

Pracownia z analizy numerycznej

Sprawozdanie do zadania P.1.6.

Prowadzący: mgr. Filip Chudy

Wrocław, 20 listopada 2022, 21:37

Spis treści

1. Wstęp	1
1.1. Metoda	2
2. Pierwsze próby	2
2.1. Interpretacja geometryczna	2
2.2. Algorytm Monte Carlo	4
2.3. Szereg Taylora	6
3. Wzór Viete’a	7
3.1. Wyniki	8
4. Algorytm Chudnowsky’ch	9
4.1. Wyniki	9
5. Algorytm Gaussa-Legendre’a	10
5.1. Wyniki	11
6. Podsumowanie	13
Literatura	13

1. Wstęp

W 1897 roku Amerykański fizyk Edward J. Goodwin oznajmił światu, że udało mu się poprawnie skonstruować kwadrat o polu równym polu koła. W tym samym roku jego wynik został przedłożony Zebraniu Stanu Indiana i w życie wszedł Artykuł o Liczbie Pi, na mocy którego $\pi = 3.2$. Ponad 2000 lata wcześniej, w III wieku p.n.e., Archimedes napisał

$$3.1408 \approx \frac{223}{71} < \pi < \frac{22}{7} \approx 3.1428,$$

co daje ograniczenia bliższe tym dzisiaj uważanym za najdokładniejsze niż artykuł z XIX wieku. W 2009 roku za pomocą algorytmu braci Chudowsky’ch, wyprowadzonego ze wzorów S. Ramanujana, zostało osiągnięte przybliżenie π przez 10 trilionów cyfr znaczących. Dzisiaj istnieją jeszcze szybsze algorytmy na przybliżanie π niż ten z 2009 roku i badanie metod przybliżania wartości matematycznych jest nadal żywą dziedziną nauki.

W tej pracy przyjmiemy się 5 metodom przybliżania liczby π . W Rozdziale 2. zaprezentujemy naiwne algorytmy, z których jedna okazała się nie odstawać od bardziej zaawansowanych sposobów zaprezentowanych w dalszych rozdziałach. W Rozdziałach 3 i 4 przyjmiemy się dwóm metodom liniowo zbieżnym do π . W Rozdziale 5. zaprezentowany jest algorytm zbieżny kwadratowo do liczby π .

1.1. Metoda

Do obliczeń używaliśmy precyzji 16 069 i zmiennych typu `BigFloat` w języku `Julia`. Za wartość dokładną π użyliśmy wartości bibliotecznej typu `Irrational`. Dzięki temu jest ona wyliczana dla każdej precyzji [5].

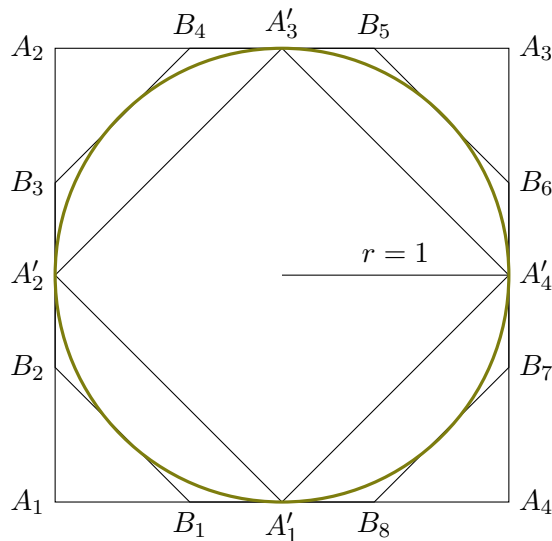
Dla każdej metody eksperymentalnie szacowaliśmy rząd zbieżności i przedstawiliśmy wyniki na odpowiednich wykresach.

2. Pierwsze próby

2.1. Interpretacja geometryczna

Bardzo często π jest definiowane jako stosunek obwodu okręgu do jego średnicy. W historii pojawiało się wiele prób wyznaczenia π korzystając z obwodu wielokątów foremnych wpisanych w oraz opisanych na okręgu jednostkowym. Wraz ze wzrostem liczby boków zwiększa się dokładność oszacowań obwodu okręgu, co daje coraz to bliższe prawdy granice na wartość ludolfiny.

Takie podejście stosował już w starożytności Archimedes. Wyprowadził on wzór rekursyjny na obwód $2n$ -kąta foremnego wpisanego oraz opisanego na okręgu na podstawie obwodu n -kąta.



Rysunek 1. Wielokąty opisane i wpisane w okrąg o promieniu 1.

Wpiszmy n -kąć foremny w okrąg o promieniu 1. Teraz na tym samym okręgu opiszmy n -kąć tak, żeby wierzchołki wielokąta wpisanego były środkami boków wielokąta opisywanego. Dostajemy w ten sposób n -kąć foremny opisany na okręgu o promieniu 1. Nietrudno zauważyć, że teraz jeśli połączymy sąsiednie boki n -kąta opisanego odcinkami st stycznymi do okręgu o końcach w równej odległości od najbliższego wierzchołka, to dostaniemy $2n$ -kąć foremny. Sytuacja dla $n = 4$ została przedstawiona na Rysunku 1.

