Centrum voor Wiskunde en Informatica
REPORT*RAPPORT*

The number of relations in the quadratic sieve algorithm

H. Boender

Department of Numerical Mathematics

NM-R9622 1996

Report NM-R9622
ISSN 0169-0388

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

The Number of Relations in the Quadratic Sieve Algorithm

Henk Boender
email: henkb@wi.leidenuniv.nl

CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

Abstract

The subject of our study is the single large prime variation of the quadratic sieve algorithm. We derive a formula for the average numbers of complete and incomplete relations per polynomial, directly generated by the algorithm. The number of additional complete relations from the incomplete relations is then computed by a known formula. Hence practical hints for the optimal choice of the parameter values can be derived. We further compare theoretical estimates for the total number of smooth integers in an interval with countings in practice.

AMS Subject Classification (1991): 11A51, 11Y05
CR Subject Classification (1991): F.2.1
Keywords & Phrases: Factorization, Multiple Polynomial Quadratic Sieve, Vector supercomputer, Cluster of work stations

1. Introduction

We assume that the reader is familiar with the multiple polynomial quadratic sieve algorithm [Bre89, Pom85, PST88, Sil87, RLW89]. We consider the single large prime variation of the algorithm and write MPQS for short. If we can predict the rate by which the complete relations in MPQS are generated as a function of the various parameters in the algorithm, then we can determine a good choice of the parameter values. Here we give a method to do so.

The outline of the paper is as follows. Section 2 contains notation and preparation. Counting and approximating the number of smooth integers in an interval are the subjects of Section 3. In Sections 4 and 5 we give an approximation of the numbers of complete and incomplete relations per polynomial in the single large prime version of the quadratic sieve algorithm. We present numerical results in Section 6. In Section 7 we give a formula to approximate the number of complete relations descendent from a given number of incomplete relations. We analyze the total amount of work in Section 8 and draw conclusions in Section 9.

2. Notation and preparation

We write $\log x/\log y$ for $(\log x)/\log y$. In the sequel u denotes $\log x/\log y$. Euler's constant is denoted by γ $(= 0.5772...)$.

Let m be an integer. We write $P(m)$ for the largest positive prime factor of m. If $P(m) \leq y$ then m is called y–smooth. The number of y–smooth positive integers $\leq x$ is denoted by $\psi(x, y)$.

Furthermore we denote by

k	:	multiplier,
n	:	k times the number to be factored,
$\mathcal{F}$	:	factor base,
B	:	upper bound for the elements in the factor base,
L	:	upper bound for the large primes,
M	:	radius of the sieve interval $(= [-M, M])$,
T	:	sieve threshold,
$W(x)$	=	$ax^2 - 2bx - c$: sieve polynomial,
Γ	:	graph of W.

We assume that the number to be factored is composite, does not contain prime divisors $\leq B$ and that n is not a perfect power. The multiplier is chosen such that $n \equiv 1 \bmod 8$. Hence 2 is a member of the factor base. In practice we choose B between 10^5 and 10^6 and L between $10B$ and $100B$. If n has about 100 decimal digits, then M is about 10^7. We say that an integer is a W–value if it equals $W(x)$ for some integer x in the sieve interval. The integers a, b and c satisfy the following conditions:

$$a \approx \sqrt{2n}/M, \quad |b| < a/2, \quad b^2 + ac = n.$$

Let $-R$ be the minimum and S the maximum of W on the sieve interval. The minimum of W is attained at $x = b/a$ and the maximum at the boundary of the sieve interval. We have $aW(x) = (ax - b)^2 - n$ so that $aR = -aW(b/a) = n$. Furthermore, $aS = aW(\pm M) = (\pm aM - b)^2 - n \approx (\pm\sqrt{2n} - b)^2 - n \approx n$ since $|b| < a/2 < \sqrt{n}/M$ and M is large. Thus $R \approx S \approx n/a \approx M\sqrt{n/2}$. Note that also $c \approx S$. Theoretically $T = \log(\frac{1}{3}M\sqrt{n/2}) - \log L$, but in practice we have to lower this value a bit to get more relations per time unit.

We use the term complete relation for a B–smooth W–value. An incomplete relation is a W–value y that is divisible by a large prime q, $B < q \leq L$, such that y/q is B–smooth. Let t_1 and t_2 be the number of complete and incomplete relations, respectively.

