Sieving for pseudosquares and pseudocubes in parallel using doubly-focused enumeration and wheel datastructures

Jon Sorenson
sorenson@butler.edu
http://www.butler.edu/~sorenson

Computer Science & Software Engineering
Butler University
Indianapolis, Indiana USA

ANTS IX @ Nancy, France, July 2010

Outline

Pseudosquares

Pseudosquares

Let (x/y) denote the Legendre symbol.
For an odd prime p, let $L_{p,2}$, the pseudosquare for p, be the smallest positive integer such that

1. $L_{p,2} \equiv 1 \pmod{8}$,
2. $(L_{p,2}/q) = 1$ for every odd prime $q \leq p$, and
3. $L_{p,2}$ is not a perfect square.

Finding pseudosquares is motivated by the pseudosquares primality test.

Pseudosquares Prime Test (Lukes, Patterson, Williams 1996)

Let n, s be positive integers. If

- All prime divisors of n exceed s,
- $n/s < L_{p,2}$ for some prime p,
- $p_i^{(n-1)/2} \equiv \pm 1 \pmod{n}$ for all primes $p_i \leq p$, and
- $2^{(n-1)/2} \equiv -1 \pmod{n}$ when $n \equiv 5 \pmod{8}$, or $p_i^{(n-1)/2} \equiv -1 \pmod{n}$ for some prime $p_i \leq p$ when $n \equiv 1 \pmod{8}$,

then n is prime or a prime power.

This combines nicely with trial division up to s or, even better, sieving by primes up to s over an interval.

Pseudocubes

Pseudocubes

For an odd prime p, let $L_{p,3}$, the pseudocube for p, be the smallest positive integer such that

1. $L_{p,3} \equiv \pm 1 \pmod 9$,
2. $L_{p,3}^{(q-1)/3} \equiv 1 \pmod q$ for every prime $q \le p$, $q \equiv 1 \pmod 3$,
3. $\gcd(L_{p,3}, q) = 1$ for every prime $q \le p$, and
4. $L_{p,3}$ is not a perfect cube.

- There is a pseudocube primality test (Berrizbeitia, Müller, Williams 2004).
- See also the next talk.

Computational Results: New Pseudosquares

New Pseudosquares

p	$L_{p,2}$
367	36553 34429 47705 74600 46489
373	42350 25223 08059 75035 19329
379	$> 10^{25}$

- Previous bound was $L_{367,2} > 120120 \times 2^{64} \approx 2.216 \times 10^{24}$ by Wooding & Williams, 2006.
- $L_{367,2}$ and $L_{373,2}$ were found in 2008 using 3 months (wall time) on Butler's *Big Dawg* cluster supercomputer.
- Extending the computation to 10^{25} took another 6 months time, finishing on January 1st 2010.

Computational Results: New Pseudocubes

New Pseudocubes

p	$L_{p,3}$
499	601 25695 21674 16551 89317
523,541	1166 14853 91487 02789 15947
547	41391 50561 50994 78852 27899
571,577	1 62485 73199 87995 69143 39717
601,607	2 41913 74719 36148 42758 90677
613	67 44415 80981 24912 90374 06633
619	$> 10^{27}$

- This took 6 months of wall time in 2009.
- $L_{499,3} > 1.45152 \times 10^{22}$ was previously found by Wooding & Williams, 2006.

Conjectured Growth Rates

- Let p_i denote the ith prime, and
- Let q_i denote the ith prime such that $q_i \equiv 1 \pmod 3$.

Using reasonable heuristics, it is conjectured that there exist constants $c_2, c_3 > 0$ such that

$$\begin{aligned} L_{p_n,2} &\approx c_2 2^n \log p_n, \\ L_{q_n,3} &\approx c_3 3^n (\log q_n)^2. \end{aligned}$$

(Lukes, Patterson, Williams 1996)
(Berrizbeitia, Müller, Williams 2004)

Conjectured Growth Rates

Let us define

$$c_2(n) := \frac{L_{p_n,2}}{2^n \log p_n},$$
$$c_3(n) := \frac{L_{q_n,3}}{3^n (\log q_n)^2}.$$

We find that

- $5 < c_2(n) < 162$ for $n \leq 74$ (averaging around 45), and
- $0.05 < c_3(n) < 6.5$ for $10 \leq n \leq 53$ (averaging around 1.22).

Note that

$$L_{p_n,2} = L_{p_{n+1},2} = \cdots = L_{p_{n+k},2}$$

for $k \geq 1$ can occur. (See proceedings page 334.)

Algorithm Outline

- Doubly-Focused Enumeration
- Parallelized by target interval
- Space-saving Wheel Datastructure

We'll focus on pseudosquares for the remainder of the talk.

Doubly-Focused Enumeration

Doubly-Focused Enumeration (Bernstein 2004)

Every integer x, with $0 \leq x \leq H$, can be written in the form

$$x = t_p M_n - t_n M_p$$

where

- $\gcd(M_p, M_n) = 1$,
- $0 \leq t_p \leq \dfrac{H + M_n M_p}{M_n}$,
- and $0 \leq t_n < M_n$.

Doubly-Focused Enumeration

We used

$$\begin{aligned} M_p &= 7 \cdot 11 \cdot 13 \cdot 17 \cdot 19 \cdot 23 \cdot 29 \cdot 31 \cdot 37 \cdot 41 \cdot 43 \cdot 53 \cdot 89 \\ &= 2057\,04617\,33829\,17717 \qquad \text{and} \\ M_n &= 8 \cdot 3 \cdot 5 \cdot 47 \cdot 59 \cdot 61 \cdot 67 \cdot 71 \cdot 73 \cdot 79 \cdot 83 \cdot 97 \\ &= 4483\,25952\,77215\,26840. \end{aligned}$$

Parallelization

We parallelized over t_p intervals:

- Each processor was assigned an interval $[a, b]$,
- Find all t_p values, $a \leq t_p \leq b$ and sort them.
- Compute a range of t_n values to correspond.
- Generate the t_n values (out of order).
- Compute an x value (implicitly at first) using binary search on the t_p list, and sieve/test it.

Wheel Datastructure

Wheel Datastructure Example

We will generate squares modulo $24 \cdot 5 \cdot 7 = 840$.
Note that all must be 1 mod 24.

Table for 5 (modulus $24 \equiv 4 \bmod 5$)

	0	1	2	3	4
square	0	1	0	0	1
jump	24	48	24	48	72

Table for 7 (modulus $120 = 24 \cdot 5 \equiv 1 \bmod 7$)

	0	1	2	3	4	5	6
square	0	1	1	0	1	0	0
jump	120	120	240	120	480	360	240

Example continued

Generating Squares

24	5	7			
1	1	1	121	361	(841)
	49	169	289	529	(1009)
	(121)				

We get the list

$$1, 121, 361, 169, 289, 529$$

of squares modulo $24 \cdot 5 \cdot 7 = 840$.

Future Work

Future Work

GPUs!!

Thank You

- For your attention
- To the organizers
- To the Holcomb Awards Committee for $$
- To Frank Levinson for the supercomputer