Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

A. Granville, J. van de Lune, H.J.J. te Riele

Checking the Goldbach conjecture on a vector computer

Department of Numerical Mathematics Report NM-R8812 September

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

Checking the Goldbach Conjecture on a Vector Computer

A. Granville
University of Toronto
Department of Mathematics
Toronto, Canada M5S 1A1

J. van de Lune
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

H.J.J. te Riele
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

The Goldbach conjecture says that every even number can be expressed as the sum of two primes and it is known to be true up to 10^8 (except for 2, if 1 is not considered a prime).
This paper describes the results of a numerical verification of the Goldbach conjecture on a Cyber 205 vector computer up to the bound $2 * 10^{10}$.
Some statistics and supporting results based on the Prime k-tuplets conjecture of Hardy and Littlewood are presented.

1980 Mathematics Subject Classification (1985 Revision): 11Y99, 11P32.
Key Words & Phrases: Goldbach conjecture, vector computer, Cyber 205.
Note: This paper has been submitted for publication elsewhere.

1. Introduction

Almost 250 years ago, in 1742, Goldbach wrote a letter to Euler where he proposed the conjecture that every even number $2m$ is the sum of two odd primes (Goldbach considered 1 as a prime number).
In 1922, Hardy and Littlewood wrote ([3]): 'There is no reasonable doubt that the theorem is correct, and that the number of representations is large when m is large, but all attempts to obtain a proof have been completely unsuccessful.' The best theoretical result known at present was established in 1966 by the Chinese mathematician Chen Jing Run ([6]) who proved that every sufficiently large integer is the sum of a prime and a product of at most two primes. For the literature and history leading to this result, we refer the reader to [6].

The best *numerical* result, known to us, was established by Stein and Stein ([4]) who verified the conjecture up to 10^8. They found that for all even numbers n with $4 < n \leqslant 10^8$, there exists a partition $n = p + q$, where p and q are odd primes, such that $p \leqslant 1093$. The 'worst' case is $n = 60{,}119{,}912$ which has $p = 1093$ as the smallest prime p for which $n = p + q$. In [5], Stein and Stein have computed the *number* of such partitions for all even numbers $n \leqslant 150{,}000$ (and, later, up to 200,000).
Bohman and Froberg ([2]) have computed the number of partitions $n = p + q$ for all even numbers $n \leqslant 350{,}000$ and compared them to theoretical estimates.

We will use the following terminology: a *Goldbach partition* of an even number n is a representation $n = p + q$, $p \leqslant q$, where p and q are odd primes. A Goldbach partition $n = p + q$ with smallest p is called the *minimal* Goldbach partition of n; the smallest prime in the minimal Goldbach partition of n is denoted by $p(n)$. The number of Goldbach partitions of the even number n will be denoted by

Report NM-R8812
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

$G(n)$. For a given odd prime q we define $S(q)$ to be the smallest even number n for which $p(n) = q$. In particular, we are interested in $L(q,x)$, which is defined as the number of positive even integers n between 1 and x (inclusive) such that $p(n) = q$.

Some examples: the minimal Goldbach partition of 30 is $7+23$, hence $p(30) = 7$; $G(14) = 2$; $S(5) = 12$; $L(3,x) = \pi(x-3)-1$ where $\pi(x)$ is the number of primes $\leq x$.

In this paper we shall give an account of our verification of the Goldbach conjecture up to $2*10^{10}$. In our computations (on a Cyber 205) up to 10^{10}, we have also collected data concerning the functions $p(n)$, $S(q)$ and $L(q,x)$. We have *not* computed the function $G(n)$ since finding *all* Goldbach partitions of n is much more time-consuming than finding the minimal Goldbach partition. In Section 2 we describe the algorithms we have used and give some details about their implementation on the 1-pipe Cyber 205 of SARA (Academic Computer Centre Amsterdam). In Section 3 we present a selection of various numerical data. Theoretical results related to the numerical data, and based on the Prime k-tuplets conjecture of Hardy and Littlewood, are given in Section 4. Section 5 presents some results and conjectures obtained by the first named author, which are related to the Goldbach conjecture.