There are relatively few W–values in the interval $(-e^T, e^T)$ compared with the total number $2M+1$ of W–values. Indeed, if $W(x) = y$ and $x \geq b/a$, then $x = \frac{b}{a} + \frac{1}{a}\sqrt{n + ay}$

so that the number of W – values in $(-e^T, e^T)$ is approximately

$$\frac{2}{a}\left(\sqrt{n+ae^T}-\sqrt{n-ae^T}\right)=\frac{4e^T}{\sqrt{n+ae^T}+\sqrt{n-ae^T}}.$$

If $T \approx \log(\frac{1}{3}M\sqrt{n/2}) - \log L$ and $a \approx \sqrt{2n}/M$, then the number of W–values in $(-e^T, e^T)$ is approximately

$$\frac{2\sqrt{2}M/(3L)}{\sqrt{1+1/(3L)}+\sqrt{1-1/(3L)}} \approx \frac{\sqrt{2}}{3L}M$$

and this is only a very small fraction of M.

The set

$$\Gamma_1 = \{(x,y) \in \mathbf{R}^2 \mid x > b/a,\ y = W(x),\ e^T \le y \le S\}$$

is called the right upper branch of Γ on the sieve interval. The right lower branch of Γ on the sieve interval is the set

$$\Gamma_2 = \{(x,y) \in \mathbf{R}^2 \mid x > b/a,\ y = W(x),\ -R \le y \le -e^T\}.$$

The left upper branch Γ_3 and left lower branch Γ_4 of Γ on the sieve interval are defined similarly: replace $x > b/a$ by $x < b/a$ in the corresponding definitions of the right upper and lower branch.

Let $t_{1,i}$ and $t_{2,i}$ be the total number of complete and incomplete relations of Γ_i ($i = 1, 2, 3, 4$), respectively. We have $t_1 \approx \sum_i t_{1,i}$ and $t_2 \approx \sum_i t_{2,i}$ since there are relatively few W–values in the interval $(-e^T, e^T)$.

3. Smooth integers in an interval

The Dickman–De Bruijn function ρ plays a key role in approximating the number of smooth integers below some bound. The function is defined by the differential–difference equation

$$\begin{aligned} \rho(u) &= 1 \quad (0 \le u \le 1), \\ u\rho'(u) + \rho(u-1) &= 0 \quad (u > 1). \end{aligned}$$

We have

$$\rho(u) = 1 - \log u \quad (1 \le u \le 2). \tag{3.1}$$

From the definition of ρ it follows that ρ is piecewise analytic and that ρ agrees with an analytic function ρ_m on the interval $[m-1, m]$ $(m = 1, 2, \dots)$. We can expand the Taylor series of ρ_m in a left neighborhood of $u = m$:

$$\rho(m-\xi) = \rho_m(m-\xi) = \sum_{i=0}^{\infty} c_i^{(m)} \xi^i \quad (0 \le \xi \le 1).$$

Bach and Peralta [BP92] describe an efficient method due to Patterson and Rumsey to compute the coefficients $c_i^{(m)}$ iteratively. Since $\rho_1(u) = 1$ $(0 \le u \le 1)$, we have $c_0^{(1)} = 1$ and $c_i^{(1)} = 0$ for $i > 0$. The coefficients $c_i^{(2)}$ can be computed from (3.1):

$$\rho(2-\xi) = 1 - \log 2 - \log(1-\xi/2) = 1 - \log 2 + \sum_{i=1}^{\infty} \frac{\xi^i}{i\, 2^i} \quad (0 \le \xi \le 1).$$

In general we have

$$c_i^{(m)} = \sum_{j=0}^{i-1} \frac{c_j^{m-1}}{i\, m^{i-j}} \quad \text{for } i > 0 \text{ and} \quad c_0^{(m)} = \frac{1}{m-1} \sum_{j=1}^{\infty} \frac{c_j^{(m)}}{j+1}.$$

Empirically, Bach and Peralta found that 55 coefficients are sufficient to compute ρ to IEEE standard double precision (with a relative error of about 10^{-17}) in the range $0 \le u \le 20$. Below we list rounded values of $\rho(m)$ for integers m in the range $2 \le m \le 11$. These values were computed from the Taylor series of ρ_m in a left neighborhood of $u = m$ using the first 55 coefficients. See also [vdLW69].

