

P5-2002-228

Любомир Александров¹, Д. Б. Баранов², Пламен Йотов³

ПОЛИНОМИАЛЬНЫЕ СПЛАЙНЫ,
ТОЧНО ВОСПРОИЗВОДЯЩИЕ ПРОСТЫЕ ЧИСЛА

Направлено в журнал «ACM Transactions on Mathematical Software»

¹email: alexandr@thsun1.jinr.ru

²email: baranov@thsun1.jinr.ru

³email: pyotov@inrne.bas.bg

1. Введение

В работе найдены действительные функции, воспроизводящие значения взаимно обратных арифметических функций $p(n) : N \rightarrow P$ (*простое число с номером n*) и $p^{-1}(p) : P \rightarrow N$ (*номер простого числа p*) [1].

Найденные функции применены для создания стандартных подпрограмм, для вычисления $p(x)$, $\frac{dp(x)}{dx}$ на $1 \leq x < \infty$, и $p^{-1}(x)$ на $2 \leq x < \infty$. Указанные подпрограммы можно использовать для численного решения различных задач над множеством простых чисел P , в том числе и для приближенного решения диофантовых уравнений.

Основная идея состоит в нахождении дифференцируемой функции, содержащей значения функции $p(n)$, которая допускала бы построение при помощи метода Ньютона значений обратной функции $p^{-1}(p)$ [2].

На искомую функцию $p(x)$, $1 \leq x < \infty$, наложены следующие условия:

- а) $p(x)$ воспроизводит все простые числа до заданного номера;
- б) существует положительная производная $\frac{dp(x)}{dx}$;
- в) существует обратная функция $p^{-1}(x) : [2, \infty) \rightarrow [1, \infty)$.

Как известно, не существуют полиномы, которые могут воспроизвести все простые числа или только простые числа. Однако в нашей работе показано, что существуют полиномиальные сплайны, воспроизводящие все простые числа.

Сплайн из полиномов с целочисленными коэффициентами назовем *арифметическим*.

В работе рассмотрены два арифметических сплайна, кубический и параболический.

Найденные сплайны не аппроксимируют ряд простых чисел. Простые числа заложены в конструкцию сплайнов, и это обеспечивает их точную воспроизводимость. Эти сплайны приводят к явно решаемым линейным системам уравнений, а их коэффициенты являются основными арифметическими функциями.

Обратная функция $p^{-1}(x)$ является дифференцируемым аналогом теоретико-числовой функции $\pi(x)$ (*число простых чисел, не превосходящих x*).

Функция $p^{-1}(x)$ сравнивается с выражением Римана для $\pi(x)$ через нули ζ -функции.

2. Кубический сплайн

Рассмотрим сплайн

$$S_{cub}(x) = \begin{cases} x + 1, & 1 \leq x \leq 1.5, \\ c_i(x), & i - 0.5 \leq x \leq i + 0.5, \quad i = 2, 3, \dots, \\ c_i(x) = c_{i+1}(x), & x = i + 0.5, \quad i = 1, 2, \dots, \\ \frac{dc_i(x)}{dx} = \frac{dc_{i+1}(x)}{dx}, & x = i + 0.5, \quad i = 1, 2, \dots, \end{cases}$$

где

$$c_i(x) = 2 \left(a_i(x - i - 0.5)^2 + b_i(x - i - 0.5) + \frac{p(i) + p(i + 1)}{2} \right) (x - i) - 2p(i)(x - i - 0.5).$$

Точная воспроизводимость простых чисел следует из тождества

$$c_i(i) \equiv p(i), \quad i = 1, 2, \dots \quad (1)$$

На узлах сшивания сплайна также выполняется тождество

$$c_i(i + 0.5) \equiv \frac{1}{2}(p(i) + p(i + 1)), \quad i = 1, 2, \dots, \quad (2)$$

что вносит в сплайн *дополнительную информацию о поведении ряда простых чисел.*

Существует *единственный кубический сплайн вида* $S_{cub}(x)$ с коэффициентами

$$\left. \begin{aligned} a_i &= \frac{1}{2}(p(i + 1) - p(i - 1)) - 1, \\ b_i &= p(i + 1) - p(i) - 1, \end{aligned} \right\} \quad i = 2, 3, \dots$$

Условием положительности производной

$$\frac{dc_i(x)}{dx} = 2(2a_i(x - i - 0.5) + b_i)(x - i) + c_i(x) - 2p(i)$$

является отрицательное значение ее дискриминанта

$$d_i = 16(p(i))^2 - 16(p(i - 1) + p(i + 1))p(i) + (p(i - 1))^2 + (p(i + 1))^2 + 14p(i - 1)p(i + 1) + 12.$$

