżeniu „rozwiązać” równanie określające nieznaną wartość po prawej stronie wzoru. Wynik potraktujemy jako pewną wartość pomocniczą. Definiujemy:

$$k = \Delta t \cdot f(x_n, t_n), \quad \text{i} \quad \tilde{x}_{n+1} = x_n + k. \quad (4.1)$$

To oczywiście jest jeszcze dalekie od  $x_{n+1}$ , ale pozwoli obliczyć  $f$ . Następnie obliczamy średnią:

$$x_{n+1} = x_n + \frac{1}{2} (k + \Delta t \cdot f(\tilde{x}_{n+1}, t_{n+1})). \quad (4.2)$$

W przypadku zależności czasowych,  $t_{n+1} = t_n + \Delta t$ , jak zwykle. Algorytm ten, zwany metodą Heuna był dość popularny (wśród najprostszych), ale konkuruje on ze zbliżonym algorytmem Rungego-Kutty, gdzie średnią oblicza się inaczej. Oto inny wariant powyższego rozumowania, stanowiący bazę metody RK2, a także metod Rungego-Kutty wyższych rzędów, z których poznamy jedną: RK4, najbardziej popularną i w ogóle jedną z najbardziej popularnych wśród wszystkich klasycznych. W metodzie RK2 zaczniemy od przybliżenia iteracyjnego następnej wartości *połówkowej* poprzez metodę Eulera:

$$t_{n+1/2} = t_n + \frac{\Delta t}{2}, \quad \tilde{x}_{n+1/2} = x_n + \frac{\Delta t}{2} \cdot f(x_n, t_n), \quad (4.3)$$

a następnie stosujemy po prostu metodę kroku połówkowego (centrowaną), z tą właśnie przybliżoną wartością:

$$x_{n+1} = x_n + \Delta t \cdot f(\tilde{x}_{n+1/2}, t_{n+1/2}). \quad (4.4)$$

Omówione tutaj metody noszą miano metod rzędu drugiego, błąd w każdym kroku jest proporcjonalny do  $\Delta t^3$  (w metodzie Eulera ten błąd jest rzędu  $\Delta t^2$ ). Którą z nich, Heuna czy RK2 użyć w praktyce? Są one zbliżone zarówno szybkością jak i dokładnością, jest to głównie kwestia stylu i przyzwyczajenia, oraz paru innych zaawansowanych kryteriów (możliwości „ulepszania” i stowarzy-

szenia z nią dodatkowych obliczeń wspomagających), których nie będziemy tutaj omawiać.

Zajmiemy się teraz kilkoma metodami wyższych rzędów, prostymi i zrozumiałymi, ważnymi dla tych, dla których prędkość obliczeń jest nieco mniej istotna niż ich dokładność.

## 5. Metoda Verleta

Jest to jedna z ulubionych metod fizyków piszących proste programy symulacyjne, gdyż jest niezła numerycznie i intuicyjnie zrozumiała. Jest ona bezpośrednio dostosowana do równań ruchu, tj. równań operujących położeniem i prędkością. Oto jej klasyczny wariant, tzw. metoda prędkościowa Verleta. Zauważamy, że jeśli dysponujemy położeniem  $x$ , a więc i siłami, czyli przyspieszeniem  $a$ , możemy od razu skorzystać z lepszego niż Eulerowskie wyrażenia na zmianę położenia, dokładniejszego, gdyż rzędu  $\Delta t^2$ :

$$x_{n+1} = x_n + v_n \Delta t + a_n \frac{\Delta t^2}{2}. \quad (5.1)$$

Jeśli przyspieszenie jest stałe (np. przy rzucie w stałym polu grawitacyjnym), wzór ten jest dokładny. Do tego należy dodać wzór iteracyjny na zmianę prędkości. Skorzystamy z faktu, że metody „centrowane” zwykle są dokładniejsze od metod ekstrapolacyjnych, więc skorzystamy z tego wariantu dla prędkości. Metoda Verleta sprowadza się do wzoru (5.1) połączonego z następującym:

