Predicting the Sieving Effort for the Number Field Sieve

Willemien Ekkelkamp

CWI, Amsterdam & UL, Leiden

Overview

- Aim of the method
- Number Field Sieve (summary)
- Technical details of the method
- Examples

Goal

- Predict the number of relations needed for factoring a given number N in practice.
- In practice := for a given implementation and for a given choice of the parameters in the NFS.
- The prediction should not be based on the number of relations used for factoring a number of comparable size.

NFS

- Polynomial selection
  - $f_1(m) \equiv f_2(m) \equiv 0 (\text{mod } N)$.
    - $f_1(x)$: linear polynomial (rational side).
    - $f_2(x)$: higher degree polynomial (algebraic side).
  - SNFS / GNFS

NFS

- Sieving
  - Choose a factorbase bound (F) and a large prime bound (L).
  - Locate pairs (a, b) such that $\gcd(a, b) = 1$ and such that $b^{\deg(f_1)} f_1(a/b)$ and $b^{\deg(f_2)} f_2(a/b)$ both have all their prime factors below F or at most two prime factors between F and L (so-called large primes).
  - Line sieving / lattice sieving.

NFS

- Linear algebra
  - Singleton removal.
  - Find a set of relations such that the product on both the rational and algebraic side is a square.

NFS

- Linear algebra
  - Singleton removal.
  - Find a set of relations such that the product on both the rational and algebraic side is a square.
- Square root
  - Find the square root of the two products.
  - Factor the number; in case of a trivial factorization: continue with the next set.

Outline of the method

- Short sieving test.
- Analysis of the relations from this test.
- Simulate relations (fast):
  - Functions that approximate the underlying distribution of the large primes.
  - Random number generator.
- Remove singletons.
- Stop simulating relations as soon as the number of relations after singleton removal exceeds the number of primes in the relations.

Short sieving test

- Representative selection.
- Sieving points should be spread over the entire sieving area.
- Takes about ten minutes for a 120-digit N. (explained later)

Analysis of the relations / Simulation

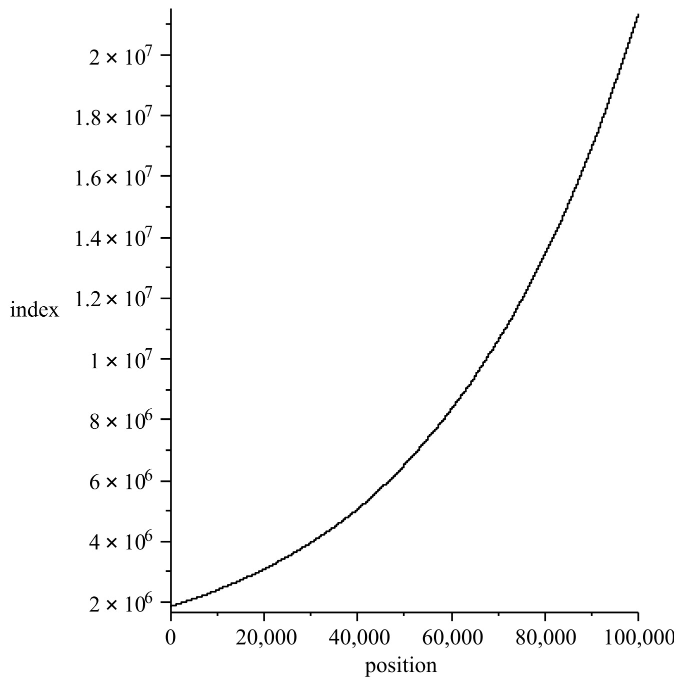
- line sieving / lattice sieving
- Divide relations into nine sets, based on the number of large primes: $r_i a_j$ for $i, j \in 0, 1, 2$.
- The mutual ratios of their cardinalities determine the ratios by which we will simulate the relations.

