УДК: 517.965

**Решение функционального уравнения Фейгенбаума**

А. А. Полуновский

ИППИ им. Харкевича

*Email: apap2009@yandex.ru*

В данной работе были вычислены с точностью более 2000 тысяч знаков универсальная константа Фейгенбаума  и с точностью более 1200 знаков универсальная константа  . Константы Фейгенбаума — универсальные постоянные, харакетризующие бесконечный каскад бифуркаций удвоения периода при переходе к детерминированному хаосу, все чаще появляющиеся в физических приложениях. Помимо приведенных значений констант  и  были также рассмотрены основные методы численного решения системы уравнений Фейгенбаума и построены графики функции Фейгенбаума  на больших интервалах изменения аргумента.

**Ключевые слова**: уравнение Фейгенбаума, универсальные постоянные Фейгенбаума α и δ, функциональные уравнения, метод неопределенных коэффициентов, метод коллокаций, гиперболические операторы, степенные ряды

**The solution of the Feigenbaum's Functional Equation**

А. A. Polunovskiy

Institute for Information Transmission Problems of the Russian Academy of Sciences (Kharkevich Institute)

*Email: apap2009@yandex.ru*

In this paper, the universal Feigenbaum constant  was calculated with an accuracy of more than 2000 characters and the universal constant  was calculated with an accuracy of more than 1200. The Feigenbaum constants are two mathematical constants which both express ratios in a bifurcation diagram for a non-linear map, increasingly appearing in physical applications. In addition to the reduced values ​​of the constants, the main methods for the numerical solution of the system of Feigenbaum equations were also considered, and graphs of the Feigenbaum function  were constructed over large intervals of variation of the argument.

**Keywords**: Feigenbaum equation, Feigenbaum universal constants α and δ, functional equations, method of undetermined coefficients, collocation method, hyperbolic operators, power series

1. **Введение. Постановка задачи**

В последние годы большой интерес вызывает система функциональных уравнений Фейгенбаума в связи с проблемами детерминированного хаоса. Данная система имеет следующий вид [1,2,6]:

 (1)

с дополнительными условиями

, (2)

где  и  – искомые функции,  и  – универсальные постоянные Фейгенбаума [12],  . Первое уравнение в системе (1) представляет собой определение неподвижной точки оператора удвоения [6]

, (3)

в классе четных унимодальных функций. Константа  в данном случае представляет собой коэффициент перемасштабирования (rescaling) [1,2].

Правая часть второго уравнения в системе (1) представляет собой линеаризацию оператора [1,2,6]

 (4)

а само уравнение представляет собой условие на нахождение собственного значения оператора 

, (5)

где  - наибольшее по модулю собственное значение оператора  ,  - собственная функция оператора  , соответствующая собственному значению .

Предполагается, что  и  – четные аналитические функции, определенные на всей вещественной оси . Требуется найти пару функций  и пару чисел . Сейчас существует только приближенное численное решение этой системы [1,4,6]:

, (6.1)