Rozważmy teraz trójkąt $\triangle A'_1 A_1 A_2$. Zauważmy, że odcinek $\overline{B_1 B_2}$ dzieli go na dwa trójkąty podobne:

$$\triangle B_1 A_1 B_2 \sim \triangle A'_1 A_1 A'_2.$$

Dla przejrzystości zapisów oznaczmy $|\overline{A_1 A_2}| = A$, $|\overline{B_1 B_2}| = B$ oraz $|\overline{A'_1 A'_2}| = a$. Z proporcji w trójkątach podobnych mamy:

$$\begin{aligned}\frac{B}{\frac{1}{2}A - \frac{1}{2}B} &= \frac{a}{\frac{1}{2}A} \\ B &= \frac{a}{A}(A - B) \\ B &= a - \frac{a}{A}B \\ B &= \frac{aA}{A + a}\end{aligned}$$

Oznaczmy teraz obwód n -kąta wpisanego jako l_n , a n -kąta opisanego - L_n . Według Rysunku 1 są one równe:

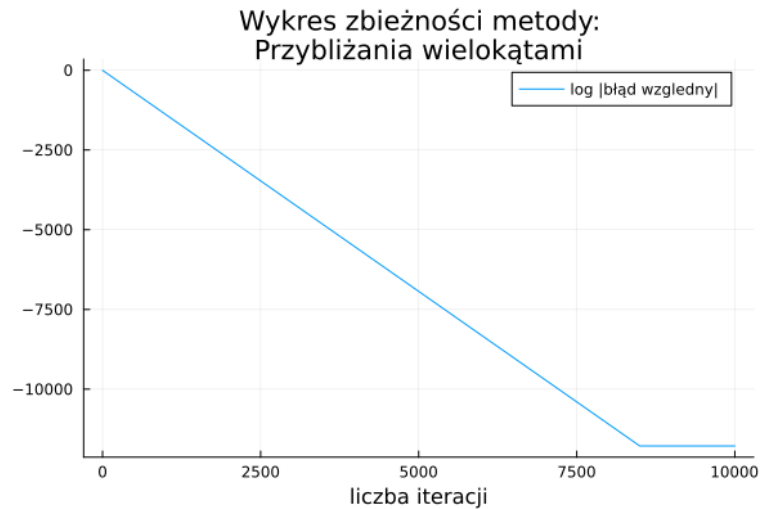
$$\begin{aligned}l_n &= na \\ L_n &= nA \\ L_{2n} &= 2nB = 2n \frac{aA}{A + a} = 2n^2 \frac{aA}{An + an} = 2 \frac{L_n l_n}{L_n + l_n}\end{aligned}$$

Dalej, oznaczmy długość boku $2n$ -kąta wpisanego jako b . Zauważmy, że wówczas:

$$\begin{aligned}B &= 2 \tan \frac{\pi}{2n} \\ a &= 2 \sin \frac{\pi}{n} \\ b &= 2 \sin \frac{\pi}{2n}\end{aligned}$$

oraz:

$$\begin{aligned}l_{2n} &= 2nb = 4n \sin \frac{\pi}{2n} = \sqrt{16n^2 \sin^2 \frac{\pi}{2n}} = \sqrt{8n^2 \frac{\sin \frac{\pi}{2n}}{\cos \frac{\pi}{2n}} 2 \sin \frac{\pi}{2n} \cos \frac{\pi}{2n}} = \\ &= \sqrt{8n^2 \tan \frac{\pi}{2n} \sin \frac{\pi}{n}} = \sqrt{2nBna} = \sqrt{L_{2n}l_n}.\end{aligned}$$



Wykres 2. Wykres logarytmu dziesiętnego z błędów względnego dla przybliżenia π za pomocą metody geometrycznej.

Zauważmy, że $\lim_{k \rightarrow \infty} L_k = 2\pi$ i $\lim_{k \rightarrow \infty} l_k = 2\pi$ oraz dla każdego k mamy

$$l_n \leq 2\pi \leq L_n,$$

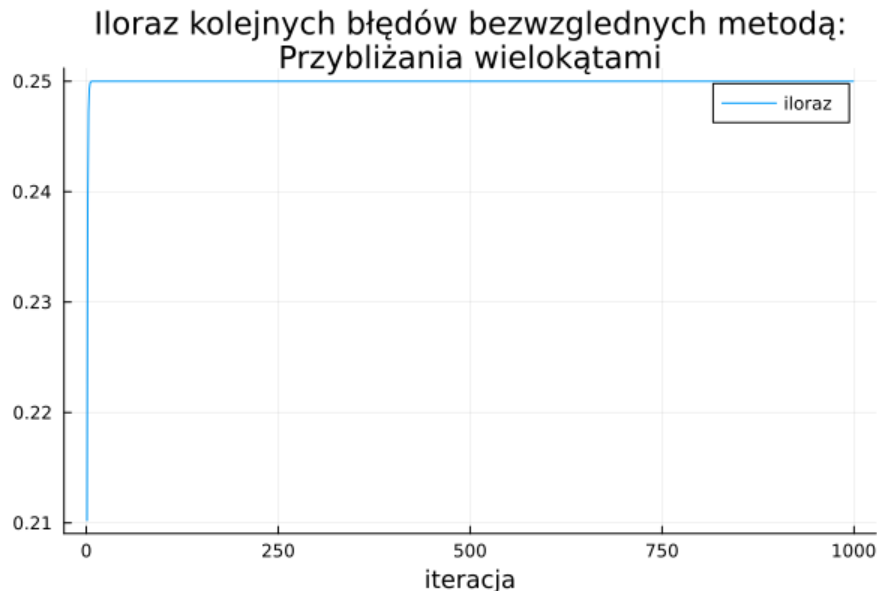
a więc możemy przybliżać π jako

$$\pi \approx \frac{L_n - l_n}{4},$$

czyli jako środek przedziału $[l_n, L_n]$. Wybierzemy punkt startowy jako trójkąt równoboczny:

$$\begin{cases} l_3 = 3\sqrt{3} \\ L_3 = 6\sqrt{3} \end{cases}$$

Metoda w okolicach 8050 iteracji uzyskiwała błąd bezwzględny rzędu 10^{-12000} . Na Wykresie 2 widzimy, że gradient wykresu logarytmu dziesiętnego od błędu względnego jest niemalże liczbą stałą. Przybliżanie liczby π przy użyciu geometrii osiąga limit precyzji, co jest bardzo imponujące biorąc pod uwagę jak stara jest to metoda.