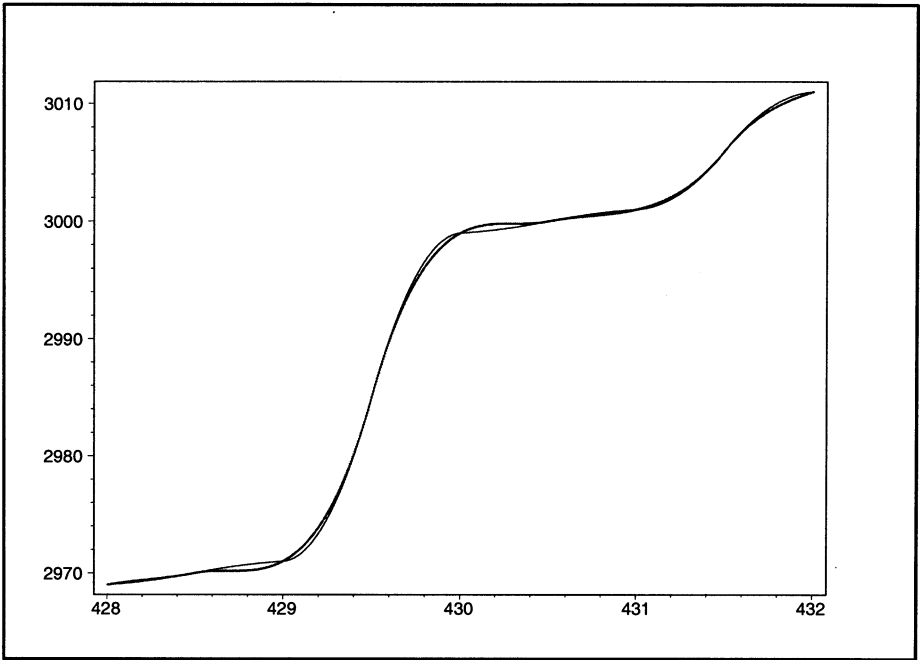


Рис. 1: Сравнение прямых функций $p(x)$ для сплайнов S_{cub} и S_{quad} : толстая линия отвечает S_{cub} , а тонкая – S_{quad}

Неравенство $d_i < 0$ приводит к следующему условию на последовательные тройки из простых чисел $p(i-1)$, $p(i)$, $p(i+1)$:

$$\frac{1}{2}(p(i-1) + p(i+1)) - \frac{1}{4}\sqrt{t_i} < p(i) < \frac{1}{2}(p(i-1) + p(i+1)) + \frac{1}{4}\sqrt{t_i}, \quad (3)$$

где

$$t_i = (p(i-1) + p(i+1))^2 - 4 > 0, \quad i = 1, 2, \dots$$

Условие (3) нарушается в случаях:

$$p(i) - p(i-1) = \Delta_1, \quad p(i+1) - p(i) \geq \Delta_2; \quad (4)$$

$$p(i) - p(i-1) \geq \Delta_2, \quad p(i+1) - p(i) = \Delta_1,$$

$$\text{где } \Delta_1 = 2, \quad \Delta_2 = 28.$$

В первых 1000 простых числах только следующие 5 троек нарушают

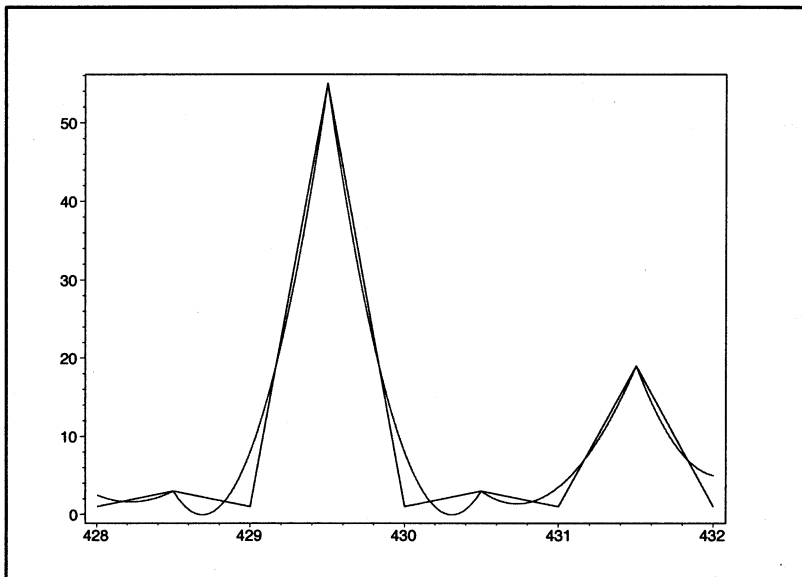


Рис. 2: Сравнение производных $S'_{cub}(x)$ и $S'_{quad}(x)$

условия (3):

$$\begin{aligned}
 &(2969, 2971, 2999), \\
 &(2971, 2999, 3001), \\
 &(3271, 3299, 3301), \\
 &(6917, 6947, 6949), \\
 &(7757, 7759, 7789).
 \end{aligned} \tag{5}$$

Кроме троек типа (4) (с парой близнецов), условие (3) нарушает также тройки вида (4) при следующих значениях Δ_1, Δ_2 :

$$\begin{aligned}
 \Delta_1 &= 4, & \Delta_2 &= 56, \\
 \Delta_1 &= 6, & \Delta_2 &= 84, \\
 \Delta_1 &= 8, & \Delta_2 &= 114, \\
 && &\text{и т.д.}
 \end{aligned}$$

Несмотря на случаи нарушения условия (3), сплайн $S_{cub}(x)$ удобен для создания подпрограмм $p(x)$, $\frac{dp(x)}{dx}$ и $p^{-1}(x)$, так как обратная функция $p^{-1}(x)$ существует в окрестности любого простого числа.