Analysis of the relations / Simulation

- $r_0 a_0$
  - Count the number of relations in this set.
- $r_1 a_0$
  - To avoid expensive prime tests, switch to indices of primes ($i_p = \pi(p)$):
    - look-up table,
    - approximation $i_p \approx \frac{p}{\log p} + \frac{p}{\log^2 p} + \frac{2p}{\log^3 p}$. (Panaitopel, 2000)

Analysis of the relations / Simulation

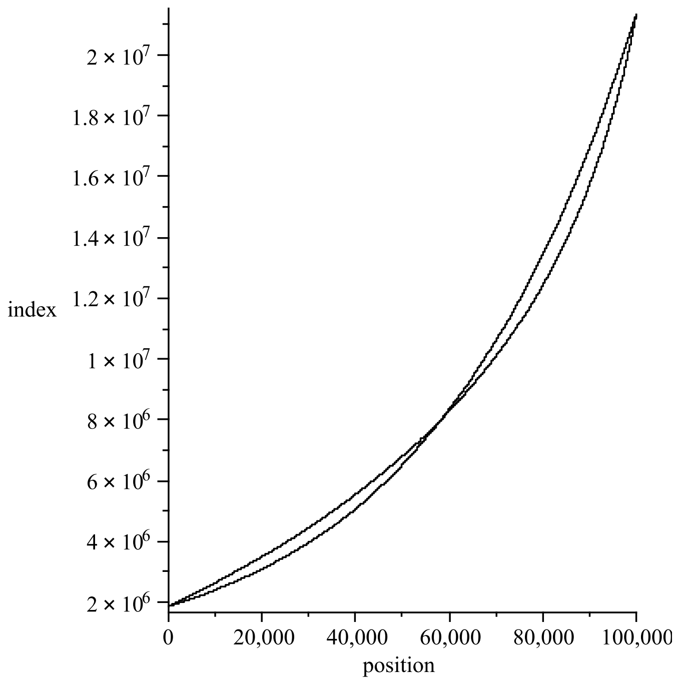
- $r_1 a_0$



- $G(x) = i_F - a\log(1 - x(1 - e^{\frac{i_F - i_L}{a}})), 0 \leq x \leq 1$
  - a = average of the indices,
  - i_F and i_L are the indices related to F and L,
  - $G(0) = i_F$, $G(1) = i_L$.

Analysis of the relations / Simulation

- $r_1 a_0$



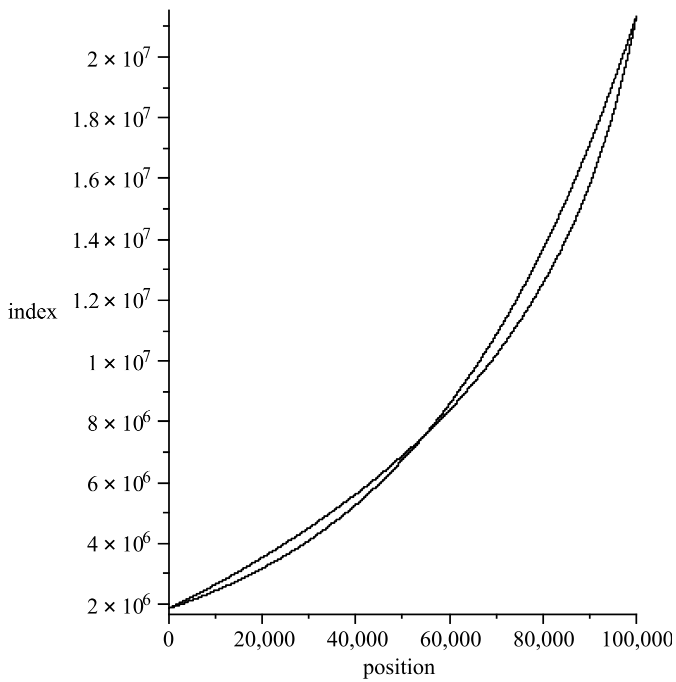
- $G(x) = i_F - a\log(1 - x(1 - e^{\frac{i_F - i_L}{a}})), 0 \leq x \leq 1$
  - $G(x)$ is the inverse of an exponential distribution function, which approximates the line of data.
  - Result after singleton removal was satisfactory.

