Computing the Prime Counting Function

Kevin Stueve under the direction of William Stein
Summer 2009
Last edited 2009-11-02
University of Washington, Seattle

kstueve@uw.edu
wstein@gmail.com

Abstract
The best known algorithm for computing the prime counting function (prime_pi) and nth prime function of numbers within a practical range, hybrid table look-up and sieving, is described. The author implemented this algorithm for inclusion in Sage, the free open-source mathematics program. The author's implementation is competitive with Mathematica, the leading commercial mathematics program, in both the range of numbers allowed and the computation time. The author's code is released under GPL version 2, or any later version (at the user's choice), and is awaiting further work by the Sage development team to make it build and function on any platform. The code is published at http://trac.sagemath.org/sage_trac/ticket/7013.

Introduction, Background, History, and Importance
A natural number is prime if it has exactly two distinct natural number factors, otherwise it is composite. The number one is considered neither prime nor composite. Primes have been studied since antiquity and are some of the most beautiful and mysterious mathematical objects. Albert Einstein said, "the most beautiful thing we can experience is the mysterious. It is the source of all true art and science." Although primes can be defined in terms of arithmetic (factors and divisibility), they have been understood from new perspectives throughout history- they have been found to have strong relationships to calculus, analysis and computer science [Chaitin]. Shor's algorithm for integer factorization, one of the few known quantum computer algorithms was shown to have polynomial time complexity in 1995, creating a link between primes and physics. New ways of understanding the primes may continue to be discovered far into the future.

The prime counting function, denoted by the lower case Greek letter π, is the number of primes less than or equal to a given number. The inverse of the prime counting function is the nth prime function, which returns the nth prime for a given n. The prime counting function is important because the Riemann Hypothesis, which has been called the most important problem in mathematics, is a statement of the growth of the prime counting function. Computing the prime counting function is one of the most fundamental problems in number theory.

> Of all the questions that can be asked [of the primes], perhaps the three most fundamental are the following
> (1) Is a given number prime?
> (2) How many primes are there less than or equal to a given number x?
> (3) What is the xth prime p_x?
> Julian Havil, Gamma

The first question was shown to be solvable in time that grows as a polynomial of the number of bits in the number in question by the AKS primality test in 2002.

The second two questions are closely related, in that if an algorithm is provided to answer one of them, an algorithm to answer the other can be constructed. All of the known algorithms to answer either of these questions require time that grows exponentially with the number of bits in the argument. The largest computed value of the prime counting function is $\pi(10^{23})$ by Tomas Oliveira e Silva in 2006 using two months of computation time.

Oliveira e Silva's algorithm is the latest improvement to the LMO family of combinatorial algorithms for calculating the prime counting function and requires $O(n^{2/3}/\log^2 n)$ time. Other algorithms for computing the prime counting function are sieving (using either the sieve of Eratosthenes or Atkin), which require O(n) and O(n/ log log n) time respectively, the analytic method which requires $O(n^{1/2})$ time, but is incredibly hard to implement and has a large constant implied by the O notation, and the hybrid table look-up sieving algorithm. Here n is the argument to the prime counting function. Because only a finite table can be stored, it is difficult to provide an asymptotic analysis of the complexity of the hybrid table look-up sieving algorithm. However, for numbers in a practical range, it outperforms the other algorithms- and the performance can always be improved by using a larger table or splitting the sieving work between multiple processors.

Description of the Algorithm

The hybrid table look-up sieving algorithm uses a large table of values of the prime counting function. To reduce space requirements for the table, only the difference between the logarithmic integral and the prime counting function is stored for each entry. To reduce the time complexity, an asymptotic series for the logarithmic integral is used.

To calculate prime_pi of a given number x, the entry in the table that is closest to x is retrieved, and the number of primes in the interval between the table entry and x is counted by a siever. If the interval is small enough, it may be faster to count the primes using a primality test instead of a siever (as a siever must generate all the primes up to the square root of x, creating overhead compared with the use of a primality test).

To calculate the nth prime, the entry in the table with with largest prime_pi(A) not greater than n is found using binary search of the table. An approximation to the prime counting function, prime_pi(A)+Li(x)-Li(A) is inverted using the Newton–Raphson method to obtain a good approximation to the nth prime (note that calculating prime_pi(A) requires only a table query). The exact prime counting function of this initial guess is calculated (one invocation of the prime counting function) and a primality test or sieving from there is used to obtain the exact nth prime.

Another possible approximation to the prime counting function besides prime_pi(A)+Li(x)-Li(A) is prime_pi(B)+Li(x)-Li(B), where B corresponds to the smallest prime_pi(B) not less than n in the table. An even better approximation is a weighted average of the two previously mentioned approximations.

Conclusion

Many questions are introduced by the hybrid table look-up and sieving algorithm described in this paper. Among the possible questions the author of this paper leaves unanswered are:

- How much can a table of values of the prime counting function be compressed without prohibitively slowing down retrieval of a table entry?
- What is the nature of the error of the approximation prime_pi(x)~=prime_pi(A)+Li(x)-Li(A) for an A close to x?
- How much would the error in the previous approximation change if Li were replaced with a more accurate approximation to the prime counting function?
- Is it possible to have a dense enough table and an accurate enough approximation to the prime counting function that an exact value of the prime counting function can be easily found?