 ,

и

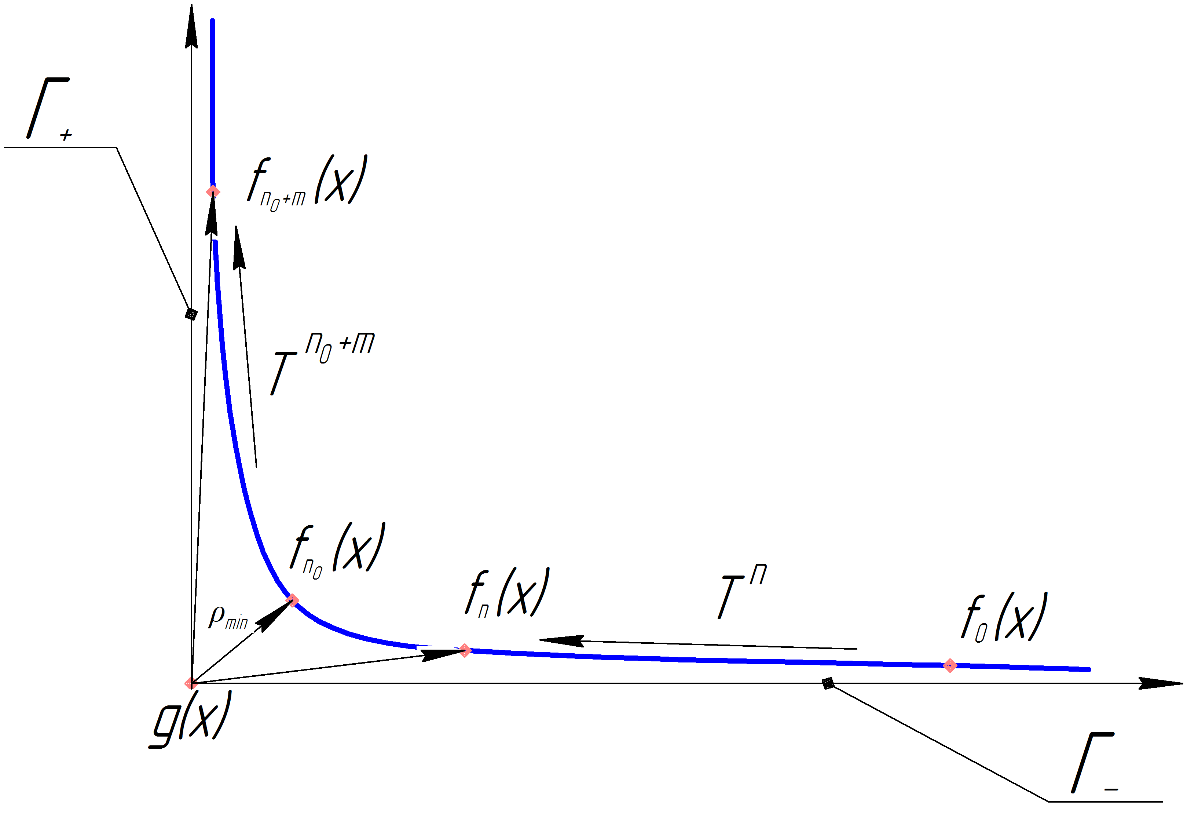
, (6.2)

.

Фейгенбаум в своих статьях [6,12] впервые в 1979 году опубликовал первые 12 знаков после запятой константы  и 13 знаков после запятой константы , а также эффективный алгоритм их вычисления. В 1991 году Кит Бриггс, используя тот же метод вычисления, получил значения данных констант с точностью 150 знаков после запятой [7]. Позже, в 1999 году , Саймон Плуфф уточнил результаты Бриггса, и вычислил 1018 знаков после запятой констант  и  [8]. В 2015 году была опубликована статья A. Molteni на arXiv.org [11], в которой был описан алгоритм, позволивший, по мнению авторов статьи, вычислить значения данных констант с точностью более 10000 знаков после запятой. Не смотря на данное утверждение, константы  и  с данной точностью не были опубликованы.

1. **Численное исследование и проблема расходимости**

В системе (1) первое уравнение не зависит от второго. Данная особенность позволяет последовательно получить сначала из первого уравнения функцию  и константу , а потом из второго функцию  и константу . Однако, функция  обладает одним неприятным свойством – она является седловой точкой оператора удвоения (3). Оператор  в этом классе функций имеет неподвижную точку - функцию  () , что фактически отражается в записи первого уравнения системы (1). Выбирая начальную функцию  ( например параболу  ), путем последовательных итераций  получаем приближение  к неподвижной точке . При первых итерациях  сходится к функции , при последующих итерациях  расходится от  (рис.1). Такое необычное явление расходимости следует из свойства гиперболичности оператора  - существования у оператора  одномерного растягивающего пространства  и коодномерного сжимающегося пространства  (рис.1) [1,3,6]. При действии оператора  наше функциональное пространство одновременно растягивается вдоль многообразия  и сжимается вдоль многообразия  [2,3].



**Рис.1** Поведение итераций отображения  вблизи гиперболической точки 

Данная особенность сильно замедляет скорость сходимости методов, основанных на итерациях унимодальных функций [9,10], что делает невозможным вычисление с большой точностью констант  и  исходя из их определения [1,7]. В следствии этого, единственный на данный момент эффективный способ вычисления постоянных Фейгенбаума является численный расчет системы уравнений (1).

1. **Метод коллокаций, применительно для решения уравнения Фейгенбаума**

Основным численным методом решения системы (1) является метод коллокаций, впервые примененный Фейгенбаумом для решения данных уравнений [6,7]. Этот метод основан на представлении функций  и  в виде некоторой суммы базисных функций

 ,  , (7)

где  ,  - набор базисных функций. Обычно, в силу аналитичности функций  и  [6], данным базисом является последовательность степеней [6,7,8], а функции  и  представлялись в виде степенных рядов

 ,  . (8)

Иногда использовались для расчетов и другие базисы, например в статье [11] разложение велось по полиномам Чебышева.