3. Параболический сплайн

Пусть заданы пары парабол

$$q_i(x) = \left\{ \begin{array}{l} q_i^{(l)}(x), \quad i - 0.5 \leq x \leq i, \quad i = 2, 3, \dots, \\ q_i^{(r)}(x), \quad i \leq x \leq i + 0.5, \quad i = 2, 3, \dots, \\ \left. \begin{array}{l} q_i^{(l)}(x) = q_i^{(r)}(x) \\ \frac{dq_i^{(l)}(x)}{dx} = \frac{dq_i^{(r)}(x)}{dx} \end{array} \right\}, \quad x = 2, 3, \dots; \text{внутреннее сшивание,}$$

где

$$\begin{aligned} q_i^{(l)}(x) &= -2a_{i-1}(x-i)^2 + x - i + p(i), \\ q_i^{(r)}(x) &= 2a_i(x-i-0.5)^2 + (2a_i+1)(x-i-0.5) + \frac{p(i)+p(i+1)}{2}, \\ a_i &= p(i+1) - p(i) - 1. \end{aligned}$$

Параболический сплайн

$$S_{quad}(x) = \left\{ \begin{array}{l} x+1, \quad 1 \leq x \leq 1.5, \text{ начальный полином,} \\ q_i(x), \quad i - 0.5 \leq x \leq i + 0.5, \quad i = 2, 3, \dots, \\ \left. \begin{array}{l} q_i(x) = q_{i+1}(x) \\ \frac{dq_i(x)}{dx} = \frac{dq_{i+1}(x)}{dx} \end{array} \right\}, \quad \begin{array}{l} x = i + 0.5, \quad i = 1, 2, \dots; \\ \text{внешнее сшивание} \end{array}$$

решает поставленную задачу лучше, чем сплайн S_{cub} .

Сплайн S_{quad} обладает свойствами:

1) имеют место тождества, аналогичные (1), (2),

$$\begin{aligned} q_i(i) &\equiv p(i), \\ q_i(i+0.5) &\equiv \frac{1}{2}(p(i) + p(i+1)), \quad i = 1, 2, \dots; \end{aligned}$$

2) производные

$$\begin{aligned} \frac{dq_i^{(l)}(x)}{dx} &= -4a_i(x-i) + 1, \\ \frac{dq_i^{(r)}(x)}{dx} &= 4a_i(x-i-0.5) + 2a_i + 1 \end{aligned}$$

приобретают минимальное значение +1 в точках внутреннего сшивания (они же точки интерполяции сплайна) и максимальные значения в точках внешнего сшивания, где совпадают с производной сплайна S_{cub} .

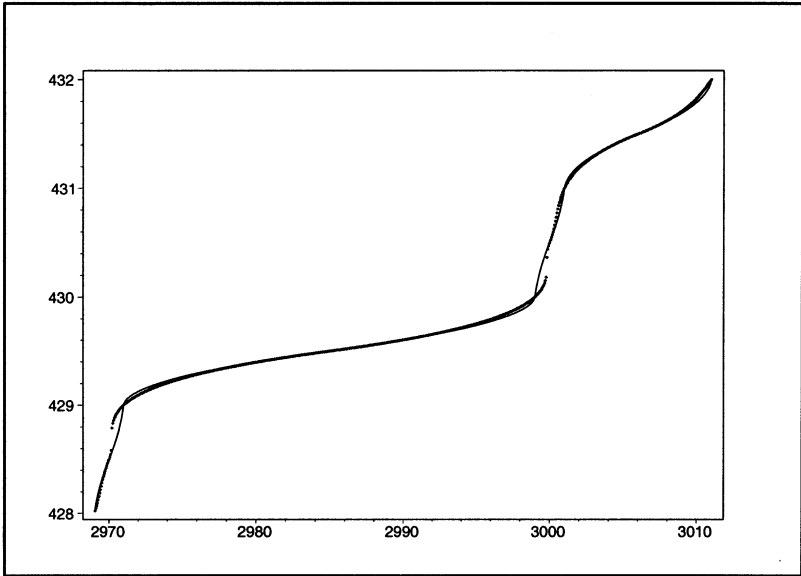


Рис. 3: Сравнение обратных функций $p^{-1}(x)$ для сплайнов S_{cub} и S_{quad} : толстая линия с двумя разрывами отвечает S_{cub} , а тонкая – S_{quad}

Положительность производных показывает, что сплайн S_{quad} монотонно возрастает на полуоси $[1, \infty)$. Это означает, что существует функция $S_{quad}^{-1}(x)$, обратная к функции $S_{quad}(x)$, определенная на полуоси $[2, \infty)$, и, таким образом, сплайн S_{quad} удовлетворяет поставленным условиям а), б), в). Кроме того, функция $S_{quad}^{-1}(x)$ дифференцируемая на $(2, \infty)$.

Арифметический сплайн S_{quad} , обладающий указанными свойствами, единствен.

Специально отметим, что базисный коэффициент a_i сплайна S_{quad} есть основная арифметическая функция – число составных чисел, лежащих внутри интервала $(p(i), p(i+1))$, что придает сплайну определённый физический смысл.

На рисунках 1, 2 и 3 представлено сравнение сплайнов S_{cub} и S_{quad} . Образ интервала $[428, 432]$ (аргумент функций $p(x)$ и $dp(x)$) содержит первые две тройки простых чисел (5), которые нарушают положительность производной $S'_{cub}(x)$ (условие (3)).

На рисунке 3 видны интервалы, в которых не существует обратная функция $p^{-1}(x)$ для сплайна S_{cub} .