2. Algorithms and implementations

An obvious approach to verify the Goldbach conjecture up to some large bound is to split the work into smaller portions of a suitable length. Here, we describe our algorithms to verify the Goldbach conjecture (i.e., to compute p(N)), for all even numbers N in the interval [N1,N2]. The functions $S(q)$ and $L(q,x)$ are updated after the interval [N1,N2] has been dealt with.

For each even N in [N1,N2] we compute the minimal Goldbach partition by successively subtracting the odd primes 3,5,... from N and by checking if the difference is prime. This may be expressed in FORTRAN as follows. The array PR(I) is the I-th odd prime and PRIME(M) is a logical function yielding .TRUE. if M is prime and .FALSE. otherwise. PIND(N1:N2) is an integer array such that upon completion of the algorithm we have PIND(N) = I, where p(N) = PR(I), for N = N1, N1+2,...,N2. The number IMAX1 is the index of the largest (odd) prime used in the search for a Goldbach partition.

Goldbach algorithm I

```
C
C WE ASSUME THE INTEGER ARRAY PIND(N) HAS BEEN INITIALIZED TO ZERO
C
    DO 20 N = N1, N2, 2
C
C WE SUPPOSE THAT N1 AND N2 ARE EVEN;
C WE SEARCH FOR THE MINIMAL GOLDBACH PARTITION OF N
C
    DO 10 I = 1, IMAX1
     IF( PRIME( N-PR(I) ) )THEN
       PIND(N) = I
       GOTO 20
     END IF
 10  CONTINUE
C
C NO GOLDBACH PARTITION N = P + Q FOUND WITH P ≤ PR(IMAX1)
C INCREASE THE VALUE OF IMAX1
C
 20  CONTINUE
```

It turns out that IMAX1 need not be chosen too large. Stein and Stein's results ([4]) show that for the even numbers below 10^8, IMAX1 = 182 is sufficient and this number appears to grow very slowly with N. In our range ($N \leqslant 2 * 10^{10}$) we worked with IMAX1 = 400.

Algorithm I has two main drawbacks. The 10-loop cannot be vectorized on the Cyber 205, and therefore runs at scalar speed. However, by *interchanging* the 10- and 20-loops, vector speed can be achieved indeed. The second drawback is that during the execution of the loops the logical function PRIME is called various times for the *same* value of its argument.
Therefore, it is much more efficient to prepare a table of all the (large) primes between N1 - PR(IMAX1) and N2 - 3 in order to avoid checking the primality of these numbers more than once. This can be done efficiently by means of the sieve of Eratosthenes. These two improvements are incorporated in Algorithm II, which is much more efficient on vector computers than Algorithm I.
Algorithm II may be described as follows. First initialize the integer array PIND and the integer array ODDPR (ODDPR(M) := 1 if M is prime, and := 0 otherwise for $M \in$ [N1 - PR(IMAX1), N2 - 3]). The algorithm then determines *all* those even N $\in$ [N1, N2] with p(N) = 3; next all those with p(N) = 5, and so on. The efficiency of this process gradually decreases, because the minimal Goldbach partitions of more and more N will have been found as the algorithm proceeds. Therefore, besides IMAX1, a second parameter IMAX2 is used, which is the maximum number of steps taken to find all the even N with the same p(N). After these IMAX2 steps, those N for which no Goldbach partitions have been found yet are treated as in Algorithm I. In our range ($N \leqslant 2 * 10^{10}$), IMAX2 = 20 turned out to yield the highest efficiency. About 84.5 % of all $N \leqslant 10^{10}$ have $p(N) \leqslant$ PR(20) (where PR(20) = 73).