- What is the ultimate time complexity of the prime counting function?
- Could the hybrid table look-up sieving algorithm be adapted to calculate floating point approximations to harmonic prime sums?
- Could a quantum algorithm calculate the prime counting function?
- What is the fastest algorithm to count the number of primes in an interval?

Acknowledgments

The author would like to acknowledge:
- The VIGRE grant from the National Science Foundation (NSF) which supported this work, and the UW mathematics department for awarding the grant
- William Stein, the author's advisor during this project, for all of his help and support, including the use of a laptop for the duration of this project
- Victor Miller, who provided lots of helpful advice
- The other Sage Days 17 participants for their thoughts and input, especially Tom Boothby and Robert Bradshaw for their thoughts regarding the variance of Li(x) - prime_pi(x) and Robert Miller for asking hard questions
- National Science Foundation Grant No. DMS-0821725, for the sage.math.washington.edu computer
- Andrew Booker, author of the Nth Prime Page at http://primes.utm.edu/nthprime/ which describes the hybrid table look-up and sieving method (which he devised)
-Tomas Oliveira e Silva, author of the fast implementation of the segmented sieve of Eratosthenes used to count the number of primes in an interval
- Andrey V. Kulsha, Anotoly F. Selvich, Tomas Oliveira e Silva, and any other contributors to http://www.primefan.ru/stuff/primes/table.html, which has extensive tables of the prime counting function
- Fredrik Johansson, for providing a fast asymptotic series approximation to the logarithmic integral function
- Andrew Ohana for offering his time to describe his optimized Legendre prime_pi algorithm
- Jonathan Voyce for pointing out a typographical error in this report
- Everyone else who helped the author accomplish this project who is not mentioned here

Bibliography

The Combinatorial Method:
Computing pi(x): The Meissel-Lehmer Method
Jeffrey Lagarias, Victor Miller, Andrew Odlyzko 1985
http://www.dtc.umn.edu/~odlyzko/doc/arch/meissel.lehmer.pdf

Computing pi(x): The Meissel, Lehmer, Lagarias, Miller, Odlyzko Method
Deleglise, Rivat 1996
http://cr.yp.to/bib/1996/deleglise.pdf

Computation of pi(x) : Improvements to the Meissel, Lehmer, Lagarias, Miller, Odlyzko, Deleglise and Rivat method
Xavier Gourdon 2001
http://numbers.computation.free.fr/Constants/Primes/Pix/piNalgorithm.ps

Computing pi(x): The combinatorial method
Tomas Oliveira e Silva 2006
http://www.ieeta.pt/~tos/bib/5.4.pdf

Computing Prime Harmonic Sums
Eric Bach, Dominic Klyve, and Jonathan P. Sorenson
Mathematics of Computation
Volume 78, Number 268, October 2009, Pages 2283–2305

The Analytic Method:
Computing pi(x): An Analytic Method
Lagarias, Odlyzko 1987
http://www.dtc.umn.edu/~odlyzko/doc/arch/analytic.pi.of.x.pdf

Implementing the Lagarias-Odlyzko Analytic Algorithm for pi(x) (draft)
William F. Galway 1998
http://www.cecm.sfu.ca/~wfgalway/SlidesETC/thesis-slides-98.ps.gz

Thesis by William Galway (UIUC): Analytic Computation of pi(x) 2004
http://www.math.uiuc.edu/~galway/PhD_Thesis/thesis-twoside.pdf

The Hybrid Table Look-up and Sieving Method:
The Nth Prime Page, A prime page by Andrew Booker
Copyright 1999-2009 Chris Caldwell
http://primes.utm.edu/nthprime/index.php

The Fluctuations of the Prime-Counting Function
Andrey V. Kulsha, Anotoly F. Selvich, Tomas Oliveira e Silva 2009-04-07
http://www.primefan.ru/stuff/primes/table.html

Sieving:
Fast Implementation of the Segmented Sieve of Eratosthenes
Tomas Oliveira e Silva, 2003-05-03
http://www.ieeta.pt/~tos/software/prime_sieve.html

Prime Sieves Using Binary Quadratic Forms
A. O. L. Atkin and D. J. Bernstein 2003-12-19
http://cr.yp.to/papers/primesieves.ps

Prime Number Theory:
What is Riemann's Hypothesis? (Draft)
Barry Mazur and William Stein 2009
http://wstein.org/rh/

Mathematical Mysteries, the Beauty and Magic of Numbers
Calvin C. Clawson 1996
Perseus Books

Gamma, Exploring Euler's Constant
Julian Havil 2003
Princeton

Prime Obsession, Bernhard Riemann and the Greatest Unsolved Problem in Mathematics
John Derbyshire 2003
Plume

Riemann's Zeta Function
H.M. Edwards 1974
Dover

An Introduction to the Theory of Numbers
G.H. Hardy, E.M. Wright 1938

Oxford Science Publications

Meta Math! The Quest for Omega
Gregory Chaitin 2005
Vintage

PRIMES is in P
Manindra Agrawal, Neeraj Kayal, Nitin Saxena
Annals of Mathematics 160 (2004), no. 2, pp. 781–793
http://www.cse.iitk.ac.in/users/manindra/algebra/primality_v6.pdf

Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer
Peter W. Shor (AT&T Research) 1996-01-25
http://arxiv.org/abs/quant-ph/9508027

The Prime Counting Function and Related Subjects
Patrick Demichel (2005)
http://web.archive.org/web/20060908033007/http://demichel.net/patrick/li_crossover_pi.pdf