m	$\rho(m)$	m	$\rho(m)$
2	$0.306\ 853$	7	$0.874\ 567 \cdot 10^{-6}$
3	$0.486\ 084 \cdot 10^{-1}$	8	$0.323\ 207 \cdot 10^{-7}$
4	$0.491\ 093 \cdot 10^{-2}$	9	$0.101\ 625 \cdot 10^{-8}$
5	$0.354\ 725 \cdot 10^{-3}$	10	$0.277\ 017 \cdot 10^{-10}$
6	$0.196\ 497 \cdot 10^{-4}$	11	$0.664\ 481 \cdot 10^{-12}$

For the number $\psi(x, y)$ of positive y–smooth integers $\le x$ we use the approximation

$$\psi(x, y) \approx x \left(\rho(u) + (1-\gamma) \frac{\rho(u-1)}{\log x} \right), \tag{3.2}$$

that descends from the relation

$$\psi(x, x^{1/u}) = x \left(\rho(u) + (1-\gamma) \frac{\rho(u-1)}{\log x} \right) + \mathcal{O}(\Delta(x, x^{1/u})) \quad (x \to \infty). \tag{3.3}$$

Here

$$\Delta(x, x^{1/u}) = \begin{cases} \frac{x^{1/u}}{\log x} + \frac{x}{\log^2 x} & \text{for} \quad 1 < u \le 2, \\ \frac{x}{\log^{3/2} x} & \text{for} \quad u > 2. \end{cases}$$

For details on the function ψ we refer to the comprehensive bibliography in Norton's memoir [Nor71].

In the sequel we also need an approximation of the number of smooth integers in an interval. Hildebrand and Tenenbaum [HT86, p. 270] proved that, under the assumption of the Riemann hypothesis, for any fixed ϵ, $0 < \epsilon < 1$, we have

$$\psi\left(x + \frac{x}{z}, y\right) - \psi(x, y) = \frac{\log(1 + y/\log x)}{z \log y}\, \psi(x, y) \left(1 + \mathcal{O}_\epsilon\left(\frac{1}{z} + \frac{\log\log(1+y)}{\log y}\right)\right), \tag{3.4}$$

uniformly in the ranges

$$x \ge 2, \quad (\log\log x)^{2/3+\epsilon} < \log y \le (\log x)^{2/5}, \quad 1 \le z \le R(x, y)^{-1}. \tag{3.5}$$

Here $R(x, y) = \exp(-y^{1/2-\epsilon}) + \exp(-b_0 u \log^{-2} 2u) \log y$, where b_0 is some positive absolute constant.

We have $\log(1+y/\log x) \approx \log y - \log\log x$ and $\log\log(1+y) \approx \log\log y$. Substituting this and approximation (3.2) into (3.4) it follows that

$$\psi\left(x + \frac{x}{z}, y\right) - \psi(x, y) \approx \left(1 - \frac{\log\log x}{\log y}\right) \sigma(x, y, z) \left(1 + c_1(\epsilon)\frac{1}{z} + c_2(\epsilon)\frac{\log\log y}{\log y}\right), \tag{3.6}$$

where the function σ is defined by

$$\sigma(x, y, z) = \frac{x}{z}\left(\rho(u) + (1 - \gamma)\frac{\rho(u-1)}{\log x}\right),$$

and the $c_i(\epsilon)$ are numbers depending on ϵ.

To test approximation (3.6) (with appropriate values for the numbers $c_i = c_i(\epsilon)$) we sieved y–smooth integers from the interval $[x, x + \Delta]$ for various values of x, y and Δ. We chose x and y such that their values corresponded to the order of magnitude of polynomial values and values of B, respectively, that we used in our experiments described in Section 6. In the experiments described in this section $1/z = \Delta/x$ is negligible compared with $\log\log y/\log y$. Using the least squares method we get the

number $\tilde{c}_2 = 1.116$ that yields good approximations when using formula (3.6) with $c_2 = \tilde{c}_2$ (and $c_1 = 0$). In the table below we list the results. (The terms $2 \cdot 10^5$ and $2 \cdot 10^6$ were added to simplify the sieve program.)