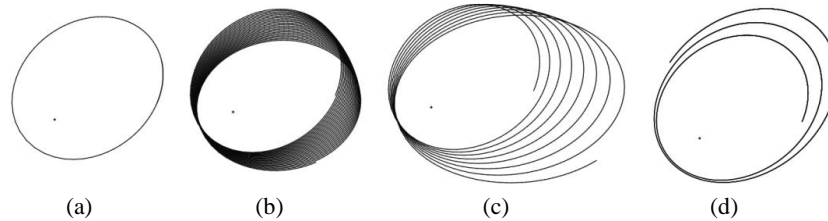
$$\begin{aligned} a_{n+1} &= F(x_{n+1})/m, \\ v_{n+1} &= v_n + \frac{1}{2} (a_n + a_{n+1}) \Delta t. \end{aligned} \quad (5.2)$$

Błąd metody Verleta jest niejednorodny, rzędu  $\Delta t^3$  dla prędkości, oraz rzędu  $\Delta t^4$  dla położenia. Globalny błąd w obu przypadkach:  $\propto \Delta t^2$ .

## 6. Przykład

Oto prosty problem Keplera, przyspieszenie jest proporcjonalne do  $-\frac{\vec{x}}{|\vec{x}|^3}$ . Rozwiązaniem jest oczywiście elipsa, przy użytych parametrach jej długa oś jest równa ok. 3. Przedstawiamy ją na rys. 3, na pierwszym wykresie z lewej; jest to wynik zastosowania metody RK4, przedyskutowanej w następnym rozdziale, z  $\Delta t = 0,25$ , co jest wartością dość dużą. Przy tej samej wartości kroku czasowego metoda Verleta precesuje. Metoda centralna Eulera również precesuje, z gorszą dokładnością niż Verlet. Metoda Heuna i RK2 precesują, a ponadto trajektoria się rozbiega, promień elipsy rośnie. Ostatni wykres jest wynikiem zastosowania podstawowej metody Eulera, ale od razu z krokiem czasowym 25-krotnie mniejszym, 0,01. Wyraźnie widać, że nie jest ona bezpośrednio użyteczna w praktyce.





Rys. 3. Porównanie metod RK4 (a), Verleta (b), Heuna (c) i Eulera (d)

Dla metody Verleta  $\Delta t$  rzędu 0,04 wystarczy, aby osiągnąć dokładność zbliżoną do RK4 powyżej. Dowolnie mały krok czasowy dla metody ekstrapolacyjnej Eulera powoduje rozbieganie się algorytmu.

## 7. Metody wyższych rzędów

### 7.1. Runge-Kutta rzędu 4

Najbardziej popularną wśród metod klasycznych jest metoda Rungego-Kutty rzędu 4. Będzie to jedyna metoda uniwersalna omówiona tutaj. Przypomnijmy, że metoda RK2 opiera się o ekstrapolacyjne oszacowanie wartości „międzykrokowej”  $\tilde{x}_{n+1/2}$ , a następnie użycie tej wartości do obliczenia  $x_{n+1}$ , przez zastosowanie metody centrowanej. Możemy tę ideę pociągnąć dalej i wprowadzić jeszcze kilka innych etapów pośrednich; będzie to równoważne przybliżeniu trajektorii między krokami nie przez prostą sieczną, tylko przez krzywą wielomianową wyższego rzędu. Oto więc algorytm RK4, już bez dodatkowych komentarzy teoretycznych. Jest on samo-startujący. Zaczynamy od czterech wartości pomocniczych:

$$\begin{aligned}
 k_1 &= f(x_n, t_n), \\
 k_2 &= f(x_n + \Delta t/2 \cdot k_1, t_n + \Delta t/2), \\
 k_3 &= f(x_n + \Delta t/2 \cdot k_2, t_n + \Delta t/2), \\
 k_4 &= f(x_n + \Delta t \cdot k_3, t_n + \Delta t).
 \end{aligned}
 \tag{7.1}$$

Następnie określamy końcowy wynik poprzez

$$x_{n+1} = x_n + \frac{\Delta t}{6} \cdot (k_1 + 2k_2 + 2k_3 + k_4).
 \tag{7.2}$$

Wielkości  $k_1, k_4$  opisują nachylenie trajektorii na początku i na końcu przedziałów, a  $k_2, k_3$  – w środku, obliczane dokładniej niż w RK2 dzięki dwukrotnej iteracji. Błąd pojedynczego kroku jest proporcjonalny do  $\Delta t^5$ , a akumulowany błąd – do  $\Delta t^4$ , więc ta metoda jest dokładniejsza od przedyskutowanych wyżej. Jej główną wadą z punktu widzenia praktyki nauczycieli fizyki, jest jej względna złożoność i brak oczywistości; sądzimy, że aby zachęcić uczniów i nauczy-

cieli do odrobiny programowania, podstawowe metody obliczeniowe winny być stosunkowo proste do przyswojenia i do odtworzenia z pamięci.