Зафиксировав некоторый базис функций, выберем теперь набор из  точек , распределенный на полуинтервале . Подставив в систему (1) представление функций  и  в виде (7), и рассмотрев полученные равенства в точках , получим систему из  нелинейных уравнений

 (9)

где , на коэффициенты  разложений (7). Добавляя к системе (9) соотношения (10), мы получаем замкнутую систему на коэффициенты  и .

 ,  (10)

Численно решая систему (9) (например, методом Ньютона), получаем приближенные значения  констант  и . Из вычислительной практики получено, что в случае выбора разложений (8), мы наблюдаем сходимость данного метода [6,7]

, (11)

где .

Решение системы (9) можно упростить, решая последовательно сначала систему уравнений, определяющих коэффициенты разложения функции , а затем систему уравнений, определяющих коэффициенты разложения функции . Кроме того, для вычисления константы  саму функцию  можно даже не находить. В силу того, что константа  является собственным значением оператора (4), достаточно найти максимальное по модулю собственное число матрицы, представляющей собой правую часть системы уравнений (9) на коэффициенты функции . Данное собственное значение и будет как раз приближением  к константе .

Система (9) решается многомерным методом Ньютона [5]. Одна из основных проблем в многомерном методе Ньютона является проблема обращения матрицы Якоби (указать). Соответственно, основная часть вычислительного времени расчета была затрачена на обращение матрицы Якоби методом Гаусса. Другая немаловажная проблема заключается в высокой скорости сходимости метода Ньютона — около двух. Это означает, что нам нужно хранить в памяти в два раза больше значащих знаков по сравнению с тем результатом, который хотели достичь. В связи с этим вопрос о загрузки памяти в данной задаче является не менее важным, нежели процессорное время.

1. **Метод неопределенных коэффициентов, применительно для решения уравнения Фейгенбаума**

Другим подходом численного решения системы (1) является метод неопределенных коэффициентов [4,6]. Данный метод также основывается на представлении решения в виде степенных рядов (8). В отличии от метода коллокаций, мы не вычисляем значения функций в конкретных точках, а рассматриваем систему уравнений, полученных после подстановки выражений (8) в систему (1), приравнивая получаемые коэффициенты при соответствующих степенях в левой и правой части равенств. После всех необходимых преобразований, мы получаем следующую систему уравнений на коэффициенты разложения функции 

, (12)

и, соответственно, систему уравнений на коэффициенты разложения функции 

 . (13)

Добавляя к системам (12) и (13) соотношения (10), мы получаем замкнутую систему на коэффициенты  и . Решая данную систему итерационными методами, мы также, как и в случае метода коллокаций, получаем некоторое приближение  к константам .

Основная сложность данного метода заключается в формировании функций  и , являющиеся многочленами от переменных  и . В случае большого числа  данные функции становятся громоздкими, и требуют специальных подходов к их вычислению. Однако, системы (12) и (13), по сравнению с системой (9), имеют более близкие свойства к исходной системе (1). Можно сказать , что системы (12) и (13) являются проекцией на конечномерное пространство  системы (1), где функции  и  заменяются конечномерными векторами  и .

Наиболее простым способом системы (12) и (13) решаются, используя метод последовательных приближений (МПП) [5]. В этом случае системы (12) и (13) заменяются реккурентной зависимостью

, (14.1)

, (14.2)

где , c дополнительными условиями (10). При выборе подходящего начального приближения, данные итерационные последовательности (14.1), (14.2) стремятся к решениям систем (12) и (13) со скоростью сходимости, близкой к единице.

1. **Численные результаты**

Численный расчет универсальных постоянных  и  проводился методом коллокаций и уточнялся методом неопределенных коэффициентов. Соответственно, решались две разные системы уравнений, определяющие коэффициенты  и константы . Результаты расчета показали, что оба метода дали одинаковый результат, что дополнительно подтверждает точность полученных результатов.