Рисунок 2 иллюстрирует свойства производных сплайнов: точки до-

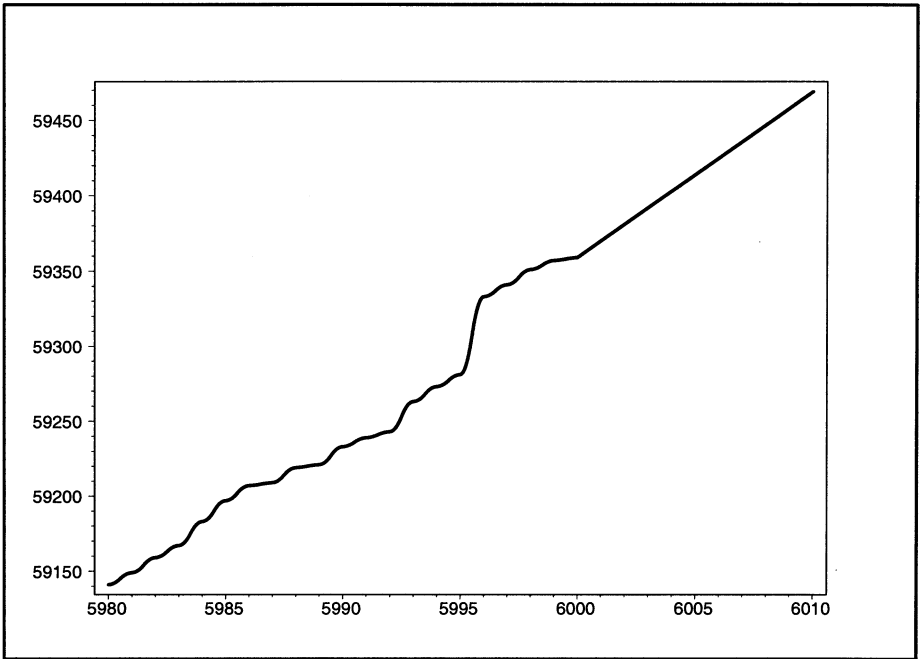


Рис. 4: Сшивание сплайна $S_{quad}(x)$ с асимптотой $\tilde{p}(x)$ в точке $x=6000$

стижения минимальных и максимальных значений и равенства производных в точках сшивания.

4. О стандартных подпрограммах $p(x)$, $\frac{p(x)}{dx}$ и $p^{-1}(x)$

Оба сплайна S_{cub} и S_{quad} использовались для создания фортрановых функций $p(x)$, $pd(x)$ $\left(\frac{dp(x)}{x}\right)$ и $p_{-}(x)$ (здесь $p_{-}(x)$ обозначает $p^{-1}(x)$). В целях удобства применения функции $p(x)$ и $dp(x)$ продолжены к $+\infty$ асимптотой [3], стр. 640)

$$\tilde{p}(x) = x \left(\ln x + \ln \ln x + \frac{\ln \ln x - 2}{\ln x} - \frac{(\ln \ln x)^2 / 2 - 3 \ln \ln x + 5.5}{(\ln x)^2} - 1 \right). \quad (6)$$

Программы для S_{quad} (пакет $S_{quad}PP_{-}$) приведены в Приложении 1.

Для получения простых чисел используются две альтернативные подпрограммы *function eratostenes(n)* и *function primes(n)*. Первая программа делает это через генерацию простых чисел до заданного n , а вторая через чтение n простых чисел из наперед заданного 6-колонного файла *primes*. В этих программах также проводится автоматическое сшивание сплайна и его производной с приближением (6) и его производной.

Для построения обратной функции $p^{-1}(x)$ в программе $p_-(x)$ используется авторегуляризованный вариант метода Ньютона ([4], стр. 43)

$$\left. \begin{aligned} y_0, \quad \varepsilon_0 > 0, \quad y_{k+1} &= y_k - \frac{p(y_k) - x}{dp(y_k) + \varepsilon_k}, \quad k = 0, 1, 2, \dots, \\ \varepsilon_k &= \frac{1}{2} \left(\sqrt{(dp(y_k))^2 + 4N|p(y_k) - x|} - dp(y_k) \right), \\ N &= (\varepsilon_0^2 + \varepsilon_0 dp(y_0)) / |p(y_0) - x|, \end{aligned} \right\} \quad (7)$$

который при нескольких комбинациях начального значения $y_0 = L(x)$ и начального регуляризатора ε_0 (эти комбинации видны в начале тела программы $p_-(x)$) обеспечивает построение обратной функции $p^{-1}(x)$ на интервале $[2, 7.10^7]$.

Исследован способ задания начального приближения y_0 со значениями

$$\begin{aligned} \frac{x}{\ln x}, \quad L(x) &= \int_2^x \frac{dt}{\ln t}, \\ R(x) &= \sum_{n=1}^{\infty} \frac{\mu(n)}{n} f(x^{1/n}), \end{aligned} \quad (8)$$

где $\mu(x)$ – функция Мебиуса, $f(x) = L(x)$ ([5], стр. 53–54). Найдено, что метод (7) лучше сочетается с $y_0 = L(x)$.

На рисунке 4 показано сшивание функции $p(x)$ с асимптотой (6).