Wykres 3. Wykres ilorazu błędów względnych wyrazu $n + 1$ i n dla metody geometrycznej.

Postawiliśmy hipotezę, że metoda ta jest zbieżna liniowo. Eksperymentalne wyznaczanie rzędu zbieżności potwierdziło to przypuszczenie, co widać na Wykresie 3. Widać na nim, że

$$\lim_{n \rightarrow \infty} \frac{|x_{n+1} - \pi|}{|x_n - \pi|} \approx \frac{1}{4},$$

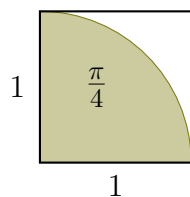
gdzie x_n to przybliżenie w n -tej iteracji.

2.2. Algorytm Monte Carlo

Tak jak w poprzedniej metodzie, możemy skorzystać z faktu, że dla koła jednostkowego π jest równe jego polu. Zauważmy, że jeżeli będziemy wybierać losowo punkty kwadratu o polu 1, to $\frac{\pi}{4}$ z nich powinno znaleźć się w ćwiartce koła o środku w jednym z wierzchołków tego kwadratu (Rysunek 4.).

Korzystając z algorytmu Monte Carlo możemy wybierać losowo współrzędne $x, y \in [0, 1]$ kolejnych punktów, a następnie sprawdzać ile z nich spełnia warunek

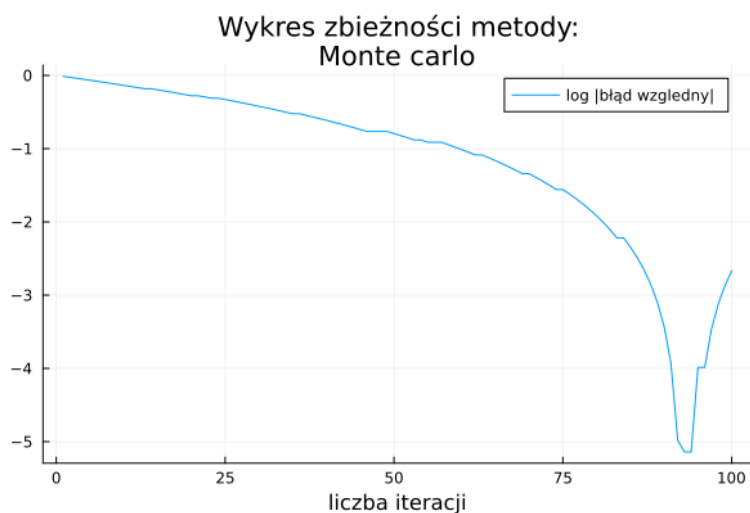
$$x^2 + y^2 \leq 1.$$



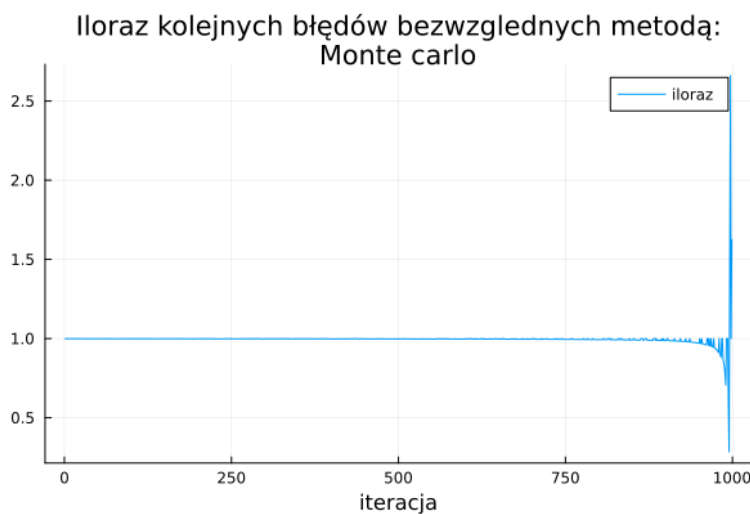
Rysunek 4. Stosunek pola ćwiartki koła jednostkowego do kwadratu o boku 1

Otrzymany stosunek będzie coraz bliższy $\frac{\pi}{4}$ wraz ze zwiększaniem ilości testowanych punktów.

Na Wykresie 5. zaprezentowany jest logarytm dziesiętny z błędu względnego metody Monte Carlo. Szacowanie zbieżności tej metody wykracza poza zakres wiedzy studenta 3 semestru ze względu na losowość tego algorytmu. Wykorzystanie technik z kursu Rachunku Prawdopodobieństwa ułatwiłoby to zadanie. Na Wykresie 6 widzimy jeden z wykresów ilorazu błędów kolejnych wyrazów jaki uzyskaliśmy przy uruchamianiu algorytmu. Na obu wykresach obserwujemy wachania lokalne zbieżności algorytmu.



Wykres 5. Wykres logarytmu dziesiętnego z błędu względnego uzyskanego dla metody przybliżenia π z pomocą algorytmu Monte Carlo.



Wykres 6. Wykres ilorazu błędów względnych wyrazu $n + 1$ i n dla metody z wykorzystaniem algorytmu Monte Carlo.