Расчет системы (9) проводился на сетке, определяемой следующей зависимостью -  , . Число точек-коллокаций было выбрано равным  для расчета констант и  для проверки точности полученных знаков. Программа для расчета универсальных констант  была написана на C++ с использованием библиотеки высокоточных вычислений GNU MPFR library [13] и стандарта openMP. Расчет проводился с использованием 14 Gb оперативной памяти (DDR4) на 6 ядерном процессоре intel i7-8700k. Общее время расчета констант  в случае 2100 точек-коллокаций заняло 1 сутки.

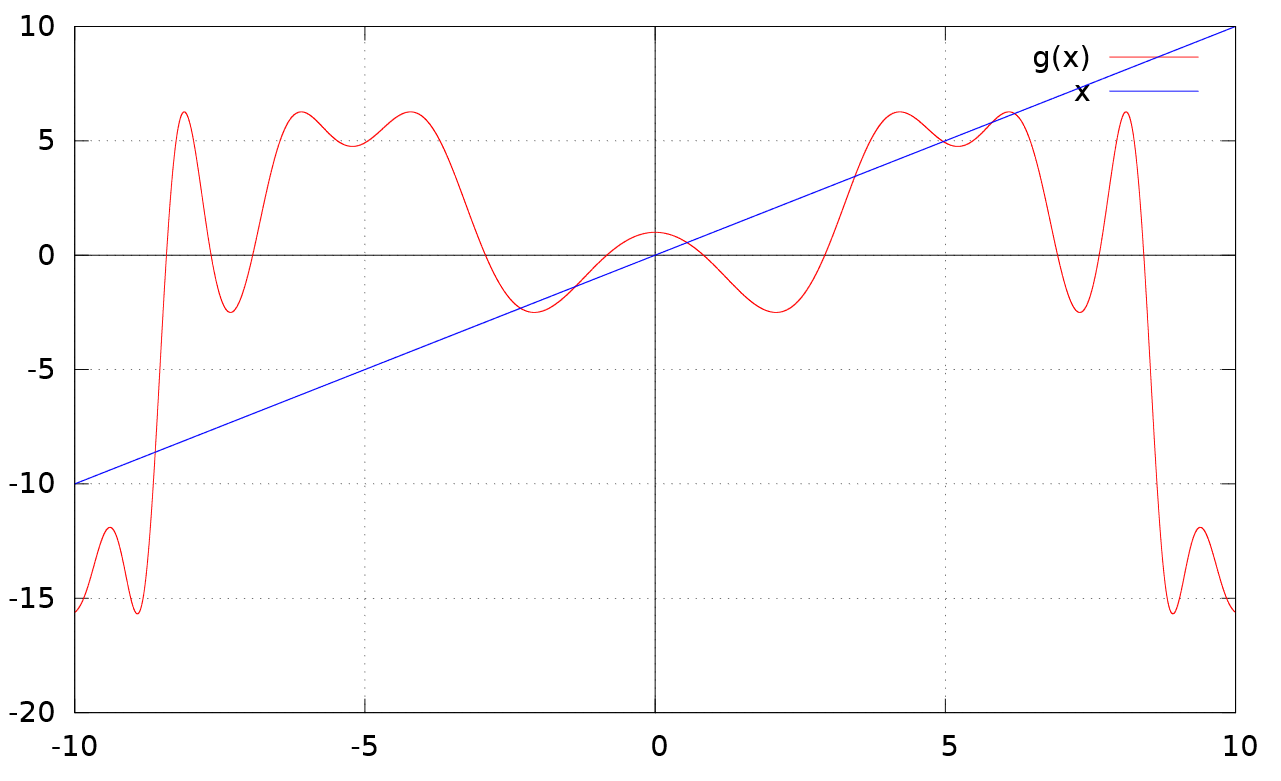
В результате расчета было получено 2852 знака после запятой числа  и 1277 знаков после запятой числа 