Goldbach algorithm II

```
C
C WE ASSUME THE INTEGER ARRAYS PIND AND ODDPR HAVE BEEN INITIALIZED
C
   DO 20 I = 1, IMAX2
     PRI = PR(I)
     DO 10 N = N1, N2, 2
       IF( PIND(N).EQ.0 .AND. ODDPR( N-PRI ) .EQ.1 ) PIND(N) = I
 10  CONTINUE
 20 CONTINUE
C
C TREAT THE EVEN N FOR WHICH PIND(N) IS STILL ZERO,
C I.E., FOR WHICH NO GOLDBACH PARTITION HAS BEEN FOUND YET
C
   DO 40 N = N1, N2, 2
     IF ( PIND(N).GT.0 ) GOTO 40
     DO 30 I = IMAX2 + 1, IMAX1
       IF( ODDPR( N-PR(I) ).EQ.1 )THEN
         PIND(N) = I
         GOTO 40
       END IF
 30  CONTINUE
 40 CONTINUE
```

The 10-loop in Algorithm II runs through the arrays PIND and ODDPR with increment 2. Of course, by a simple transformation this can easily be converted into a loop with increment 1, which is processed more efficiently on the Cyber 205. In our actual implementation we indeed worked with step 1, but in order not to confuse the reader with too many details, we have expressed the algorithm here in the above form. Our actual implementation also differs for another reason: the 10-loop can only be processed at vector speed if we express it in terms of a so-called WHERE-statement (we assume that the data transformation has been carried out enabling us to run through the arrays with step 1):

```
LLOOP = (N2 - N1)/2 + 1
PRIH = (PRI-1)/2
WHERE ( PIND(N1; LLOOP).EQ.0 .AND. ODDPR(N1-PRIH; LLOOP).EQ.1 )
  PIND(N1; LLOOP) = I
END WHERE
```

PIND(N1; LLOOP) is the vector with first element PIND(N1), second element PIND(N1 + 1), and so on, and its length is LLOOP = (N2 - N1)/2 + 1. When the above piece of FORTRAN 200 is executed, a so-called bit vector of length LLOOP is generated with a 1 on those places where the condition in the WHERE-statement is true and a 0 otherwise. Next, the constant I is assigned to those elements of PIND which correspond to a 1 in that bit vector.

With algorithm II we have verified the Goldbach conjecture up to 10^{10} in about 15 hours CPU-time on the Cyber 205 (checking the known range up to 10^8 took about 5 minutes CPU-time). We processed 10,000 (=LLOOP) even numbers at a time. The time to process the WHERE-statement above amounted to about 10,000 * 3 clock cycles = 10,000 * 3 * 20 nsec. = 0.6 msec. Since IMAX2 = 20, and $5*10^9$ even numbers had to be processed, the total time spent in the WHERE-statement part amounted to $20 * 0.0006 * 5*10^9/10^4 = 6000$ sec. The remainder of the 15 CPU-hours was spent on the processing (with scalar speed) of the even numbers N with $p(N) > 73$ and to the generation of the integer array ODDPR.

As suggested by Walter Lioen, Algorithm II can be speeded up further by changing the integer arrays PIND and ODDPR into bit arrays. The elements of bit arrays can have the values 0 or 1 and 64 elements are packed in one word of 64 bits. The Cyber 205 is able to perform binary operations on these vectors (like AND, OR) with a speed of 16 elements per clock cycle of 20 nsec. However, there is a price to pay, namely: if we convert the array PIND into a bit array, we can no longer store the index of the prime in the minimal Goldbach partition into this array, so that we have to be satisfied with the binary information: a 1 if a Goldbach partition has been found, a 0 if not (yet). The 20-loop now looks as follows (PIND has been converted into bit array PBIT and ODDPR into bit array ODDPRBIT):

BIT-VECTOR VERSION OF 20-LOOP IN ALGORITH II

```
   DO 20 I = 1, IMAX2
     PRIH = (PR(I)-1)/2
     PBIT(N1; LLOOP) = PBIT(N1; LLOOP) .OR. ODDPRBIT(N1-PRIH; LLOOP)
20 CONTINUE
```

Since this loop is executed much faster than the WHERE-statement above, the value of IMAX2 must be increased, in order to reach the optimal performance for this loop. We found IMAX2 = 100 to yield the best results. After this loop, the remaining even N for which $p(N) > PR(100)$ were processed with the 40-loop of Algorithm II (with PIND replaced by PBIT).