2.3. Szereg Taylora

W matematyce bardzo często w celu przybliżania porządkanych wartości używa się szeregów Taylora. Tak dla przykładu, korzystając z rozszerzenia funkcji $\arctan x$ w punkcie 0 możemy oszacować wartość $\frac{\pi}{4}$:

$$(1) \quad \begin{aligned} \frac{\pi}{4} = \arctan 1 &= \sum_{k=0}^{\infty} \frac{\arctan^{(k)} 0}{k!} (1-0)^k = \\ &= 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots = \sum_{k=0}^{\infty} \frac{(-1)^k}{2k+1}. \end{aligned}$$

W obliczeniach praktycznych nie możliwe jest dodawanie kolejnych elementów sumy w nieskończoność. Konieczne jest więc zatrzymanie się na pewnym N , co daje pewien błąd, R_N :

$$\frac{\pi}{4} \approx \sum_{k=0}^N \frac{(-1)^k}{2k+1} + R_N.$$

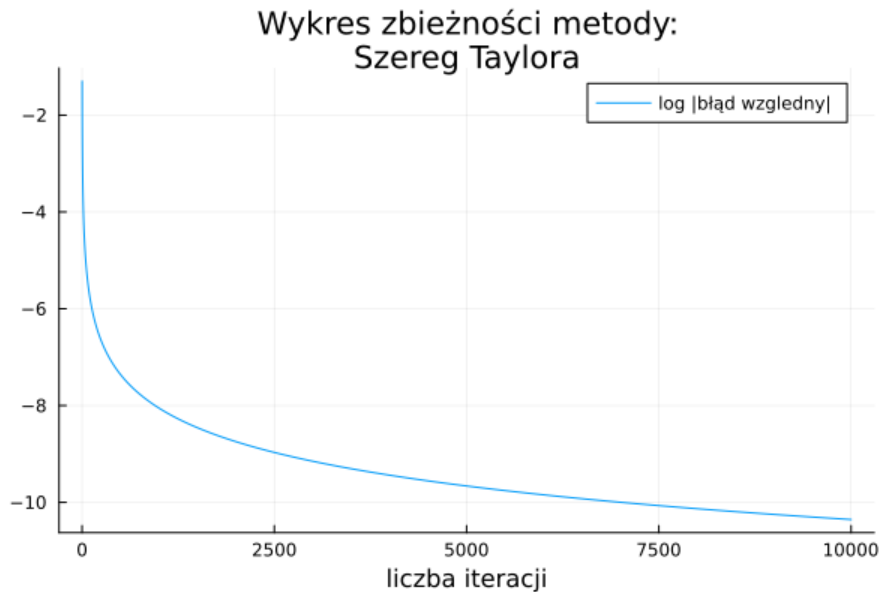
Oznaczmy tę sumę jako P_N . Ponieważ dla przybliżeń funkcji szeregiem Taylora coraz wyższego stopnia dostajemy coraz dokładniejszy wynik, to P_{N+1} powinno być dokładniejsze niż P_N . Zauważamy też, że

$$P_{N+1} - P_N = \frac{(-1)^{N+1}}{2N+3}$$

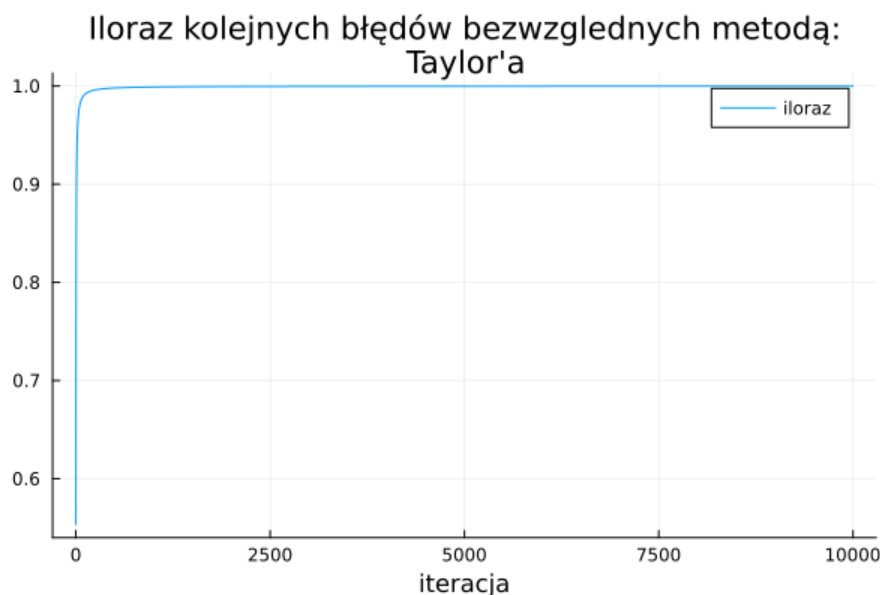
w takim razie możemy oszacować błąd dla szeregu Taylora N -tego stopnia za pomocą

$$R_N \approx \max \frac{(-1)^{N+1}}{2N+3}.$$

Powyższa metoda jest o wiele wolniejsza od metody Archimedesesa, mimo że powstała później. W metodzie geometrycznej osiągaliliśmy błąd rzędu 10^{-12000} , natomiast wzór Taylora daje błąd rzędu 10^{-10} . Iloraz wyrazu $n+1$ do wyrazu n eksperymentalnie zbiega do 1, co sugeruje zbieżność nadliniową [6].



Wykres 7. Wykres logarytmu dziesiętnego z błędów względnego uzyskanego dla metody przybliżenia π za pomocą szeregu Taylora.



Wykres 8. Wykres ilorazu błędów względnych wyrazu $n + 1$ i n dla metody z wykorzystaniem szeregu Taylora.

3. Wzór Viete'a

Viete wyprowadził swoją formułę na π obserwując stosunek pola 2^n -kąta foremnego do pola 2^{n+1} -kąta foremnego. Poprzez zwiększanie n w nieskończoność, jesteśmy w stanie dostać stosunek 2^2 -kąta foremnego, czyli kwadratu, do pola koła w które został on wpisany. Można ją też wyprowadzić za pomocą tożsamości udowodnionej przez Eulera ponad 100 lat po śmierci Viete'a.