Analysis of the relations / Simulation

- $r_0 a_1$
  - Algebraic primes: not all primes can occur, each prime that does occur can have up to $\deg(f_2)$ different roots.
  - Heuristically the amount of pairs $(prime, root)$ with $F < prime < L$ is about equal to the amount of primes between F and L.
  - Same approach as for $r_1 a_0$.

Analysis of the relations / Simulation

- $r_0 a_1$



Analysis of the relations / Simulation

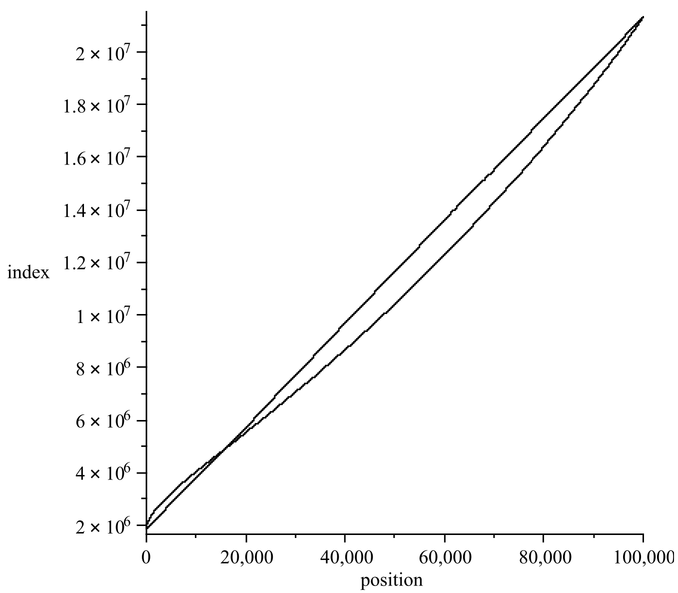
- $r_1 a_1$
  - The value of the index on the rational side is assumed to be independent of the value of the index on the algebraic side.
  - Combine the approaches of $r_1 a_0$ and $r_0 a_1$.

Analysis of the relations / Simulation

- $r_1 a_1$
  - The value of the index on the rational side is assumed to be independent of the value of the index on the algebraic side.
  - Combine the approaches of $r_1 a_0$ and $r_0 a_1$.
- $r_2 a_0$
  - Two rational primes q_1 and q_2, $q_1 > q_2$.
  - Observation q_1: linear distribution.

Analysis of the relations / Simulation

- $r_2 a_0,\ q_1$



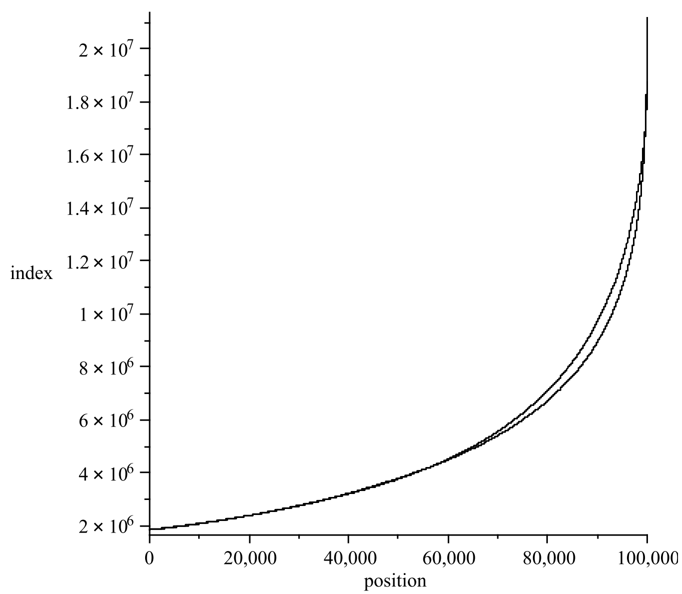
- $H_1(x) = i_F + x(i_L - i_F)$
  - $H_1(x)$ approximates the inverse of the line of observation.