= 2.50290787509589282228390287321821578638127137672714997733619205677923546317959020670329964974643383412959523186999585472394218237778544517927286331499337257811216359487950374478126099738059867123971173732892766540440103066983138346000941393223644906578899512205843172507873377463087853424285351988587500042358246918740820428170090171482305182162161941319985606612938274264970984408447010080545496779367608881264464068851815527093240075425064971570470475419932831783645332562415378693957125097066387979492654623137674591890981311675243422111013091312783716095115834123084150371649970202246812196440812166865274580430262457825610671501385218216449532543349873487413352795815351016583605455763513276501810781194836945957485023739823545262563277947539726990201289151664579394201989202488033940516996865514944773965338769797412323540617819896112494095990353128997733611849847377946108428833293833903950900891408635152562680338141466927991331074334970514354520134464342647520016213846107299226419943327729189777690538025968518850841613864279936834741390166705544353112159412076097447476975360415562684762316863202036958955323302591969942848633937659618260681047820499176267330237410190585259508031282402710592499993884498730466113597295106650513245319934436198567306784849235809205523704253031442693729196574033848547282599137175415828319291662030849437529190242970645159431832597653248819510930031308222068457541350628357528419520349745892014786408007778222864734293253286487082017289870335956148587603058294765749795007685996283001331698230618274215756544271889584049779070749215052863008192239338223670789370549815234149424093095382024189917656108663349195188827701571446844680703356348092938515560071929854269973847625695912052273616189207044162966822751126069943803094537261000357524601395837218537256178165146708417297398598406872592381291993509584854066218363979292126454131799585751022981936551098662441566798762787075489215597600579682260572455179176040845371381479108344598320681539872941485146629411417058312639377907126978315886082270600564957374356937181468036309526461848199005414043