Функции $p(x)$, $\frac{dp(x)}{dx}$, $p^{-1}(x)$ продолжены на $(-\infty, -0.5]$ как нечетные, с использованием для интервала $[-0.5, +0.5]$ связывающего полинома $8x^3 + x$. В настоящем варианте программного пакета используется естественное продолжение указанных функций к $-\infty$, основанное на значениях начального полинома $x + 1$.

Программный пакет $S_{quad}PP_$ написан на языке Fortran77. Он непосредственно применим к Compaq- и MS- Fortran и просто адаптируется к другим Fortran-версиям. Существуют версии пакета на Fortran90 с q-точностью (32 десятичных знака), а также версия, работающая на всей оси $(-\infty, +\infty)$. Предстоит создание версии с произвольной точностью при помощи программ D. H. Bailey [2].

5. О применении функций $p(x)$ и $p^{-1}(x)$

5.1. На основе функций $p(x)$ и $p^{-1}(x)$ вводим новые функции $\sin(p(x))$, $\cos(p(x))$, $\tan(p(x))$, $e^{-p(x)}$ и $\ln(p(x))$, которые можно применять в теоретической физике в случаях, когда есть необходимость использовать специфический характер неасимптотического распределения простых чисел (см., например, [6] и [7]).

5.2. Функции $p(x)$ и $p^{-1}(x)$ созданы для решения W_3 -system и для проверки PNSW-гипотезы о существовании паутин из простых чисел, сформулированной в работе [1].

5.3. Диофантовы уравнения можно решать на основе следующего приближенного метода: к диофантовому уравнению

$$f(x_1, x_2, \dots, x_n) = 0 \quad (9)$$

добавим уравнение

$$\sin^2(\pi x_1) + \sin^2(\pi x_2) + \dots + \sin^2(\pi x_n) = 0, \quad (10)$$

или уравнение

$$\sin^2(\pi p^{-1}(x_1)) + \sin^2(\pi p^{-1}(x_2)) + \dots + \sin^2(\pi p^{-1}(x_n)) = 0. \quad (11)$$

Решая системы (9, 10) и (9, 11), мы находим реальные приближения к целым натуральным или простым числам, являющимся решениями уравнения (9).

Указанные системы решаются итеративными методами, которые работают в случае вырождения производной в решении (см., например, [8]).

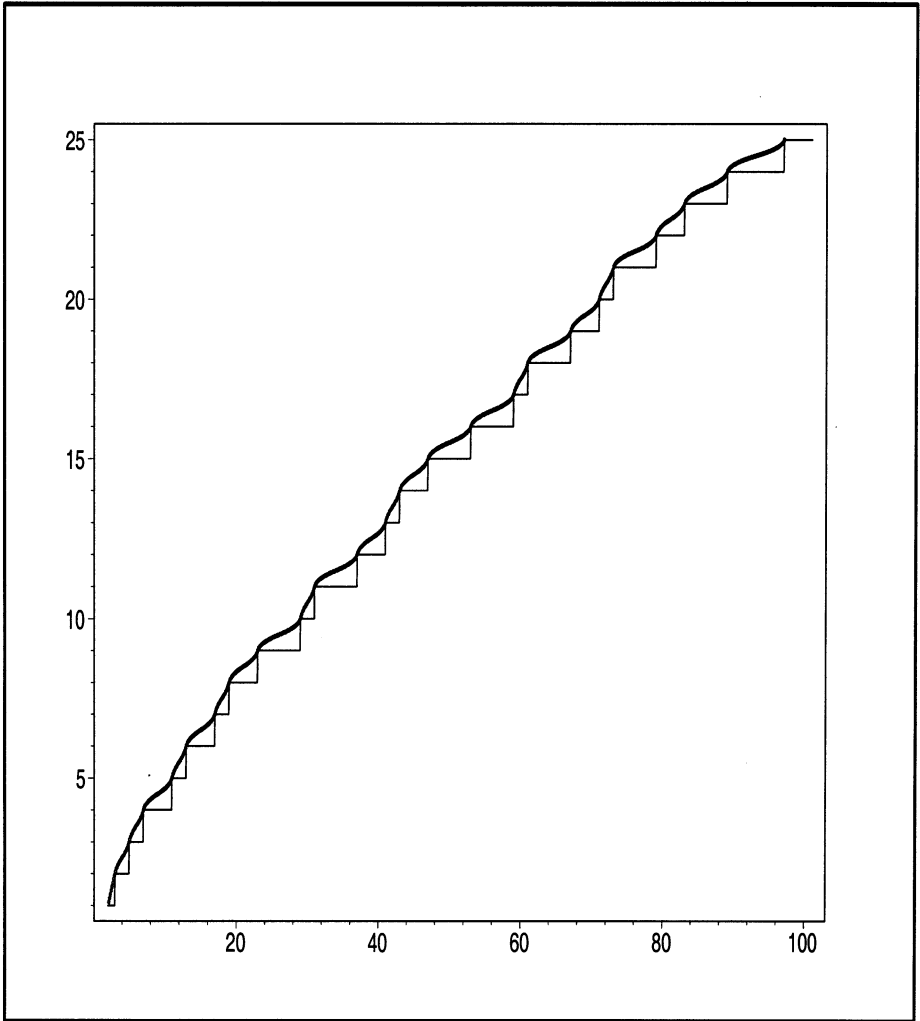


Рис. 5: Сравнение дифференцируемой функции $p^{-1}(x)$ (толстая линия) с функцией Римана-фон Мангольда $\pi(x)$ (ступенчатая линия)

5.4. На рисунке 5 обратная функция $p^{-1}(x)$ сравнивается с гауссовской ступенчатой функцией $\pi(x)$ и с непрерывной ступенчатой функцией Римана–фон Мангольдта $\pi(x)$, которая получается из (8) заменой

$$f(x) = L(x) - \sum_{\rho} L(e^{\rho \ln x}) + \int_x^{\infty} \frac{dt}{(t^2 - 1)t \ln t} - \ln 2,$$

где $\{\rho\}$ суть комплексные решения уравнения

$$\zeta(s) = 0 \tag{12}$$

вида $\rho_n = \frac{1}{2} + it_n$, $n = 1, 2, \dots$, и где $\zeta(s)$ есть функция Римана (см., например, [5], стр. 48).

