Sieve theory and gaps between primes: Narrow admissible tuples

Andrew V. Sutherland
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
(on behalf of D.H.J. Polymath)

Explicit Methods in Number Theory
MATHEMATISCHES FORSCHUNGSINSTITUT OBERWOLFACH

July 10, 2015

Explicitly proving bounded gaps

Recall that our goal is to prove upper bounds on

$$H_m := \liminf_{n\to\infty} p_{n+m} - p_n.$$

We do this by finding an integer k and a function $F \in L^2(\mathcal{R}_k)$ for which[†]

$$\rho(F) > 4m.$$

This shows $M_k = \sup \rho(F) > 4m$, which in turn implies DHL$[k, m+1]$.

We can then conclude that $H_m \le H(k)$, where

$$H(k) := \min\{\operatorname{diam} \mathcal{H} : \mathcal{H} \text{ is an admissible } k\text{-tuple}\}.$$

We are thus interested in explicit bounds for $H(k)$.
We obtain upper bounds by constructing narrow admissible k-tuples.

[†]We can alternatively use $F \in L^2(\alpha\mathcal{R}_k)$ with $\alpha := \min(\frac{\delta}{1/4+\varpi}, 1)$, for any $600\varpi + 180\delta < 7$.
In this case we only require $\rho(F) > \frac{1}{1/4+\varpi}$.

An easy asymptotic upper bound

Recall that a k-tuple $\mathcal{H} = \{h_1, \ldots, h_k\}$ is *admissible* if the reduction map $\mathcal{H} \to \mathbb{Z}/p\mathbb{Z}$ is not surjective for any prime p. This clearly holds for $p > k$.

If we put $n = \pi(k)$, then the set

$$\{p_{n+1}, p_{n+2}, \ldots, p_{n+k}\}$$

is an admissible k-tuple. If we apply the asymptotic bounds

$$p_n = n \log n + n \log \log n - n + O\left(n \frac{log \log n}{\log n}\right),$$

$$\pi(k) = \frac{k}{\log k} + O\left(\frac{k}{(\log k)^2}\right)$$

with $n = \pi(k)$ we obtain

$$H(k) \leq p_{n+k} - p_{n+1} = k \log k + k \log \log k - k + o(k).$$

An asymptotic lower bound

Let $M(y)$ be the largest integer k for which $\operatorname{diam}\mathcal{H} < y$ for some admissible k-tuple $\mathcal{H}$. Then $M(H(k)) = k-1$, and for any integers k and d,

$$M(d) \leq k-1 \iff H(k) \geq d.$$

An explicit form of the Brun-Titchmarsh theorem due to Montogmery and Vaughan states that

$$\pi(x+y) - \pi(x) \leq \frac{2y}{\log y}$$

for all integers $x \geq 1$ and $y \geq 2$. The proof involves sieving the interval $[x+1, x+y]$ and can be adapted to show that $M(y) \leq \frac{2y}{\log y}$ for all $y \geq 2$. From this one may deduce $M(\frac{1}{2}k\log k) \leq k-1$ for $k \geq 8$, and therefore

$$H(k) \geq \frac{1}{2}k\log k,$$

which in fact holds for all integers $k \geq 2$.

Incompatibility of the Hardy-Littlewood conjectures

The prime tuples conjecture if the first of two conjectures by Hardy Littlewood made in the same paper. The second is the following.

Conjecture (Hardy-Littlewood 1923)

For all $x, y \geq 2$ *we have* $\pi(x+y) - \pi(x) \leq \pi(y)$.

Taken together, the two Hardy-Littlewood conjectures together imply

$$H(k) \geq p_k.$$

Theorem (Hensley-Richards 1972)