## 7.2. Ogólny schemat poprawek wyższego rzędu

Zakończmy ten rozdział omówieniem ogólnego schematu poprawek wyższego rzędu, dość oczywistego, przedstawionego – dla prostoty – dla układów autonomicznych, bez jawnej zależności czasowej. Jeśli  $\dot{x} = f(x)$ , i znamy początkową wartość  $x$ , oraz jawną postać funkcji  $f$ , potrafimy też obliczyć krzywiznę i wyższe pochodne  $x$ :  $\ddot{x} = f'(x) \cdot \dot{x} = f'(x)f(x)$ . Następnie  $x^{(3)} = f''(x) \cdot (\dot{x})^2 + f'(x) \cdot \ddot{x}$  itd. Ekstrapolacyjna metoda Eulera drugiego rzędu przyjmie więc postać

$$x_{n+1} = x_n + \Delta t \cdot f(x_n) + \frac{\Delta t^2}{2} f'(x_n) \cdot f(x_n) \quad (7.3)$$

itd. są to po prostu człony szeregu Taylora dla  $x$ . Opłacalność tej metody nie jest oczywista, ale służy ona czasami do konkretnej oceny lokalnego błędu metod prostszych.

## 8. Stabilność i poprawki energetyczne

Przykład Keplera powyżej jest „zbyt przyzwoity”. Trajektorie całkowalne, periodyczne, a więc stabilne, są słabym testem metod numerycznych. Aby uniknąć przykrych niespodzianek, przy testowaniu i parametryzacji użytej metody należy odważyć się na problem, w którym trajektorie są nieregularne. W tym przypadku układ często zachowuje się ergodycznie, trajektorie wypełniają przestrzeń i jeśli mamy do czynienia z systemem grawitacyjnym, może się zdarzyć, że dwa ciała zblizną się do siebie tak blisko, że bardzo duże przyspieszenie grawitacyjne destabilizuje metodę. Oto przykład z układem trzech ciał. Jak wiadomo, nie jest on całkowalny i chaotycznych trajektorii na ogół nie unikniemy<sup>1</sup>. Jednak w prawdziwej fizyce energia i moment pędu są zachowane i całość winna globalnie zajmować stały obszar przestrzeni położeń i pędów. Nie jest to prawdą w symulacji... Rysunek 4 przedstawia wynik metody Verleta dla trzech ciał o identycznych masach i  $\Delta t = 0,01$ , co winno wystarczyć dla stabilności w typowych warunkach. Warunki początkowe są dość losowe, ale tak dobrane, aby ruch odbywał się w skończonym obszarze przestrzeni. Na osiach mamy  $x$  i  $y$  w umownych jednostkach,  $z$  jest współrzędną prostopadłą do rysunku.

Trajektorie są nieregularne, ale przez dłuższy czas całkowita energia jest zachowywana (i jest ujemna, tj. trajektorie nie rozbiegają się do nieskończoności).

<sup>1</sup> Bywają również orbity regularne i stabilne wielu ciał. Bardzo zabawną jest *wspólna*, płaska orbita o kształcie lemniskaty dla trzech identycznych ciał, które jak gdyby „zonglowały” wokół siebie nawzajem. Zob.: <http://arxiv.org/pdf/math/0011268v1.pdf>.

Jednak w pewnej, dość przypadkowej chwili dwa ciała zbliżają się do siebie „niebezpiecznie”. Ich przyspieszenie gwałtownie wzrasta, dodatnia energia kinetyczna przekracza ujemną potencjalną i dwie „planety” uciekają (dwa odcinki prawie proste na rysunku) ze względu na destabilizację użytej metody numerycznej. Jest to niedopuszczalny artefakt, który można odroczyć zmniejszając  $\Delta t$ , ale wówczas może on nastąpić później...



Rys. 4. Niestabilny układ trzech ciał

Jednym ze sposobów uniknięcia tej katastrofy jest monitorowanie energii podsystemów i całkowitej. Gdy jej zachowanie przekracza dopuszczalny błąd, należy zmodyfikować energię kinetyczną jednego lub kilku ciał, aby wrócić do normy. Technicznie jest to bardzo trudne, gdyż *w naturze zbliżone zjawiska mogą też wystąpić*, np. dwa ciała zbliżają się tak, że ich energia potencjalna staje się bardzo ujemna, za to jakieś trzecie nabiera większej prędkości i ucieka. Takie zjawiska mogą wystąpić w pasie asteroidów, który „strzela meteorami” i przewidzieć je jest niezmiernie trudno.