В нашем случае функция $\pi(x)$ выражается через $p^{-1}(x)$ формулой

$$\pi(x) = \lfloor p^{-1}(x) \rfloor. \tag{13}$$

Сравнение показывает, что функция $p^{-1}(x)$ более удобна для применения, чем функция Римана–фон Мангольдта $\pi(x)$, и что $p(x)$ и $p^{-1}(x)$ можно применять сразу, не дожидаясь окончательного представления функции $\pi(x)$ через нули ρ_n уравнения (12).

5.5. При изучении неасимптотического поведения простых чисел вместо функций $p(x)$ и $p^{-1}(x)$ можно использовать родственные им функции с явно выраженной иерархией различного вида *приближенных почти периодов*

$$\begin{aligned} a(x) &= p(x) - \tilde{p}(x) - (p(x_0) - \tilde{p}(x_0)), \quad x \geq x_0, \\ b(x) &= p^{-1}(x) - R(x) - (p^{-1}(x_0) - R(x_0)), \quad x \geq x_0, \end{aligned}$$

где $\tilde{p}(x)$ есть асимптота (6), а $R(x)$ есть упрощенная формула Римана для $\pi(x)$ (8).

5.6. Фортрановые функции $p(x)$ и $p^{-1}(x)$ применяются так же просто, как и стандартные функции $\sin(x)$ и $\exp(x)$.

В Приложении 2 дана программа *pp_test*, при помощи которой вычислены все таблицы для рисунков в работе. Она, вместе с программой *Dirichlet*, служит иллюстрацией применения функций $p(x)$ и $p^{-1}(x)$. Там же приводится и MAPLE Worksheet, с помощью которого выполнены рисунки в работе.

Приложение 1

Программный пакет $S_{quad}PP_{-}$

```
function p(x)
C 1 <= x < infinity
C Real function "prime number p(x) from real x"
implicit real*8(a-h,o-z)
common/wn/wn
w=1.d0
if(x <= 1.5d0) then
  xx=1.d0; p = x+1.d0
  goto 10
endif
if(x > 1.5d0 .and. x <= wn) then
  ix=dint(x+0.5d0); xx=dfloat(ix); p= sq(ix,x,xx)
  goto 10
endif
if(x > wn) then
  p= r(x)
  goto 10
endif
10 continue
return
end

C
C
function dp(x)
C 1 <= x < infinity
C Derivative of the function "prime number p(x) from real x"
implicit real*8(a-h,o-z)
common/wn/wn
w=1.d0
if(x <= 1.5d0) then
  xx=1.d0; dp = 1.d0
  goto 10
endif
if(x > 1.5d0 .and. x <= wn) then
  ix=dint(x+0.5d0); xx=dfloat(ix); dp= dsq(ix,x,xx)
  goto 10
endif
if(x > wn) then
  dp=dr(x)
  goto 10
endif
10 continue
return
end
```

```

function sq(ix,x,xx)
C      Pair of "left-right" quadrics
implicit real*8(a-h,o-z)
common /protiarithmi/q(10000000)
if(x <= xx) then
    sq=-2.d0*(q(ix)-q(ix-1)-1.d0)*(x-xx)**2+x-xx+q(ix)
else
    sq=2.d0*(q(ix+1)-q(ix)-1.d0)*(x-xx-0.5d0)**2+
&      (2.d0*(q(ix+1)-q(ix))-1.d0)*(x-xx-0.5d0)+
&      (q(ix)+q(ix+1))/2.d0
endif
return
end

C

function dsq(ix,x,xx) !Derivative of "left-right" quadrics
implicit real*8(a-h,o-z)
common /protiarithmi/q(10000000)
if(x <= xx) then
    dsq=-4.d0*(q(ix)-q(ix-1)-1.d0)*(x-xx)+1.d0
else
    dsq=4.d0*(q(ix+1)-q(ix)-1.d0)*(x-xx-0.5d0)+
&      2.d0*(q(ix+1)-q(ix))-1.d0
endif
return
end

C

function r(t) !wn <= t < infinity, Approximation of function p(t),
C R. L. Graham, D. E. Knuth, Oren Patashnik, "Concrete Mathematics",
C A.-W., Menlo Park, CA, 1998, exercise 9.21
implicit real*8(A-H,O-Z)
common/rcorr/rc,drc
r=t*(dlog(t)+dlog(dlog(t)))+(dlog(dlog(t))-2.d0)/
& dlog(t)-((dlog(dlog(t)))**2-6.d0*dlog(dlog(t))+11.d0)/
& (2.d0*dlog(t)**2)-1.d0)+rc
return
end
function dr(x) !wn <= t < infinity, Derivative of function r(t)
implicit real*8(a-h,o-z)
common/rcorr/rc,drc
t1 = log(x); t2 = log(t1); t3 = t1**2; t4 = t3*t1; t7 = t1-2.D0
t8 = log(t7); t18 = t2**2; t21 = t3**2
t35 = -56.D0-4.D0*t2*t4+4.D0*t8*t1+50.D0*t1-13.D0*t3+2.D0*t4-6.D0*
&t8*t3-26.D0*t2*t1+4.D0*t18*t1+2.D0*t2*t21+2.D0*t8*t4-1.D0*t18*t3+
&6.D0*t2*t3+28.D0*t2-4.D0*t18-4.D0*t21+2.D0*t21*t1
dr = 0.5D0*t35/t4/t7+drc
return
end