Wzór zaproponowany przez Viete'a, uznawany za prekursor analizy matematycznej w matematyce poprzez pierwsze wykorzystanie nieskończonego ilorazu, jest następujący:

$$(2) \quad \frac{2}{\pi} = \prod_{k=1}^n \frac{a_k}{2},$$

gdzie $a_1 = \sqrt{2}$ oraz

$$a_k = \sqrt{2 + a_{k-1}}.$$

Wiemy, że

$$\frac{\sin x}{x} = \cos \frac{x}{2} \cos \frac{x}{4} \dots = \lim_{n \rightarrow \infty} \prod_{k=1}^n \cos \frac{x}{2^k}$$

oraz

$$\cos \frac{x}{2} = \sqrt{\frac{1 + \cos x}{2}}.$$

Jeśli wstawimy $x = \frac{\pi}{2}$ i oznaczymy $b_k = \cos \frac{x}{2^k}$, $b_1 = \frac{\sqrt{2}}{2}$, dostaniemy

$$\frac{\sin \frac{\pi}{2}}{\frac{\pi}{2}} = \frac{2}{\pi} = \lim_{n \rightarrow \infty} \prod_{k=1}^n \cos \frac{x}{2^k} = \lim_{n \rightarrow \infty} \prod_{k=1}^n b_k = \lim_{n \rightarrow \infty} \prod_{k=1}^n \frac{a_k}{2},$$

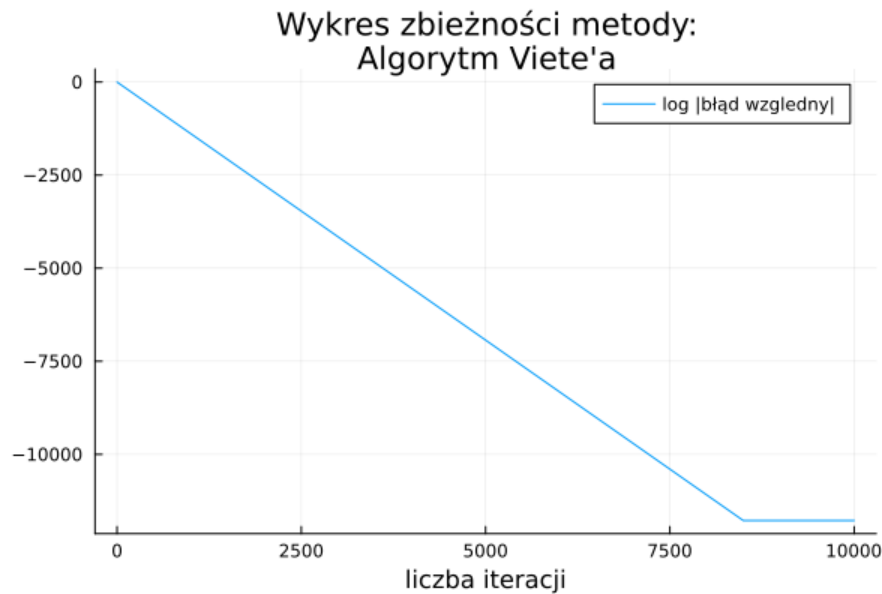
gdzie

$$a_k = 2b_k = 2\sqrt{\frac{1 + b_{k-1}}{2}} = \sqrt{2 + 2b_{k-1}} = \sqrt{2 + a_{k-1}}$$

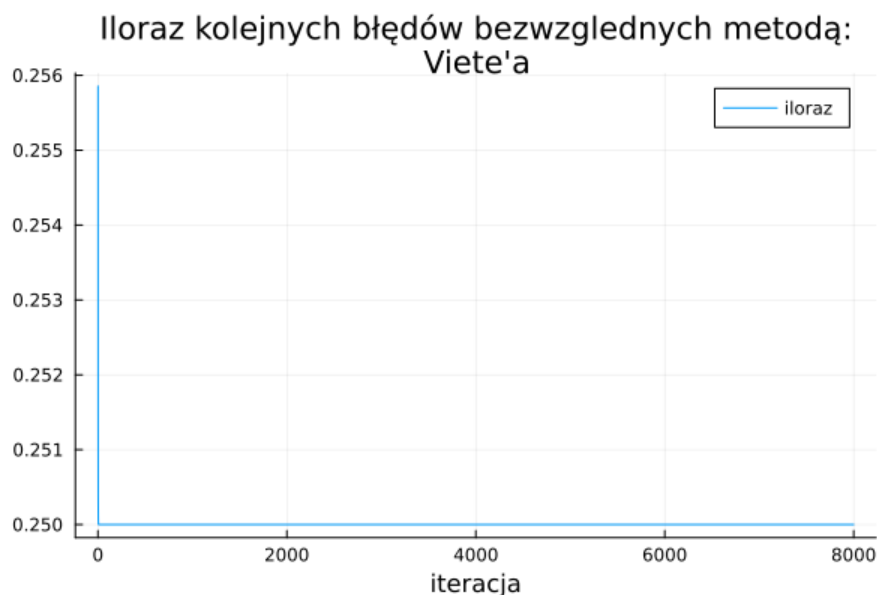
i $a_1 = \sqrt{2}$.

3.1. Wyniki

Na Wykresie 9. zaprezentowany jest wykres zbieżności algorytmu Viete'a. Eksperymentalne wyznaczenie rzędu zbieżności, tak jak i gradient prezentowanego wykresu, sugerują liniową zbieżność tej metody wyliczania Viete. Od około 8750 iteracji wartość zwracana przez metodę Viete'a pokrywa się z wartością wyliczaną bibliotecznie.



Wykres 9. Wykres logarytmu dziesiętnego z błędów względnego dla przybliżenia π za pomocą metody Viete'a.



Wykres 10. Wykres ilorazu błędów względnych wyrazu $n + 1$ i n dla metody z wykorzystaniem metody Viete'a.