## 9. Wnioski końcowe

Nasz wykład miał charakter jakościowy, sprowadził się do kilku przepisów, bez dokładniejszej analizy jakości rozwiązań. Głównym jego celem jest pokazanie, że rozwiązywanie numeryczne równań różniczkowych w sposób racjonalny i nienaiwny, jest dość intuicyjne.

Pominęliśmy całą, bardzo ważną, klasę metod ekstrapolacyjno-interpolacyjnych (jedynie metodę Heuna można potraktować jako prymitywną metodę tej klasy), które „ratują” efektywność obliczeń tam, gdzie metody RK wymagałyby znacznie mniejszego kroku. Nie wspomnieliśmy o metodach operujących zmiennym krokiem, ani o tzw. problemach sztywnych, w których zachowanie trajektorii miesza ze sobą zmiany wolniejsze i szybsze. W astronomii (nie tylko) korzysta się z metod wymagających rozwinięć  $x$  w potęgę  $\Delta t$  dość wysokiego stopnia. Pominęliśmy także bardzo ważny problem ruchu ograniczonego, z więzami, którego znaczenie dla fizyków jest podstawowe i być może do tego kiedyś wrócimy.

Techniki rozwiązywania równań różniczkowych rozwijają się nieprzerwanie od 250 lat. Metody Rungego-Kutty powstały na początku dwudziestego wieku. Aktualnie nie tylko szlifuje się matematykę, ale został dokonany olbrzymi postęp w dziedzinie paralelizacji obliczeń, przyspieszenia rachunków dzięki architekturom i algorytmom współbieżnym, które są dość trudne. Fizyk interesujący się symulacją winien w tym wszystkim się nieco orientować lub współpracować z dobrym informatykiem.

„Najlepszej” metody nie ma, gdyż nasze kryteria jakości mogą być różne, np. zależy nam na szybkości obliczeń bardziej niż na dokładności. Ponadto, w pewnych przypadkach (np. gdy obiekty mechaniczne w programie są sterowane przez mechanizmy losowe lub przez użytkownika) i układ nie jest autonomiczny, błędy wynikające z użycia prymitywnej metody nie zdążą narosnąć, układ nie ma czasu się destabilizować. (To zjawisko występuje w grach, gdzie rzetelność fizyki bywa często umowna).

Tak więc próba odpowiedzi na pytanie: na której metodzie się skoncentrować, przypomina bardzo znaną anegdotę amerykańską. Turysta pyta przechodnia: „Proszę pana, jak trafić do Carnegie Hall?” (Jest to słynna, wręcz legendarna nowojorska sala koncertowa). Odpowiedź brzmi: „Pracą, pracą, i jeszcze raz ciężką pracą, proszę pana...”.

Należy testować i jeszcze raz testować i porównywać różne metody. Jeśli głównym celem symulacji jest wizualizacja, nie ma potrzeby „podkręcenia” dokładności do granic przewyższających znacznie rozdzielczość ekranu, nie zapominając jednak, że z czasem błąd narasta. Jak wspomnieliśmy w rozdz. 8, dobrze jest czasami kontrolować zmiany energii, ale także innych zachowywanych wielkości, np. momentu pędu. Jeżeli Czytelnik zechce pogłębić swoje wiadomości o *technikach specyficznym dostosowanych do równań ruchu* w fizyce, polecamy literaturę o *całkowaniu symplektycznym*. Jest to tematyka zaawansowana, ale intuicyjna, stosunkowo oczywista i pięknie pokazująca po raz kolejny potęgę podejścia geometrycznego do mechaniki.

Nie podajemy referencji bibliograficznych, gdyż na poziomie typowego Czytelnika wystarczy wiedzieć, że Wikipedia zawiera odpowiednio dużo bardziej szczegółowego materiału, a ponadto praktycznie każda solidniejsza książka poświęcona metodom numerycznym zawiera rozdział o równaniach różniczkowych. Celem tego tekstu było zebranie w jedną całość informacji użytecznych dla prostych, typowych problemów, które można opanować szybko oraz podanie kilku informacji praktycznych, które zwykle są pomijane w podręcznikach specjalistycznych. Zachęcam do skontaktowania się ze mną w przypadku pytań lub problemów obliczeniowych.