For those N which have $p(N) > 547$ (=PR(100)), we have, of course, collected the same data as we did in the original version of Algorithm II.

With the help of the bit vector version of Algorithm II we have extended the verification of the Goldbach conjecture from 10^{10} to $2*10^{10}$ in about 9000 sec. CPU-time on the Cyber 205. We have

checked 50,000 even numbers at a time. The time needed to run the bit vector statement above was about 50,000 * 20/16 = 0.0625 msec. The total range of even numbers between 10^{10} and $2*10^{10}$ took $0.0625 * 100 * 5*10^9/5*10^4 = 625$ sec.
The scalar processing of the remaining even N took only 130 sec. and the generation of the (large) primes required about 8245 sec. (this means an average prime generation speed of more than 50,000 primes per second in the interval $(10^{10}, 2*10^{10}])$.

3. Numerical results

In this section we present some tables of numerical results selected from our computations. Table 1 presents q, $S(q)$ and $L(q, 10^{10})$ for the odd primes q below 100 and similar data for some selected primes > 100. In addition, the cumulative frequency percentages are given of the numbers of even numbers N below 10^{10} for which $p(N) \leqslant q$.

Table 1

I	$PR(I) =: q$	$S(q)$	$L(q, 10^{10})$	% of even $N \leqslant 10^{10}$ for which $p(N) \leqslant q$
1	3	6	455,052,510	9.10
2	5	12	427,649,831	17.65
3	7	30	400,229,833	25.66
4	11	124	350,840,599	32.68
5	13	122	320,898,559	39.09
6	17	418	276,936,926	44.63
7	19	98	267,951,521	49.99
8	23	220	226,031,301	54.51
9	29	346	199,319,687	58.50
10	31	308	201,862,574	62.54
11	37	1,274	170,425,547	65.94
12	41	1,144	147,748,455	68.90
13	43	962	138,381,620	71.67
14	47	556	118,054,048	74.03
15	53	2,512	101,504,888	76.06
16	59	3,526	90,311,298	77.86
17	61	1,382	106,906,523	80.00
18	67	1,856	91,418,970	81.83
19	71	4,618	68,641,994	83.20
20	73	992	69,457,153	84.59
21	79	3,818	69,182,416	85.98
22	83	7,432	53,268,347	87.04
23	89	12,778	47,140,891	87.98
24	97	5,978	51,345,000	89.01
29	113	19,696	26,537,015	92.63
30	127	6,008	31,047,922	93.25
55	263	485,326	2,842,690	99.00
56	269	407,128	2,524,569	99.05
57	271	137,708	4,557,244	99.14
65	317	686,638	1,351,658	99.51
66	331	128,168	2,447,734	99.56

103	569	17,726,098	65,419	99.97
104	571	4,493,498	169,264(2.59)	99.97
108	599	15,860,818	41,965	99.98
109	601	1,077,422	122,261 (2.91)	99.98

Table 2 presents counts of $L(q,.)$ in the intervals $(10^{10}, 10^{10}+10^9]$ and $(2*10^{10}-10^9, 2*10^{10}]$, for some odd primes $q>PR(100)$ (in Section 2 we have explained why we have chosen not to collect such data for the first 100 odd primes). The numbers in parentheses in columns 3 and 4 are quotients of consecutive elements in these columns. A comparison of these two columns shows that these quotients are reasonably stable (also compare the quotients in Table 1 on the lines with $I = 104$ and $I = 109$).