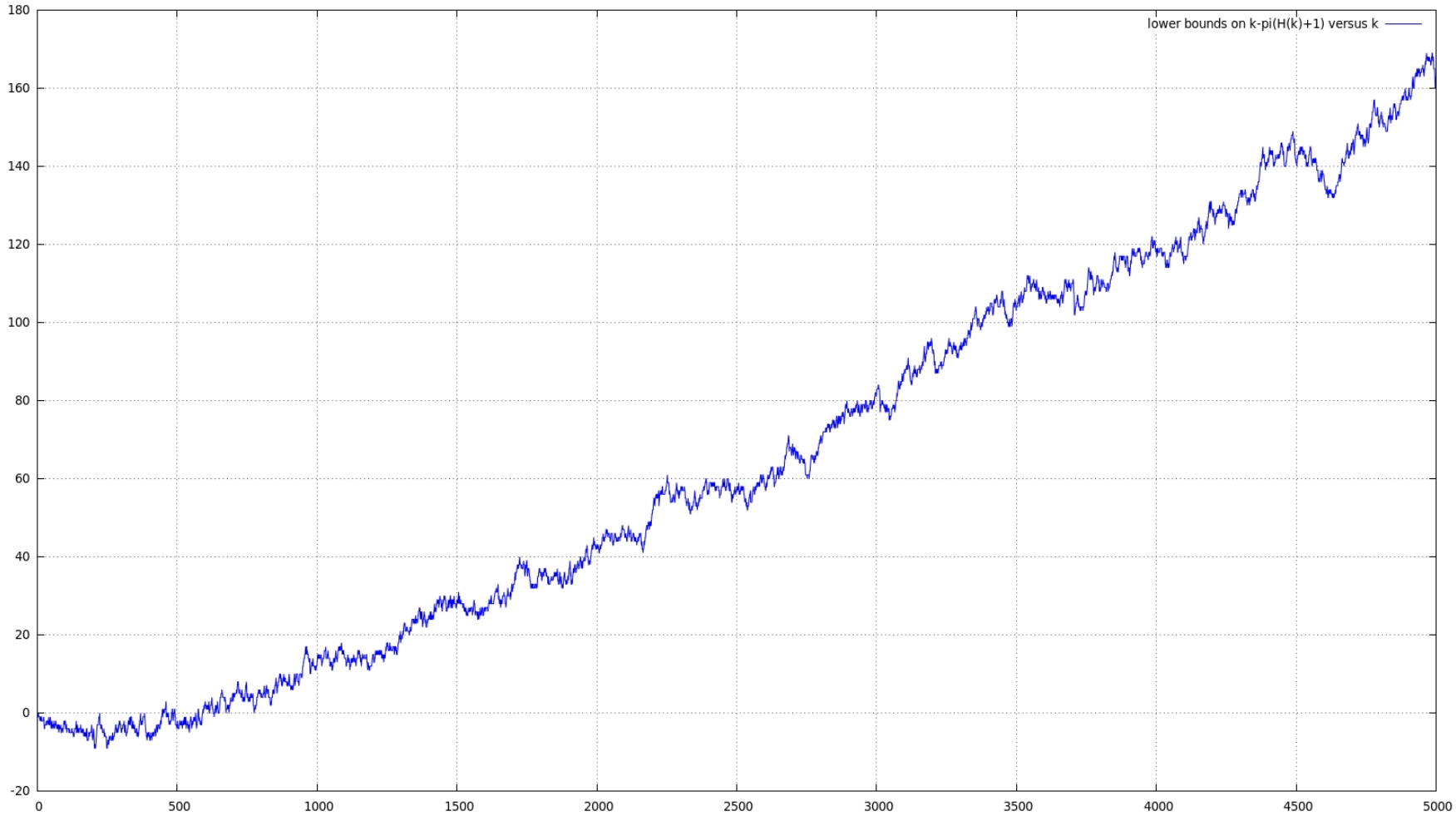
$H(k) < p_k$ *for all sufficiently large* k.

Theorem (Engelsma 2005)

$H(447) \leq 3158 < p_{447} = 3163\dagger$.

†Conjecturally, the first Dickson 447-tuple should lie between 10^{174} and 10^{1199}.

Incompatibility of the Hardy-Littlewood conjectures



Better asymptotic upper bounds on $H(k)$

Hensley and Richards obtained their results by sieving a centered interval $[-x/2, x/2]$ rather than $[0, x]$. They proved the asymptotic bound

$$H(k) \leq k \log k + k \log \log k - (1 + \log 2)k + o(k).$$

Schinzel showed that by sieving $[0, x]$ at $1 \bmod 2$ and $0 \bmod p$ for odd primes p should conjecturally improve this to

$$H(k) \leq k \log k + k \log \log k - (1 + 2 \log 2)k + o(k),$$

and Hensley and Richard pushed this further and conjectured the bound

$$H(k) \leq k \log k + k \log \log k - (1 + o(1))k \log \log \log k.$$

But we believe that in fact $H(k) \leq k \log k + (1 + o(1))k$.

Conjecture

$H(k) < k \log k + k$ *for all* $k \geq 389$.

Constructing admissible tuples by sieving

We can construct an admissible k-tuple by sieving the integers of one residue class modulo each prime $p \leq k$ and taking the k least survivors. (sieving $0 \bmod p$ amounts to taking the first k primes greater than k).