Analysis of the relations / Simulation

- $r_2 a_0$, q_2
  - Exponential distribution.
  - Average value; based on q_2-indices < q_1.
  - List of averages a_{q_2}, where $a_{q_2}[j]$ contains the average of the first j q_2-indices.
  - $H_2(x) = i_F - a_{q_2}[j] \log(1 - x(1 - e^{\frac{i_F - i_L}{a_{q_2}[j]}}))$

Analysis of the relations / Simulation

- $r_2 a_0$, q_2
  - First compute q_1, look up which average value to use and compute q_2.



Analysis of the relations / Simulation

- $r_0 a_2$
  - Same approach as used for $r_2 a_0$.
- $r_1 a_2$
  - $r_1 a_0$, $r_0 a_2$
- $r_2 a_1$
  - $r_2 a_0$, $r_0 a_1$
- $r_2 a_2$
  - $r_2 a_0$, $r_0 a_2$

Adjustment for lattice sieving

Same model, add a special prime to each relation as follows:

Adjustment for lattice sieving

Same model, add a special prime to each relation as follows:

- Sieve test: average number of relations per pair $(special\ prime, root)$.
- Total number of relations to simulate.

Adjustment for lattice sieving

Same model, add a special prime to each relation as follows:

- Sieve test: average number of relations per pair $(special\ prime, root)$.
- Total number of relations to simulate.
- Select an appropriate interval.
- Divide this interval in a (small) number of sections.

Adjustment for lattice sieving

Same model, add a special prime to each relation as follows:

- Sieve test: average number of relations per pair $(special\ prime, root)$.
- Total number of relations to simulate.
- Select an appropriate interval.
- Divide this interval in a (small) number of sections.
- Per section select randomly the special primes.

Adjustment for lattice sieving

Same model, add a special prime to each relation as follows:

- Sieve test: average number of relations per pair $(special\ prime, root)$.
- Total number of relations to simulate.
- Select an appropriate interval.
- Divide this interval in a (small) number of sections.
- Per section select randomly the special primes.

This covers the entire interval of special primes, but leaves enough variation in the amount of relations per special prime.

Stop Criterion

- Goal: find dependencies in a matrix.
- Stop criterion: the number of relations after singleton removal exceeds the number of different primes that occur in the remaining relations.

Stop Criterion

- Goal: find dependencies in a matrix.
- Stop criterion: the number of relations after singleton removal exceeds the number of different primes that occur in the remaining relations.
- Oversquareness $O_r := \frac{n_r}{n_l + n_F - n_f} \times 100\,\%$,
  - n_r: number of relations after singleton removal,
  - n_l: number of different large primes after singleton removal,
  - n_F: number of primes in the factorbase $(\pi(F_{rat}) + \pi(F_{alg}))$,
  - n_f: number of free relations from factorbase elements $(\frac{1}{g}\pi(\min(F_{rat}, F_{alg}))$.

Stop Criterion

- Possible choices for O_r ($100\,\%$, $102\,\%$).
- To minimize the resulting matrix, O_r should be larger.

Stop Criterion

- Possible choices for O_r ($100\,\%$, $102\,\%$).
- To minimize the resulting matrix, O_r should be larger.
- Lattice sieving / duplicates.
  - Act as if there are no duplicates.
  - Add a certain percentage to the number of necessary relations (Aoki, Franke, Kleinjung, Lenstra, Osvik, 2007).
  - Basic idea: run a sieve test and find out which relations have more than one prime in the special primes interval.
  - If such a relation would be found by more than one lattice, than this gives a duplicate relation.

Experiments

- Type 1: the complete data set for factoring N is known, simulate the same number of relations based on $0.1\,\%$ of the relations.
- Type 2: assume only $0.1\,\%$ is given; simulate relations until $O_r \geq 100\,\%$.

Experiments

- Type 1: the complete data set for factoring N is known, simulate the same number of relations based on $0.1\,\%$ of the relations.
- Type 2: assume only $0.1\,\%$ is given; simulate relations until $O_r \geq 100\,\%$.
- $0.1\,\%$?
  - We started experiments based on $100\,\%$ data and lowered the percentage until the result after singleton removal was too far from the real data.
  - In some cases we could go to 0.01% and still get good results.
  - Better solution is probably based on using the law of large numbers (work in progress).