320608665404468636297028102220910619540288698443842340944582428321937688302448340396701919834033939733314004987477777858459988856548506710723647412825261046970014961811343672028087410939053241020006460150881223186667819269607148893443672144564744821762971861519129542270610697683233714727863995560234771344485408531503679799747683901060114357643484894101666276676295196774877277417262929423829600205550958140829523483328739540589112662208814425826839839852024014490033771479394167756788689687378385451441688715474253931542330600679648245724259642063947983400689987655326248435937845837834536446049160687128349299369970284568401377016855476946909112640466150557154553122762648270685040725794450691522805944174672106930456278945606512048959627448493205381082354054...,

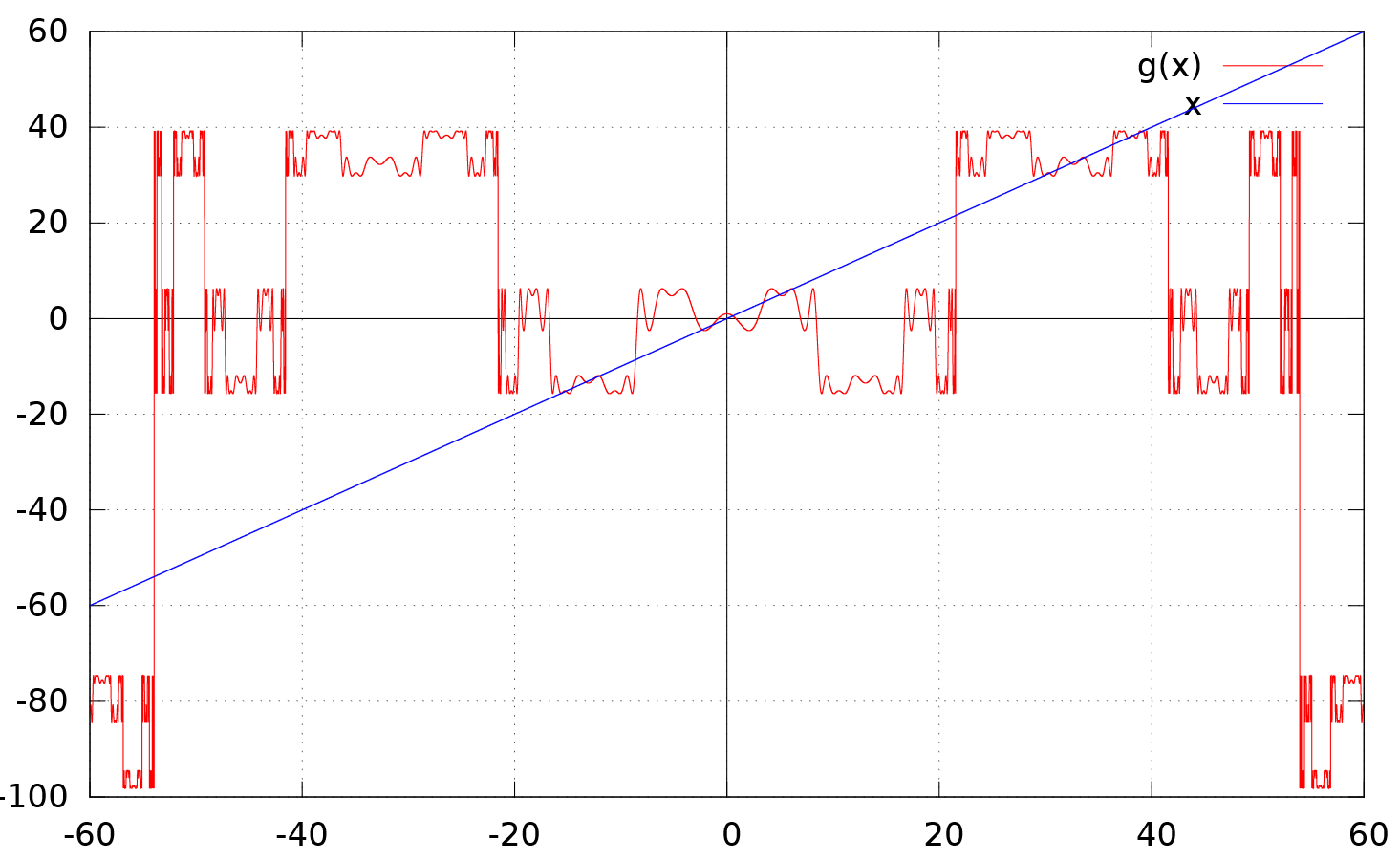
= 4.66920160910299067185320382046620161725818557747576863274565134300413433021131473713868974402394801381716598485518981513440862714202793252231244298889089085994493546323671341153248171421994745564436582379320200956105833057545861765222207038541064674949428498145339172620056875566595233987560382563722564800409510712838906118447027758542854198011134401750024285853824983357155220522360872502916788603626745272133990571316068753450834339344461037063094520191158769724322735898389037949462572512890979489867683346116268891165631234744605751795391220455624728070952021981990945585819461368774456173960741156140742437544354992048691809826486523684387027996490173977934251347238087371362116018601281861020563818183540975984779641739003289361714321598782407897766143913957640377605371190969320669983619842889818370032294120302106557432955503888458497370347275321219257069584140746618419819610061296401614877129444159014054679418001981332533785924933658830704599999383754117265635530168625290322108623205506345106793990233416751176315298404340058573927197651891997600721935124568600526428574371151276314636065699549375229309422609651800806943524617318607101604035227383471263260294144252525370124578953421898372981465669318649049509872669725924698692208505234597334972306170991843524072… .