But this is overkill, one can typically terminate the sieve early.

Some examples (start by picking a sufficiently large x):

1. Eratosthenes: sieve $[2, x]$ at $0 \bmod p$ until the k least survivors form an admissible k-tuple.
2. Hensley-Richards: sieve $[-\frac{x}{2}, \frac{x}{2}]$ at $0 \bmod p$ until the k survivors of least absolute value form an admissible k-tuple
3. Schinzel: sieve $[2, x]$ at $1 \bmod p$ for $p \leq y$ and $0 \bmod p$ for $p > y$ until the k least survivors form an admissible tuple.
4. Greedy: sieve $[0, x]$ of a minimally occupied residue class $a \bmod p$ until the k least survivors form an admissible tuple.

Shifting the sieve interval slightly often yields better results.

Discipline versus greed

All of the structured approaches are demonstrably sub-optimal.
And the greedy approach is often worse than any of them!

However, there is a hybrid approach works that remarkably well.

Let $w = k \log k + k$, and for even integers s in $[-w, w]$:

1. Sieve $[s, s + w]$ at $1 \bmod 2$ and $0 \bmod p$ for primes $p \leq \sqrt{w}$.
2. For increasing primes $p > \sqrt{w}$ sieve a minimally occupied residue class mod p until the tuple $\mathcal{H}$ of survivors is admissible.
3. If $|\mathcal{H}| \neq k$, adjust the sieving interval and repeat until $|\mathcal{H}| = k$.

Output an $\mathcal{H}$ with minimal diameter among those constructed.

This algorithm is not optimal, but it typically gets within one percent of the best known results (including cases where $H(k)$ is known).

(demo)

Sieve comparison

k	632	1783	34 429	341 640	3 500 000	75 845 707	3 473 955 908
k primes past k	5028	16 174	420 878	5 005 362	59 874 594	1 541 858 666	84 449 123 072
Eratosthenes	4860	15 620	411 946	4 923 060	59 093 364	1 526 698 470	83 833 839 848
H-R	4918	15 756	402 790	4 802 222	57 554 086	1 488 227 220	81 912 638 914
Shifted Schinzel	4868	15 484	399 248	4 740 846	56 789 070	1 467 584 468	80 761 835 464
Shifted hybrid	4710	15 036	388 076	4 603 276	55 233 744	1 431 556 072	not available
Best known	4680	14 950	386 344	4 597 926	55 233 504	1 431 556 072	80 550 202 480
$\lfloor k \log k + k \rfloor$	4707	15 130	394 096	4 694 650	56 238 957	1 452 006 268	79 791 764 059

Getting that last one percent

Given a narrow admissible tuple, there are a variety of combinatorial optimization methods that we can apply to try and improve it.

These include local search and perturbation methods, such as simulated annealing.

The technique that we found most useful uses a genetic algorithm. For $k < 10^6$, all of the best known admissible k-tuples not previously found by Engelsma's search were constructed by some version of this algorithm.

Given a k-tuple $\mathcal{H}$, we generate a new tuple $\mathcal{H}'$ by sieving the same interval of the same residue classes for $p \leq \sqrt{k \log k}$, and randomly choosing a nearly minimally occupied class for $p > \sqrt{k \log k}$.

The set $\mathcal{H} \cup \mathcal{H}'$ contains an admissible k-tuple (namely, $\mathcal{H}$), but if we sieve this set by greedily choosing residue classes as required, we may obtain a k-tuple $\mathcal{H}''$ that is actually narrower than $\mathcal{H}$.

Database of admissible tuples

We have established an online database of admissible tuples.

It includes at least one example of an admissible k-tuple of least known diameter for $2 \leq k \leq 5000$ and is open for submission.

For $k \leq 342$ it contains optimal tuples contributed by Engelsma.
For many $k > 342$ we have tuples narrower than those obtained by Engelsma, and in other cases we independently matched his results.

Finding better lower bounds for $H(k)$ remains an open problem.

For $k = 632$ we were able to prove $H(632) \geq 4276$, but we expect that in fact the upper bound $H(632) \leq 4680$ is tight.

(admissible 632-tuple of diamter 4680)

Explicit lower bounds

There are two methods for proving explicit lower bounds that may be significantly better than $\frac{1}{2}k\log k$ for small to medium size k.