Experiments: GNFS (line sieving)

- Parameters

number	# dec. digits	F	L	g	$n_F - n_f$
13,220+	117	30M	400M	120	3 700 941

Experiments: GNFS (line sieving)

- Parameters

number	# dec. digits	F	L	g	$n_F - n_f$
13,220+	117	30M	400M	120	3 700 941

- Type 1 experiment

13,220+	Original data	Simulated data
# relations before s.r.	35 496 483	35 496 483
# relations after s.r.	21 320 864	21 394 640 ($0.35\,\%$)
# large primes after s.r.	13 781 518	13 950 420 ($1.22\,\%$)
oversquareness ($\%$)	121.96	121.21 ($-0.61\,\%$)

Experiments: GNFS (line sieving)

- Timings

GNFS	13,220+
simulation (sec.)	224
singleton removal (sec.)	927
sieving (hrs.)	316

Experiments: GNFS (line sieving)

- Timings

GNFS	13,220+
simulation (sec.)	224
singleton removal (sec.)	927
sieving (hrs.)	316

- Type 2 experiment

# rel. before s.r.	O_r S (%)	O_r O (%)	rel. diff. (%)
28M (13,220+)	99.66	99.87	–0.21
29M (13,220+)	103.15	103.29	–0.14

Experiments: SNFS (line sieving)

- Parameters

number	# dec. digits	F	L	g	$n_F - n_f$
80,123–	150	55M	450M	18	6 383 294

Experiments: SNFS (line sieving)

- Parameters

number	# dec. digits	F	L	g	$n_F - n_f$
80,123–	150	55M	450M	18	6 383 294

- Type 1 experiment

80,123–	Original data	Simulated data
# relations before s.r.	36 552 655	36 552 655
# relations after s.r.	20 288 292	20 648 909 ($1.78\,\%$)
# large primes after s.r.	12 810 641	12 973 952 ($1.27\,\%$)
oversquareness ($\%$)	105.70	106.67 ($0.92\,\%$)

Experiments: SNFS (line sieving)

- Timings

SNFS	80,123–
simulation (sec.)	223
singleton removal (sec.)	771
sieving (hrs.)	200

Experiments: SNFS (line sieving)

- Timings

SNFS	80,123–
simulation (sec.)	223
singleton removal (sec.)	771
sieving (hrs.)	200

- Type 2 experiments

# rel. before s.r.	O_r S (%)	O_r O (%)	rel. diff. (%)
34M (80,123–)	99.93	98.66	1.29
35M (80,123–)	102.82	101.50	1.30

Experiments: 7,333- (lattice sieving)

- Parameters

	7,333−
# dec. digits	177
F	16 777 215
L	250 000 000
special primes	[16 777 333, 29 120 617]
	[60 000 013, 73 747 441]
g	6
$n_F - n_f$	1 976 740

Experiments: 7,333- (lattice sieving)

- Experiments

# rel. before s.r.	O_r S (%)	O_r O (%)	rel. diff. (%)
17M	98.34	97.45	0.91
18M	103.96	103.08	0.85
25 112 543	135.39	136.64	−0.91

Implementation

- CWI line siever
- Bruce Dodson (lattice sieving)
- Thorsten Kleinjung (lattice sieving)

Conclusions / future work

- By specifying a model for the large primes in the relations, we can simulate relations efficiently.
- Experiments show that what we find with our simulation and singleton removal, agrees within $2\,\%$ with real sieving data.

Conclusions / future work

- By specifying a model for the large primes in the relations, we can simulate relations efficiently.
- Experiments show that what we find with our simulation and singleton removal, agrees within $2\,\%$ with real sieving data.
- Find the correct model for the lattice sieve data sets of Kleinjung.
- Find a theoretical explanation for the occurrence of the various distributions.
- What is the optimal oversquareness for minimizing the resulting matrix.