Metoda Viete'a daje wyniki podobne do podejścia geometrycznego opisanego w Sekcji 2.1. Obie metody mają błąd bardzo bliski zera dla około 8000 iteracji metody. W obu metodach korzystamy z dwóch zmiennych, więc są podobne pamięciowo. W metodzie zaproponowanej przez Archimedes'a musimy zapamiętywać obwód figury opisaną i wpisaną w okrąg, natomiast dla metody Viete'a potrzebujemy zapisywać dotychczasowy iloczyn oraz kolejny wyraz ciągu a_k . Różnią się one jedynie

ilością operacji jakie wykonujemy w jednej iteracji, więc metoda Viete’a ma marginalnie lepszą stałą czasową.

Dodatkowo, jak widać na Wykresie 10., eksperymentalnie potwierdziliśmy, że metoda ta jest zbieżna liniowo. Tak jak w przypadku metody Archimedesesa mamy

$$\lim_{k \rightarrow \infty} \frac{|x_{k+1} - \pi|}{|x_k - \pi|} \approx \frac{1}{4},$$

gdzie x_k to wynik k -tej iteracji metody.

4. Algorytm Chudnowsky’ch

Algorytm zaproponowany przez braci Chudnowsky opiera się na 17 wzorach na $\frac{1}{\pi}$ opracowanych przez Srinivasa Ramanujan[1]:

$$\frac{1}{\pi} = \frac{1}{426880\sqrt{10005}} \sum_{k=0}^{\infty} \frac{(6k)!(13591409 + 545140134k)}{(3k)!(k!)^3(-640320)^{3k}}.$$

Z tego można uzyskać π wprost w formie wzoru:

$$\pi = C \left(\sum_{q=0}^{\infty} \frac{M_q \cdot L_q}{X_q} \right)^{-1},$$

gdzie

$$\begin{cases} C = 426880\sqrt{10005} \\ L_{q+1} = L_q + 545140134 & L_0 = 13591409 \\ X_{q+1} = X_q \cdot (-262537412640768000) & X_0 = 1 \\ K_{q+1} = K_q + 12 & K_0 = -6 \\ M_{q+1} = M_q \cdot \left(\frac{K_{q+1}^3 - 16K_{q+1}}{(q+1)^3} \right) & M_0 = 1. \end{cases}$$

Aby wyprowadzić ten wzór, jak i inne podane przez Ramanujana, potrzebna jest znajomość między innymi teorii funkcji eliptycznych[1]. Z tego też względu w tym raporcie nie podejmiemy się uzasadniania poprawności wyżej podanego wzoru. Dla zainteresowanych polecamy lekturę ”Collected Papers of Srinivasa Ramanujan”[2].

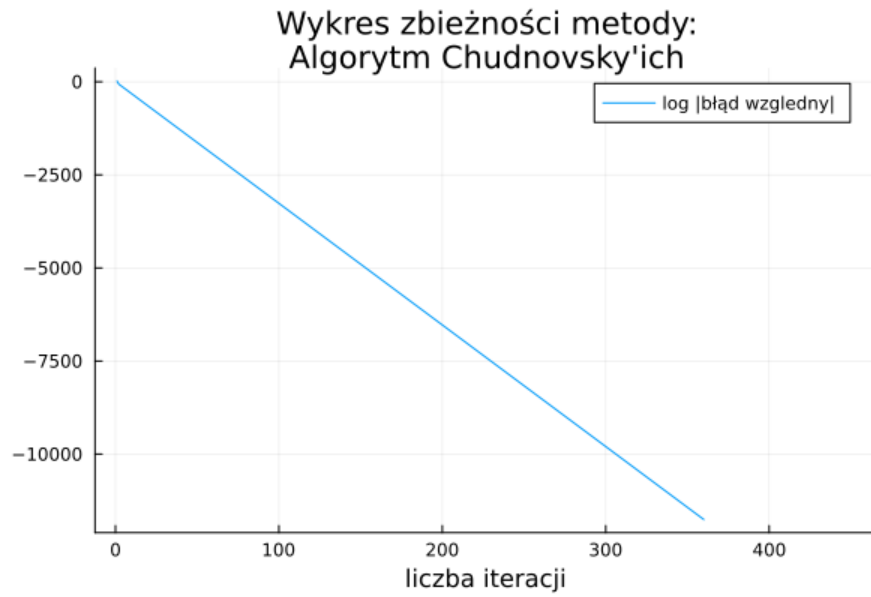
4.1. Wyniki

Logarytm z błędu względnego algorytmu Chudnowsky’ch dla pierwszych 400 iteracji został zaprezentowany na Wykresie 11. Już dla 359 iteracji błąd bezwzględny jest równy 0. Każe to sugerować, że to właśnie ta metoda została użyta jako implementacja funkcji `pi()` w języku Julia. Dlatego od 359 iteracji nie ma zaznaczonej wartości \log_{10} od błędu względnego. Powoduje to anomalie widoczne na wykresach.

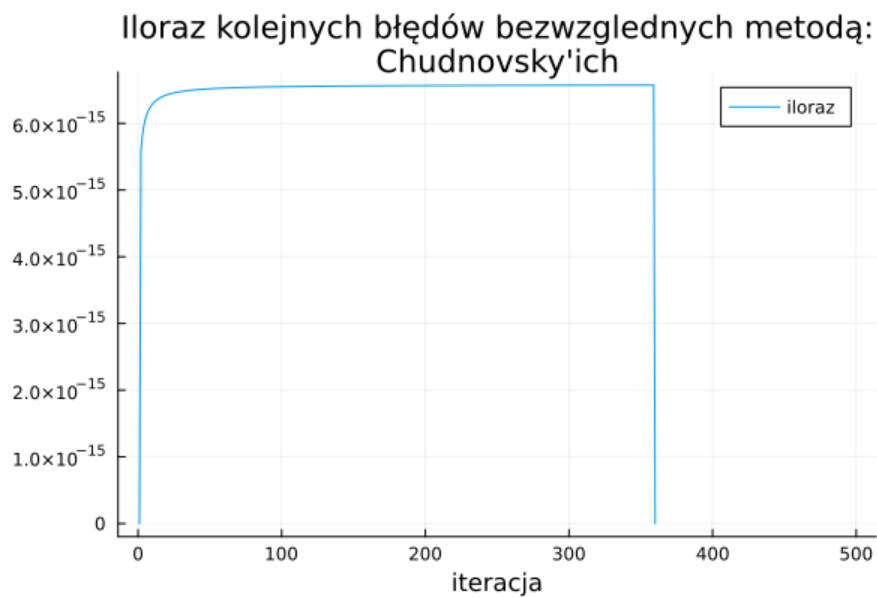
Eksperymentalne wyznaczanie zbieżności metody Chudnowsky’ch sugeruje zbieżność liniową, tak jak na Wykresie 12.. Nietypowe załamanie w okolicach 359 iteracji jest spowodowane zerową wartością błędu bezwzględnego w tym miejscu. Z pozostałej części wykresu możemy wydedukować, że