Table 2

I	$PR(I) =: q$	$L(q, 11^* 10^9)$ - $L(q, 10^* 10^9)$	$L(q, 20^* 10^9)$- $L(q, 19^* 10^9)$
101	557	12,981	15,822
102	563	10,284 (1.26)	12,438 (1.27)
103	569	9,057 (0.88)	11,101 (0.89)
104	571	22,794 (2.52)	26,904 (2.42)
105	577	14,957 (0.66)	18,089 (0.67)
106	587	8,511 (0.57)	10,718 (0.59)
107	593	6,651 (0.78)	8,349 (0.78)
108	599	5,898 (0.89)	7,476 (0.90)
109	601	16,661 (2.82)	19,696 (2.63)
110	607	11,148 (0.67)	13,421 (0.68)
151	881	358	540
152	883	539 (1.51)	736 (1.36)
153	887	309 (0.57)	374 (0.51)
154	907	499 (1.61)	693 (1.85)
155	911	250 (0.50)	385 (0.56)
156	919	538 (2.15)	702 (1.82)
157	929	217 (0.40)	293 (0.42)
158	937	337 (1.55)	447 (1.53)
159	941	207 (0.61)	242 (0.54)
160	947	177 (0.86)	250 (1.03)

In Table 3 we give even numbers n with corresponding $p(n)$ such that $p(m)<p(n)$ for all even $m<n$. This is an extension of a table presented by BOHMAN and FROBERG ([2]). We also list the quotients $\log(n)/\log(p(n))^2$. After a clear decreasing trend in the beginning of this table, this quotient shows an increasing tendency at the end of the table. Table 3 implies that for all even $n \leqslant 2*10^{10}$ we have $p(n) \leqslant 2029$.
It should be added that the larger primes occur extremely rarely as $p(n)$-values. For example, there are only six even n below $2*10^{10}$ for which $p(n)>1861$, viz., the three given in Table 3 and the three given by: $p(18,113,547,184) = 1871$, $p(19,326,123,574) = 2003$ and $p(15,317,795,894) = 2017$.

Table 3

n	$p(n)$	*quot*	n	$p(n)$	*quot*	n	$p(n)$	*quot*
6	3	1.485	113672	313	0.353	113632822	1163	0.372
12	5	0.959	128168	331	0.349	187852862	1321	0.369
30	7	0.898	194428	359	0.352	335070838	1427	0.372
98	19	0.529	194470	383	0.344	419911924	1583	0.366
220	23	0.549	413572	389	0.364	721013438	1789	0.364
308	31	0.486	503222	523	0.335	1847133842	1861	0.376
556	47	0.426	1077422	601	0.339	7473202036	1877	0.400
992	73	0.375	3526958	727	0.347	11001080372	1879	0.407
2642	103	0.367	3807404	751	0.346	12703943222	2029	0.401
5372	139	0.353	10759922	829	0.359			
7426	173	0.336	24106882	929	0.364			
43532	211	0.373	27789878	997	0.360			
54244	233	0.367	37998938	1039	0.362			
63274	293	0.343	60119912	1093	0.366			

4. The asymptotic behaviour of $L(q,N)$

A look at Table 1 shows that the function $L(q,\ 10^{10})$ is generally decreasing, as may be expected, although not monotonically: in particular, we often see that, when q and $q+2$ are (twin) primes, then $L(q,\ 10^{10})<L(q+2,10^{10})$! In fact, our counts show that for all twin primes $(q,q+2)$ with $q<1800$ we have $L(q,\ 10^{10})<L(q+2,10^{10})$, except for the pairs (3, 5), (5, 7), (11, 13), (17, 19) and (41, 43). More general, a similar behaviour can be observed for primes q and $q+d$, where $d\equiv 2 \pmod 6$.