For any partition $d = d_1 + \cdots + d_n$ of a positive integer d we have

$$M(d) \le \sum M(d_i).$$

For $d_i \le 342$ optimal bounds on $M(d_i)$ are known, and we can take the minimum of $\sum M(d_i)$ over all such partitions as an upper bound on $M(d)$, which implies a lower bound on $H(k)$. This works fairly well for $k \le 1000$.

For larger k we use an inclusion/exclusion approach (see next slide).

Both approaches can be combined with an exhaustive sieving step, in which we restrict to tuples that do not occupy a fixed set of residue classes modulo small primes p (say $p \le 19$). Iterating over all possible choices of and taking the worst case yields a general bound.

Examples of lower bounds obtained by these methods can be found here.

Lower bounds via inclusion/exclusion

Let $\mathcal{H}$ be an admissible k-tuple of diameter $H(k)$ in the interval $I = [0, H(k)]$. For $p \leq k$, let a_p denote a residue class modulo p not present in $\mathcal{H}$, and let $S_p := \{n \in I : n \equiv a_p \bmod p\}$. Then $\mathcal{H} = I - \bigcup_p S_p$, and

$$|\mathcal{H}| = |I| - \left|\bigcup_p S_p\right|,$$

and this implies $H(k) + 1 = k + \left|\bigcup_p S_p\right|$, so lower bounds on $\left|\bigcup_p S_p\right|$ imply lower bounds on $H(k)$. By inclusion/exclusion we have

$$\left|\bigcup_p S_p\right| \geq \sum_p |S_p| - \sum_{\substack{p,q \\ q<p}} |S_p \cap S_q| = \sum_p \left(|S_p| - \sum_{\substack{q \\ q<p}} |S_p \cap S_q|\right).$$

We may bound each term on the RHS from below by iterating over residue classes $a_p \bmod p$, and for each $q < p$ choosing a_q to maximize $|S_p \cap S_q|$; the minimum value of $|S_p| - \sum_q |S_p \cap S_q|$ is then a lower bound on the pth term on the RHS, and we sum the nonzero lower bounds thus obtained.

Fast admissibility testing

Simple approach: for each prime $p \leq k$, construct a bit-vector $\mathbf{v}$ with the $(h \bmod p)$th bit of $\mathbf{v}$ set for each $h \in \mathcal{H}$. If $\mathbf{v} = (1, \ldots, 1)$ for any p then reject, otherwise accept. $O(k^2 \log\log k \log\log\log k)$ time, assuming fast arithmetic.

This can be heuristically improved by a factor of $\approx \log k$ as follows:

1. Represent $\mathcal{H}$ as a bit-vector $\mathbf{b}$ with $\mathbf{b}_h = 1 \Leftrightarrow h \in \mathcal{H}$.
2. For each $p \leq k$, choose m so that k randomly dropped balls are unlikely to occupy all of the first m of p bins (so $m = O(1)$ for $p = k/c$).
3. To test whether $\mathcal{H}$ occupies the residue classes $[1, m] \bmod p$, check $\mathbf{b}_1, \ldots, \mathbf{b}_m, \mathbf{b}_{p+1}, \ldots, \mathbf{b}_{p+m}, \ldots$, a total of $m\lceil|\mathcal{H}|/p\rceil$ bits.
4. If $\mathcal{H}$ does not occupy every residue classes in $[1, m] \bmod p$ then $\mathcal{H}$ is admissible at p; otherwise test admissibility at p as above.

In practice this algorithm is typically faster than the simple approach by a factor of 10 or 20 for $k \in [10^5, 10^{10}]$.

Fast sieving for very large k

The Hensley-Richards and Schinzel sieves can both be achieved in quasi-quadratic time (essentially the cost of a single admissibility test).

Let us illustrate how this is done in the case of the Hensley-Richards sieve.

Consider the k-tuple

$$\mathcal{H}(m) := (-p_{m+\lfloor k/2 \rfloor - 1}, \ldots, -p_{m+1}, \ldots, -1, 1, \ldots, p_{m+1}, p_{m+\lceil k/2 \rceil - 1}).$$

For $m = \pi(k)$ we know $\mathcal{H}(m)$ is admissible. Now iteratively decrement m, and for each new value check whether $\mathcal{H}(m)$ is admissibile modulo p_{m+1}. As soon as this fails, increment m and do a full admissibility test.

This will usually succeed, but if not, increment m until it does.

When k is very large, in order to save space one may use a windowed sieve, with a window size approximately equal to the square-root of the length of the interval being sieved.