Саймон Плуфф в 1999 году проводил свой расчет [8] в течении 3 дней в случае 700 точек-коллокаций и 3.5 дней в случае 720 точек-коллокаций. Для проведения расчета он использовал 400 МБ оперативной памяти и процессор DeCalpha c тактовой частотой 433MHz. Его результатом расчета является 1018 вычисленных знаков чисел . Все знаки констант , полученные Саймоном Плуффом, были подтверждены в данном расчете.

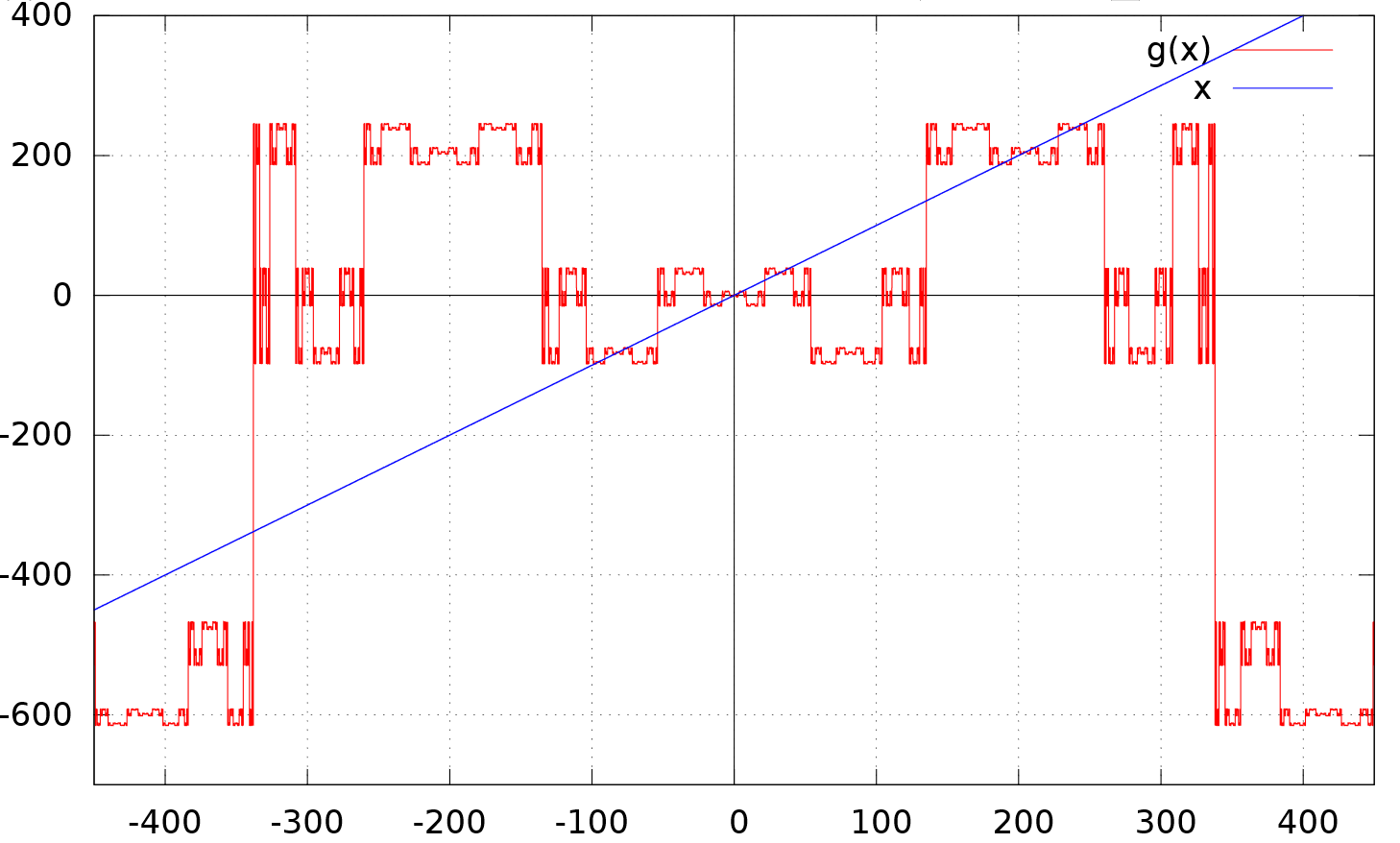
По мимо полученных значений констант , был построен график функции  на различных интервалах значения аргумента до [-500,500]. Данный график был ранее построен Фейгенбаумом [2] только на интервале [-15,15].