```

```

subroutine eratosthenes(k)
implicit real*8(a-h,o-z)
common /protiarithmi/q(10000000)
common/rcorr/rc,drc/wn/wn
np=0
C
do n=2,k
  id=1
  isqrtn=dsqrt(dfloat(n))
1  continue
  id=id+1
  if(n == 2) goto 2
  if(mod(n,id) == 0) goto 3
  if(id >= isqrtn) goto 2
  goto 1
2  np=np+1
  q(np)=n
3  continue
enddo
rc=0.d0
wn=np; !rc=dsign(p(wn)-r(wn), q(n)-r(wn))
rc=p(wn)-r(wn)
drc=dp(wn)-dr(wn);
end
C
C
subroutine primes(n)
implicit real*8(a-h,o-z)
common /protiarithmi/q(10000000)
common/rcorr/rc,drc/wn/wn
C
wn=dble(n)
open(20, file='primes')
do i=1, n/6+1
  i1=6*i-5;i2=6*i-4;i3=6*i-3;i4=6*i-2;i5=6*i-1;i6=6*i;
  read(20, *) q(i1),q(i2),q(i3),q(i4),q(i5),q(i6)
enddo
rc=0.d0
rc=p(wn)-r(wn)!rc=dsign(p(wn)-r(wn), q(n)-r(wn))
drc=dp(wn)-dr(wn);
end

```

```

function p_(x)
C   2 <= x < infinity
C   Invers function of the real function "prime number p(x) from real x"
   implicit real*8(a-h,o-z)
   if(x > 1.d0) then
     if(x < 1.d4) then
       y=x/dlog(x)
     else
       y=dlogaritmicintegral(x)
     endif
   else
     y=x
   endif
   dt0=dp(y); r0=dabs(p(y)-x)
   if(x < 359.d4) then
     eps0=350.d0 !initial value
   else
     eps0=500.d0 !new initial value
   endif
   !of autoregularizator
   en=(eps0**2+eps0*dabs(dt0))/r0 !constant for autoregularizator
   ytol=1.d-10; dytol=1.d-10; r=0.d0
   ktol=1000; k=0; small=1.d200
1  continue
   k=k+1
   yy=y
   rr=r
   t=p(yy)-x; dt=dp(yy); r=dabs(t) !current status of
   eps=0.5d0*(dsqrt(dt**2+4.d0*en*r)-dabs(dt)) !autoregularizator
   y=yy-t/(dt+eps) !autoregularized Newtonian iterator
   dif=dabs(y-yy)
   dr=dabs(r-rr)
   if(r >= ytol .and. k <= ktol .and. dr > dytol) then
     if(dif <= small) then
       small=dif
       ybest=y
       rr=r
     endif
     goto 1
   else
     p_=ybest
   endif
   return
end

```



```

function dlogaritmicintegral(x)
implicit real*8(a-h,o-z)
C
dli=0.d0
i=0
itol=100
small=1.d-10
2 continue
    i=i+1
    dliw=dli
    dm=(dlog(x))**i
    if(dm > small) then
        dli=dli+fact(i-1)/dm
        if(dabs(dliw-dli) > small.and.i <= itol) goto 2
    else
        goto 1
    endif
1 continue
dlogaritmicintegral=x*dli
return
end

C
C
C
function fact(l)
implicit real*8(a-h,o-z)
C
if(l.ne.0)then
    ifact=1
    do i=1,l
        ifact=ifact*i
    enddo
    fact=dfloat(ifact)
return
else
    fact=1.d0
return
endif
end

```

Приложение 2

```
program PP_TEST
implicit real*8(a-h,o-z)

open(21, file='px.txt')
open(24, file='in.txt')
open(23, file='de.txt')
open(27, file='ap.txt')
      n=1000000
call eratosthenes(n) !call primes(n)
!   s=10000.d0; sm=0.05d0; ns=1000 !generates curve # 6
!   s=1000.d0; sm=0.05d0; ns=1000 !generates curve # 7
!   s=100.d0; sm=0.05d0; ns=1000 !generates curve # 8
!   s=5980.d0; sm=0.05d0; ns=1000 !uses function primes(n) at n=6000;
!                                   !returns tab for fig 4
!   s=428.d0; sm=0.01d0; ns=600    !returns tabs for figs 1, 2 and 3
s=2.d0; sm=0.05d0; ns=500;        !returns tab for fig 5
      ss=s
do i=1, ns
      s=s+sm
      write(21,*) s, p(s)
      write(23,*) s, dp(s)
      write(27,*) s, p(s)-r(s)-(p(ss)-r(ss))
enddo
      sm=(p(s)-p(ss))/dfloat(ns)
      s=p(ss); sss=s
do i=1, ns
      s=s+sm
      write(24,*) s, p_(s)
enddo
end
```

```

Program Dirichlet_2dimensional_quadric_generator_of_primes
implicit real*8(a-h,o-z)
                n=1000000
call erathostenes(n)
open(23, file='xyz.txt')
pi=dacos(-1.d0)
a=11.d0; b=7.d0; c=13.d0                !Dirichlet's parameters:
                m=100                    !(a,2b)=1, (2b,c)=1, (a,c)=1, b^2 /= ab
do i=1, m
  x=dfloat(i)
  do j=1, m
    y=dfloat(j)
    z=a*x**2+2.d0*b*x*y+c*y**2        !Dirichlet prime surface
    f=dsin(pi*p_(z))                !the real function z primality test
    if(dabs(f) < 1.d-12) then
      write(23, *) x, y, z
    endif
  enddo
enddo
enddo
end