In this Section we present a theoretical result, based on the truth of the Prime k-tuplets conjecture of Hardy and Littlewood, which explains, at least asymptotically, this behaviour of the function $L(q,N)$. We recall

The Prime k-tuplets conjecture (Hardy and Littlewood [3])
Suppose that $b_1,b_2,...,b_k$ are given integers, and let $P_{b_1,b_2,...,b_k}(N)$ be the number of positive integers n with $1\leqslant n\leqslant N$ such that $n+b_1,n+b_2,...,n+b_k$ are all prime numbers. Then, as $N\to\infty$,

$$P_{b_1,b_2,...,b_k}(N) = C(b_1,...,b_k)\frac{N}{(\log N)^k}\{1+o(1)\}$$

where

$$C(b_1,...,b_k) = \prod_{p\ prime} (1-\frac{1}{p})^{-k}(1-\frac{\omega_{b_1,b_2,...,b_k}(p)}{p})$$

and $\omega_{b_1,b_2,...,b_k}(p)$ is the number of distinct residue classes $(\text{mod}\ p)$ which contain some b_i.
Now, we have the following

Theorem. *Suppose the Hardy-Littlewood Prime k-tuples conjecture is true. Then, for a given odd prime q, we have*

$$L(q,N) = \pi(N)-C\frac{N}{\log^2 N}E(q)+o(\frac{N}{\log^2 N})$$

where

$$E(q) = \sum_{\substack{r\ \text{odd}\ prime \\ r<q}} \prod_{\substack{p\mid q-r \\ p\geqslant 3}} (p-1)/(p-2)$$

and

$$C = 2 \prod_{p \text{ odd } prime} \{1-(p-1)^{-2}\}.$$

COROLLARY. *As* $N \to \infty$, $L(q,N) > L(q',N)$ *iff* $E(q) < E(q')$ *and* $L(q,N) < L(q',N)$ *iff* $E(q) > E(q')$.

PROOF OF THE THEOREM. Suppose $p_1 = 3, p_2, ..$ is the sequence of odd primes. Then $L(p_r,N) = \sharp\,\{\text{even } n \leqslant N : n - p_r \text{ is prime and } n - p_j \text{ is not prime}, \forall j \leqslant r-1\}$ $=$

$$= \sum_{j=0}^{r-1} (-1)^j \sum_{\substack{J \subset \{1,2,\ldots,r-1\} \\ |J|=j}} P_{D(J)}(N) + O(1)$$

by the combinatorial sieve, where $D(J) = \{0\} \cup \{p_r - p_i : i \in J\}$. Now, by the Prime k-tuplets conjecture, we have

$$P_K(N) = O(\frac{N}{\log^2 N}) \text{ if } |K| \geqslant 3,$$

$$P_{0,2k}(N) = CD_{2k}\frac{N}{\log^2 N}\{1+o(1)\}$$

where

$$C = 2\prod_{\substack{p \text{ } prime \\ p>2}} \frac{1-2/p}{(1-1/p)^2} \text{ and } D_k = \prod_{\substack{p|k \\ p \geqslant 3}} \frac{1-1/p}{1-2/p},$$

and $P_0(N) = \pi(N)$.
Therefore,

$$L(p_r,N) = \pi(N) - C\frac{N}{\log^2 N}\sum_{j=1}^{r-1} D_{p_r - p_i} + o(\frac{N}{\log^2 N}). \qquad \square$$