$$\lim_{k \rightarrow \infty} \frac{|x_{k+1} - \pi|}{|x_k - \pi|} \approx 6.5 \cdot 10^{-15},$$

gdzie x_k to oszacowanie π uzyskane w k -tej iteracji.



Wykres 11. Wykres logarytmu dziesiętnego z błędów względnych dla przybliżenia π za pomocą algorytmu braci Chudnowsky'ich.



Wykres 12. Wykres ilorazu błędów względnych wyrazu $n + 1$ i n dla algorytmu Chudowsky'ich.

5. Algorytm Gaussa-Legendre'a

Algorytm Gaussa-Legendre'a jest aktualnie jednym z najszybciej zbiegających algorytmów używanych do wyliczania liczb π . Został wyprowadzony na podstawie prac Carla Friedricha Gaussa oraz Adrien-Marie Legendre na podstawie współczesnych algorytmów do mnożenia i pierwiastkowania. Jest on, niestety, bardzo wymagający pamięciowo. Poniżej prezentujemy implementację tego algorytmu[4]:

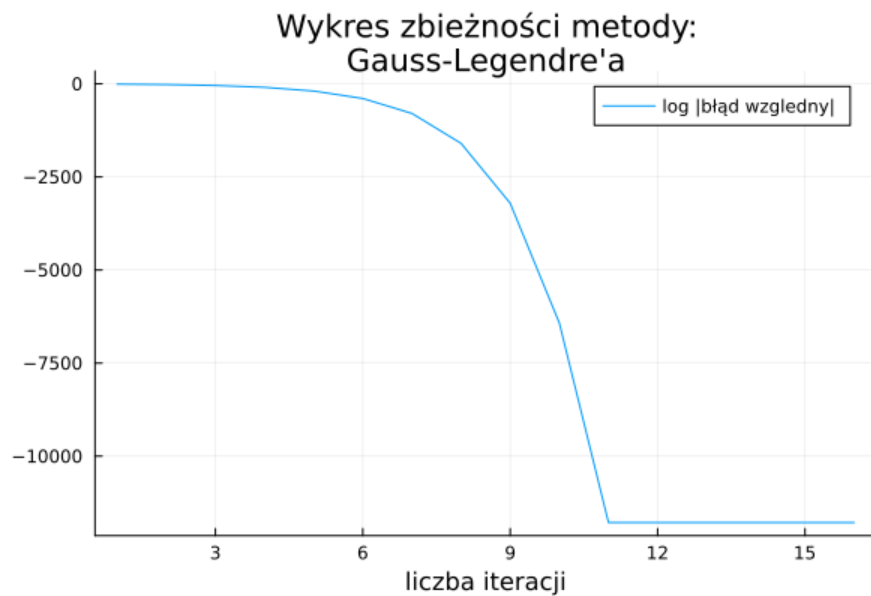
```

1 function gauss_legrendre (max):
2     a = 1
3     b = 1 / sqrt(2)
4     t = 1 / 4
5     p = 1
6     i = 0
7     while i <= max:
8         an = (a + b) / 2
9         b = sqrt(a * b)
10        t = t - p * (a - an) * (a - an)
11        p = 2 * p
12        a = an
13
14    return (a + b) * (a + b) / (4 * t)

```

5.1. Wyniki

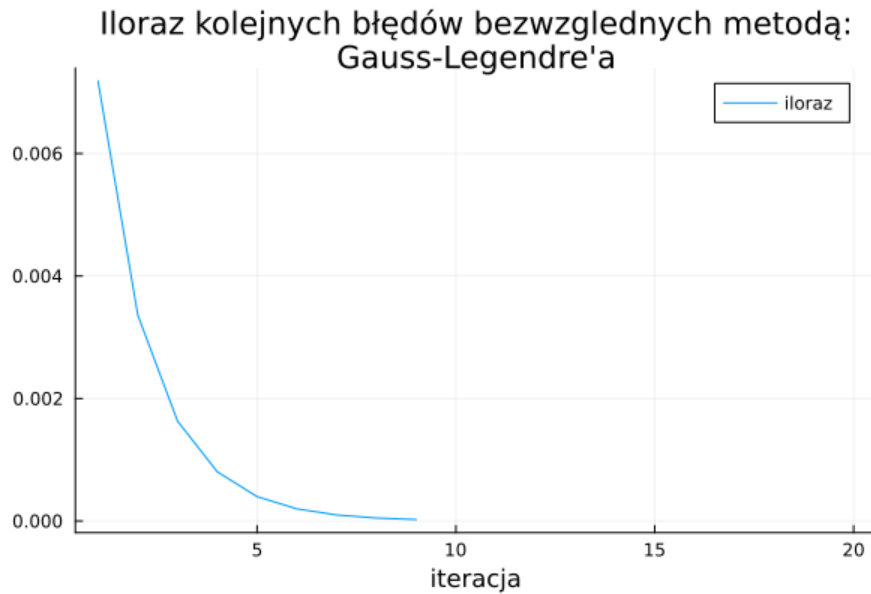
Metoda Gaussa-Legendre'a okazała się zbiegać do implementacji bibliotecznej funkcji `pi()` z języka Julia wyjątkowo szybko, bo już w 11 iteracji kwadrat błędu maszynowo był równy zero, co widać na Wykresie 13.