**Рис. 2** График функции  на интервале [-10,10]

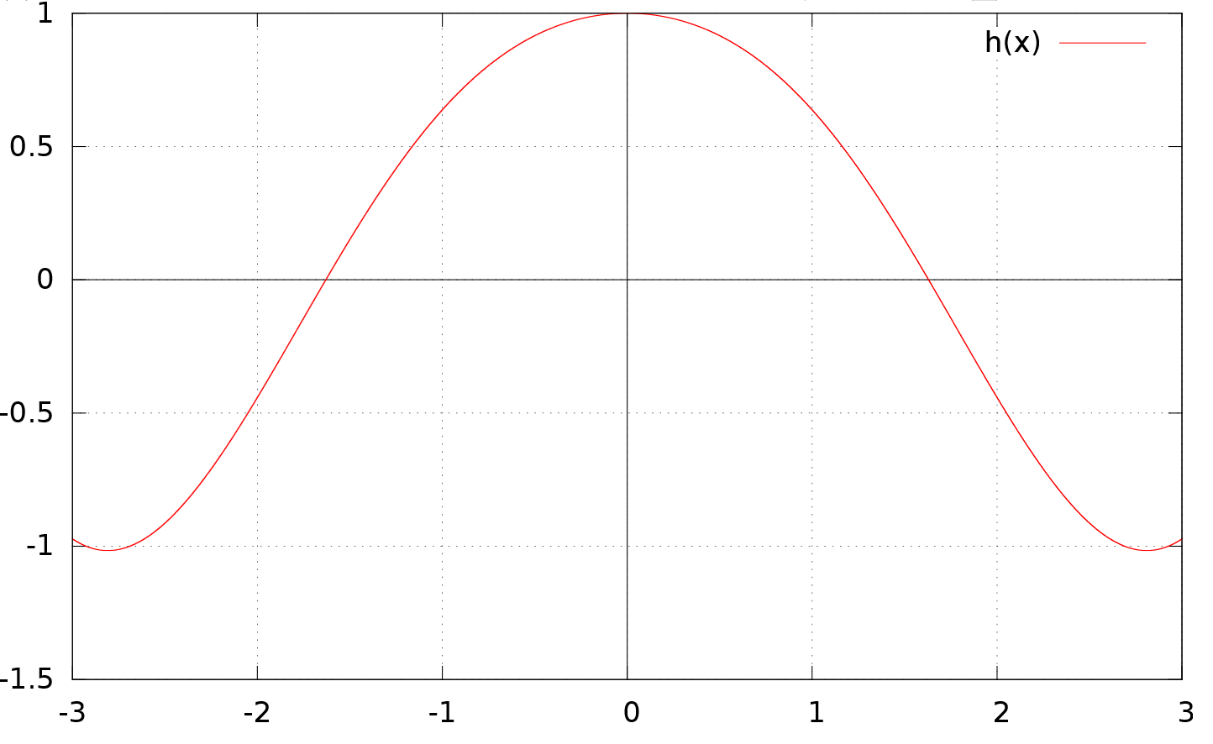


**Рис. 3** График функции  на интервале [-60,60]



**Рис. 4** График функции  на интервале [-500,500]

На рисунках 2, 3 и 4 видно, что функция  обладает ярко выраженным свойством самоподобия. По мимо графика функции , был построен график функции  на интервале [-3,3]



**Рис. 5** График функции  на интервале [-3,3]

1. **Заключение**

В данной статье были вычислены 2852 знака после запятой константы и 1277 знаков константы , описаны основные численные методы решения системы уравнений Фейгенбаума (1), кратко описана история расчета констант Фейгенбаума  и были построены графики функций  и . Последний опубликованный расчет данных констант был произведен в 1999 году, и на тот момент было получено 1018 знаков констант [8]. В 2015 году была опубликована работа на arXiv.org [11], в которой утверждается, что авторами было получено более 10000 знаков констант . Несмотря на это утверждение, данные результаты так и не были опубликованы.

1. **Использованная литература**

1. *Г*. *Шустер.* Детерминированный хаос. - М.: Мир, 1988, 240 с.

2. *М.* *Фейгенбаум.* Универсальность в поведении нелинейных систем // Успехи Физических Наук,1983 г.

3. *В.И. Арнольд.* Геометрические методы в теории обыкновенных дифференциальных уравнений. - Ижевск: ИРТ, 1999, - 400 с

4. *С.П. Кузнецов*. Динамический хаос.- изд. 2, М.: Физматлит, 2006

5. *О. Джеймс, Р. Вернер.* Итерационные методы решения нелинейных систем уравнений со многими неизвестными.- М.: Мир, 1975.

6. *Mitchell J. Feigcnbaum.* The Universal Metric Properties of Nonlinear Transformations // Journal of Statistical Physics, Vol. 21, No. 6, 1979

7. *K. Briggs.* A precise calculation of the feugenbaum constants. Mathematics of computation // Vol. 57, No. 195, 1991.

8. *D. Broadhurst.* Feigenbaum constants to 1018 decimal places // <http://www.plouffe.fr/simon/constants/feigenbaum.txt> , March 1999.

9. *K. M. Briggs*. How to calculate the Feigenbaum constants on your PC // Austral. Math. Soc. Gaz. 16 (1989), 89–92.

10. *J. B. McGuire and C. J. Thompson.* Asymptotic properties of sequences of iterates of nonlinear transformations // J. Statist. Phys. 27 (1982), 183-200.

11. *A. Molteni*. An efficient method for the computation of the feigembaum constants to high precision // arXiv.org. 15.12.2015. URL: <https://arxiv.org/pdf/1602.02357.pdf>

12. *M. J. Feigenbaum.* Quantitative universality for a class of nonlinear transformations // J. Stat. Phys. 19 (1978), no. 1, 25–52.

13. *L. Fousse, G. Hanrot, V. Lef`evre, P. P´elissier, and P. Zimmermann.* MPFR: A multipleprecision binary floating-point library with correct rounding // ACM Trans. Math. Software 33 (2007), no. 2, 13

Сведения об авторе:

**Андрей Андреевич Полуновский (A. A. Polunovskiy),**

сотрудник ИППИ им. Харкевича

*E-mail:* apap2009@yandex.ru