Table 4

q	$E(q)$	q	$E(q)$	q	$E(q)$	q	$E(q)$
3	0.000	103	41.202	239	85.013	389*	127.790
5	1.000	107	43.468	241	85.705	397	129.358
7	2.000	109*	43.104	251	88.854	401	132.535
11	4.000	113	47.569	257	91.997	409*	132.082
13	5.333	127	47.715	263	93.586	419	135.539
17	7.533	131	51.124	269	95.451	421*	133.988
19	8.200	137	53.070	271**	92.937	431	139.807
23	10.667	139*	51.532	277	95.223	433*	139.480
29	12.535	149	56.154	281	100.355	439	140.896
31	12.824	151*	54.716	283*	99.417	443	145.233
37	15.358	157	58.547	293	104.247	449	146.239
41	17.437	163	60.156	307*	102.477	457*	145.811
43	18.683	167	62.532	311	108.060	461	150.366
47	21.111	173	65.574	313*	106.448	463*	149.524
53	23.292	179	66.425	317	110.897	467	154.053
59	25.050	181*	64.879	331*	107.856	479	156.574
61*	24.340	191	70.000	337	111.243	487*	154.574
67	26.695	193	70.375	347	116.850	491	158.973
71	30.084	197	73.578	349*	113.385	499*	156.746
73	30.825	199*	71.979	353	119.700	503	163.340
79	31.494	211	74.249	359	120.019	509	165.522
83	35.046	223	78.235	367	120.386	521	168.123
89	37.066	227	80.539	373	122.004	523*	164.724
97	37.321	229*	80.291	379*	121.753	541	164.872
101	40.689	233	83.535	383	128.371	547	167.976

In order to compare the Corollary with our numerical data, we have computed $E(q)$ for the first 2000 odd primes. In Table 4 we present these values for the first 100 odd primes. An asterisk indicates that the corresponding E-value is *smaller* than the previous one. In one case, viz., $q = 271$, $E(q)$ is also smaller than the 'pre-previous' one (cf. the corresponding entries in Table 1).

With respect to the various prime differences d among the first 2000 odd primes, we have counted in Table 5 how often $E(q)<E(q+d)$ and how often $E(q)>E(q+d)$. We have grouped the counts according to the residues of d (mod 6). In the cases where one of the two categories is small compared to the other, we have explicitly given all the prime pairs belonging to the smaller category.

Table 5

d	# of prime pairs$(q,q+d)$ with $E(q)<E(q+d)$	$E(q)>E(q+d)$	exceptional cases
2	12	290	(3,5) (56,7) (11,13) (17,19) (29,31) (41,43) (71,73) (101,103) (191,193) (239,241) (1871,1873) (2381,2383)
8	3	167	(89,97) (359,367) (389,397)
14	3	93	(113,127) (839,853) (2039,2053)
20	0	33	
26	0	10	
32	0	2	
44	0	1	
4	316	1	(1867,1871)
10	185	0	
16	58	0	
22	32	0	
28	15	0	
34	6	0	
6	330	141	
12	130	46	
18	49	24	
24	23	5	
30	17	2	
36	2	2	
42	1	0	

For the first 100 odd primes, we have counted how often our actual counts of $L(q, 10^{10})$ match with our Corollary (for consecutive primes q and q'). In 87 of the 99 cases we observe a perfect match between theory and practice. In the 12 remaining cases we find $L(q, 10^{10})<L(q', 10^{10})$ *and* $E(q)<E(q')$. Of these 12 prime pairs, 9 occur as exceptional cases in Table 5.

5. Discussion

A simple explanation of our empirical observation that $E(q)>E(q+2)$ for so many of the small prime pairs $q,q+2$ (and, more general, for prime pairs $q,q+d$ with $d = 2 \pmod 6$) reads as follows. Recall that

$$E(q) = \sum_{\substack{r<q \\ r \text{ odd prime}}} D_{q-r} .$$

If $3|k$ then $D_k \geqslant 2$. However, if $3 \nmid k$ then it is easy to see that in order to have $D_k \geqslant 2$, k should satisfy $k \geqslant 5.7.11.13.17$ ($= 85085$). Hence, we may expect D_k to contribute a lot more to $E(q)$ in those cases where $3|k$, than when $3 \nmid k$. Now let, as usual, $\pi(x;a,b)$ be the number of primes $\leqslant x$ which are congruent to b (mod a). It is well-known that $\pi(x;3,2)>\pi(x;3,1)$ for $x<6*10^{12}$ ([1]). So, if q is a prime $\equiv b \pmod 3$ there are $\pi(q-1;3,b)$ primes $r<q$ such that $3|q-r$. This number is greater (when q is small) when $b = 2$, than when $b = 1$. Now, for any prime pair q, $q+d = p$ where $d \equiv 2 \pmod 6$ we must have $q \equiv 2 \pmod 3$ and $p \equiv 1 \pmod 3$, and we *should* expect $E(q)>E(p)$.