```

MAPLE Worksheet

```

> restart:                                digits:=8;
> p_:=readdata("f:\October02\in.txt", [float,float]);
> p:=readdata("f:\October02\px.txt", [float,float]);
> dp:=readdata("f:\October02\de.txt", [float,float]);
> ap:=readdata("f:\October02\ap.txt", [float,float]);

> plot(p,color=black,style=point,symbol=circle,symbolsize=4,
      scaling=UNCONSTRAINED,axes=boxed);
> plot(ap,color=navy,style=point,symbol=circle,symbolsize=4,
      scaling=UNCONSTRAINED,axes=boxed);
> plot(dp,color=maroon,style=point,symbol=circle,symbolsize=4,
      scaling=UNCONSTRAINED,axes=boxed);
> plot(p_,color=red,style=point,symbol=circle,symbolsize=4,
      scaling=UNCONSTRAINED,axes=boxed);

```

Список литературы

- [1] Lubomir Alexandrov, *Prime Number Sequences and Matrices Generated by Counting Arithmetic Functions*, JINR Communication, E5-2002-55, Dubna, 2002; *Prime Number Logarithmic Geometry on the Plane*, e-print: math.GM/0204167.
- [2] D. H. Bailey, *Multiprecision Translation and Execution of FORTRAN Programs*, ACM Transactions on Mathematical Software, Vol. 19, No. 3, September (1993), 288-319.
- [3] Р. Грэхем, Д. Кнут, О. Паташник, *Конкретная математика*, Москва, "Мир", 1998
- [4] Л. Александров, *Регуляризованные вычислительные процессы Ньютона-Канторовича*, ЖВМиМФ, Т. 11, №1, (1971), 36-43.
- [5] Hans Riesel, *Prime Numbers and Computer Methods for Factorisation*, Birkhauser, 1985.
- [6] M. V. Berry, *Quantum chaology* Proc. Roy. Soc. Lond., A413, , 183-198 (1987).
- [7] Alain Connes, *Trace formula in noncommutative Geometry and the zeros of the Riemann zeta function*, e-print: math.NT/9811068 .
- [8] L. Alexandrov, *Регуляризованные траектории приближения ньютоновского типа для решения нелинейных уравнений*, Ж. дифф. уравнения, XIII, №7, 1281-1292 (1977).

Получено 3 октября 2002 г.

Александров Л., Баранов Д. Б., Йотов П.
Полиномиальные сплайны,
точно воспроизводящие простые числа

P5-2002-228

Построена дифференцируемая и обратимая действительная функция $p(x)$, воспроизводящая ряд простых чисел до заданного N . Для функций $p(x)$, $\frac{dp(x)}{dx}$ на $1 \leq x < \infty$, и $p^{-1}(x)$ на $2 \leq x < \infty$ созданы стандартные программы. Обратная функция $p^{-1}(x)$ является дифференцируемым аналогом теоретико-числовой функции $\pi(x)$. Функция $p^{-1}(x)$ сравнивается с выражением Римана для $\pi(x)$ через нули ζ -функции.

Работа выполнена в Лаборатории теоретической физики им. Н. Н. Боголюбова ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна, 2002

Перевод авторов

Alexandrov L., Baranov D. B., Yotov P.
Polynomial Splines Reproducing Primes

P5-2002-228

A differentiable and reversible real function $p(x)$ reproducing primes up to a given N has been built. Standard programs for the functions $p(x)$, $\frac{dp(x)}{dx}$ on $1 \leq x < \infty$, and $p^{-1}(x)$ on $2 \leq x < \infty$ were created. The inverse function $p^{-1}(x)$ is a differentiable analogue of the number-theoretic function $\pi(x)$. The function $p^{-1}(x)$ is compared with the Riemann's exact expression for $\pi(x)$.

The investigation has been performed at the Bogoliubov Laboratory of Theoretical Physics, JINR.

Preprint of the Joint Institute for Nuclear Research. Dubna, 2002

Редактор *М. И. Зарубина*
Макет *Н. А. Киселевой*

Подписано в печать 09.10.2002.

Формат 60 × 90/16. Бумага офсетная. Печать офсетная.

Усл. печ. л. 1,18. Уч.-изд. л. 1,2. Тираж 325 экз. Заказ № 53558.

Издательский отдел Объединенного института ядерных исследований
141980, г. Дубна, Московская обл., ул. Жолио-Кюри, 6.