Wykres 13. Wykres logarytmu dziesiętnego z błędem względnym dla przybliżenia π za pomocą algorytmu Gaussa-Legendre'a.

Eksperymentalne obliczenia rzędu zbieżności tej metody jedynie potwierdzają wyższą zbieżność tego algorytmu niż w przypadku innych opisanych metod (Wykres 14). Dla precyzji wynoszącej 16 069 bitów obliczenia na podstawie dzielenia błędu $(n + 1)$ -ego wyrazu przez kwadrat błędu n -tego wyrazu nie dają konkretnych wyników przez zbyt szybkie dążenie tej metody do π . W literaturze metoda ta jest określana jako zbieżna kwadratowo [3].

Pomimo tak dobrej zbieżności, metoda ta nie jest powszechnie wykorzystywana, gdyż zużywa więcej pamięci niż metoda Chudnowsky'ch.



Wykres 14. Wykres ilorazu błędów względnych wyrazu $n + 1$ i n dla algorytmu Gaussa-Legendre'a.

Wartość π obliczona dla 10 iteracji naszego programu daje:

3.14159265358979323846264338327950288419716939937510582097494459230781640628620899862803482534211706798214808651
32823066470938446095505822317253594081284811174502841027019385211055596446229489549303819644288109756659334461284756
4823378678316527120190914564856692346034861045432664821339360726024914127372458700660631558817488152092096282925409
171536436789259036001133053054882046652138414695194151160943305727036575959195309218611738193261179310511854807446
2379962749567351885752724891227938183011949129833673362440656643086021394946395224737190702179860943702770539217176
2931767523846748184676694051320005681271452635608277857713427577896091736371787214684409012249534301465495853710507
9227968925892354201995611212902196086403441815981362977477130996051870721134999999837297804995105973173281609631859
50244594553469083026425223082533446850352619311881710100031378387528865875332083814206171776691473035982534904287554
68731159562863882353787593751957781857780532171226806613001927876611195909216420198938095257201065485863278865936153
38182796823030195203530185296899577362259941389124972177528347913151557485724245415069595082953311686172785588907509
83817546374649393192550604009277016711390098488240128583616035637076601047101819429555961989467678374494482553797747
26847104047534646208046684259069491293313677028989152104752162056966024058038150193511253382430035587640247496473263
91419927260426992279678235478163600934172164121992458631503028618297455570674983850549458858692699569092721079750930
29553211653449872027559602364806654991198818347977535663698074265425278625518184175746728909777727938000816470600161
45249192173217214772350141441973568548161361157352552133475741849468438523323907394143334547762416862518983569485562
09921922218427255025425688767179049460165346680498862723279178608578438382796797668145410095388378636095068006422512
52051173929848960841284886269456042419652850222106611863067442786220391949450471237137869609563643719172874677646575
73962413890865832645995813390478027590099465764078951269468398352595709825822620522489407726719478268482601476990902
64013639443745530506820349625245174939965143142980919065925093722169646151570985838741059788595977297549893016175392
84681382686838689427741559918559252459539594310499725246808459872736446958486538367362226260991246080512438843904512
44136549762780797715691435997700129616089441694868555848406353422072225828488648158456028506016842739452267467678895
25213852254995466672782398645659611635488623057745649803559363456817432411251507606947945109659609402522887971089314
5669136867228748940560101503308617928680920874760917824938589009714909675985261365549781893129784821682998948722658804857564014270

6. Podsumowanie

Po pierwsze, metoda zaproponowana w treści zadania jest nieoptymalna. Zbiega do π w sposób nadliniowy, podczas gdy pozostałe metody, z pominięciem algorytmu Monte Carlo, zbiegają co najmniej liniowo. Metoda Gaussa-Legendre'a zbiega najszybciej, jednak to metoda Chudnowsky'ch stanowi najlepszy kompromis między szybkością osiągania kolejnych liczb znaczących a zużywaną pamięcią. Co jednak najważniejsze, ludolfina została określona ze złożonością obliczeniową $O(1)$:

Następnie sporządził odlew "morza" o średnicy dziesięciu łokci, okrągłego, o wysokości pięciu łokci i o obwodzie trzydziestu łokci [7].

Ponieważ $\pi = \frac{L}{2r} = \frac{30}{10} = 3$. W takim razie wszystkie nasze obliczenia okazują się niepoprawne.



Literatura

- [1] N. D. Baruah, C. Bruce, Berndt, and Heng Huat Chan., *Ramanujan's Series for $\frac{1}{\pi}$: A Survey.*, The American Mathematical Monthly 116, no. 7 (2009): 567-87, <http://www.jstor.org/stable/40391165>.
- [2] Srinivasa Ramanujan, *Collected Papers of Srinivasa Ramanujan*, Cambridge University Press, 1st edition, 2015
- [3] Richard P. Brent, *Multiple-precision zero-finding methods and the complexity of elementary function evaluation*, 2010, <https://arxiv.org/pdf/1004.3412.pdf>.
- [4] Hans Herneave, *Gauss-Legendre Algorithm*, accessed 20.11.2022, <https://cage.ugent.be/~hvernaev/Gauss-L.html>
- [5] S. Byrne, L. Benet and D. Sanders, *Some fun with π in Julia*, accessed 20.11.2022, <https://julialang.org/blog/2017/03/piday/>
- [6] D. Kincaid, W. Cheney, *Analiza numeryczna*, Wydawnictwo Techniczno-Naukowe, Warszawa, 2006
- [7] *Biblia Tyśiąclecia*, 1 Krl 7, 23, Wydawnictwo Palottinum, Poznań, 2003