On the Prime k-tuplets conjecture, we can prove that, on average, we have $E(q) = C\pi(q)\{1+o(1)\}$, where

$$C = \prod_{\substack{p \text{ prime} \\ p \geq 3}} \{1+\frac{1}{(p-1)(p-2)}\} \ (= 1.742725...).$$

An inspection of the values of $E(q)/\pi(q)$ for the first 2000 odd primes shows a good agreement with this result: from the 271-st odd prime q $(=1747)$ onwards, $E(q)/\pi(q)$ fluctuates between 1.70 and 1.75.

On probabilistic grounds we conjecture that $\forall n \geq 10$, $p(n) \ll \log^2 n \log\log n$. From Table 3 we derive that $p(n)/ (\log^2 n \log\log n) < 1.603$ for all $n \leq 2*10^{10}$.

On the Prime k-tuplets conjecture, we have

$$\sharp\{n \leq N : p(n) \leq Q\} = \pi(Q)\pi(N)(1-\frac{C^*\pi(Q)}{\log N}\{1+o(1)\})$$

where

$$C^* = 2\prod_{\substack{p \text{ prime} \\ p \geq 3}} \{1-(p-1)^{-3}\} \ (= 1.710784...).$$

Let $p_1 = 3$, $p_2 = 5$,... be the successive odd primes, and define

$$F_k(N) := \sharp\{n \leq N : E(p_n) \leq E(p_{n+k})\}.$$

CONJECTURE: For any fixed integer $k \neq 0$, $F_k(N) \sim N/2$, as $N \to \infty$.

If we define, slightly different from $G(n)$ in Section 1, $G^*(n) := \sharp\{p<q \text{ both prime}: p+q=n\}$, then, trivially, we have $0 \leq G^*(n) \leq \pi(n)-\pi(n/2)$. Now $G^*(210) = \pi(210)-\pi(105)$ and Pomerance conjectured that:

$$\forall n \geq 212,\ G^*(n) \leq \pi(n)-\pi(n/2)-1.$$

We can prove that if $n \geq 10^{520}$ then $G^*(n) \leq \pi(n)-\pi(n/2)-1$.

ACKNOWLEDGEMENT

We like to express our gratitude to Walter Lioen for his help with the optimization of our algorithm on the Cyber 205.

REFERENCES

[1] C. BAYS and R.H. HUDSON, *Details of the first regions of integers x with* $\pi_{3,2}(x) < \pi_{3,1}(x)$, Math. Comp. 32, 1978, 571-576.

[2] J. BOHMAN and C.-E. FROBERG, *Numerical results on the Goldbach conjecture*, BIT 15, 1975, pp. 239-243.

[3] G.H. HARDY and J.E. LITTLEWOOD, *Some problems of 'partitio numerorum'*; III: *On the expression of a number as a sum of primes*, Acta Math. 44, 1922, pp. 1-70.

[4] M.L. STEIN and P.R. STEIN, *Experimental results on additive 2-bases*, Math. Comp. 19, 1965, pp. 427-434.

[5] M.L. STEIN and P.R. STEIN, *New experimental results on the Goldbach conjecture*, Math. Mag. 38, 1965, pp. 72-80.

[6] WANG YUAN (ed.), *Goldbach conjecture*, World scientific, Singapore, 1984. (This is a collection of original papers concerning the Goldbach conjecture with an Introduction of 15 pages. Papers originally written in Chinese, French, German or Russian have been translated into English.)