x	y	Δ	(3.6)	sieved	quotient
10^{27}	$5 \cdot 10^4$	$10^8 + 2 \cdot 10^5$	3606	3521	1.024
10^{35}	$3 \cdot 10^5$	$10^8 + 2 \cdot 10^5$	527	529	0.996
10^{40}	$8 \cdot 10^5$	$10^8 + 2 \cdot 10^5$	159	149	1.067
10^{45}	$6.5 \cdot 10^5$	$10^{11} + 2 \cdot 10^6$	6771	6818	0.993
10^{50}	$8.5 \cdot 10^5$	$10^{11} + 2 \cdot 10^6$	646	666	0.970
10^{50}	10^6	$10^{11} + 2 \cdot 10^6$	912	928	0.983

We conclude that approximation (3.6) (and thus approximation (3.2)) is useful in practice in the range of our interest, even if we do not satisfy conditions (3.5) completely.

4. Complete relations

We show how we approximate the number $t_{1,1}$ of complete relations of the right upper branch Γ_1. We divide the interval $[e^T, S]$ in N subintervals $[y_i, y_{i+1}]$ ($i = 0, 1, \ldots, N-1$) in the following way. Let $h = (\log S - T)/N$, $f_i = T + ih$ and choose $y_i = e^{f_i}$ ($i = 0, 1, \ldots, N-1$). Hence $y_{i+1} = y_i + \frac{y_i}{z}$, where $z = (e^h - 1)^{-1}$.

To apply (3.6) with an appropriate choice of the numbers c_1 and c_2, we certainly must have $z \geq 1$ so that $e^h - 1 \leq 1$. This means that $N \geq (\log S - T)/\log 2$. Since $S \approx M\sqrt{n/2}$, $T \approx \log(\frac{1}{3}M\sqrt{n/2}) - \log L$, and in practice $L \geq 10B$ and $B \geq 10^5$, N must be larger than $(\log 3 + 6\log 10)/\log 2 \approx 21.5$. Our calculations indicate that $N = 100$ is a safe lower bound. We can choose N much larger so that $1/z$ becomes much smaller (see the end of Section 3), but then our algorithm to predict $t_{1,1}$ becomes too slow. Instead we stick to $N = 100$ and take the error constants in approximation (3.6) into account.

Let $x_i \in [-M, M]$ be the number (not necessarily an integer) such that $(x_i, y_i) \in \Gamma_1$. For the slope s_i of the chord between the points (x_i, y_i) and (x_{i+1}, y_{i+1}) we have $s_i = (y_{i+1} - y_i)/(x_{i+1} - x_i)$.

Let Y be a positive number and let $t_{1,1}^{(Y)}$ $\left(t_{2,1}^{(Y)}\right)$ denote the number of (in)complete relations $y = W(x)$ with $(x, y) \in \Gamma_1$ and $y \leq Y$. Clearly we have

$$t_{1,1} = \sum_{i=0}^{N-1} (t_{1,1}^{(y_{i+1})} - t_{1,1}^{(y_i)}). \tag{4.1}$$

Now we investigate the smoothness probability of polynomial values. Approximation (3.6) estimates the number of *random* smooth integers in an interval and thus we cannot

simply apply (3.6) to *special* smooth numbers such as complete relations. P.L. Montgomery [Mon95] proposed an elegant way to compare the smoothness probabilities of W–values and random numbers. The idea is as follows.

We compute the expected contribution of a prime $p \leq B$ to $W(x)$. Let $p \leq B$ be a prime not dividing the discriminant of W, i.e., p is not a divisor of $4n$. For those primes p we define r_p as the number of roots of the congruence equation $W(x) \equiv 0 \bmod p$. We have $r_p = 2$ or 0 according as n is a quadratic residue mod p or not. Any root modulo p corresponds to a unique root mod p^j for any $j > 1$ via Hensel lifting. Hence the expected factor contribution of p to $W(x)$ is

$$p^{r_p\left(\frac{1}{p}+\frac{1}{p^2}+\cdots\right)} = p^{r_p/(p-1)}.$$

We do not sieve with the prime divisors of n and so we put $r_p = 0$ if p divides n. Since $n \equiv 1 \bmod 8$ the expected contribution of prime 2 is

$$2^{2\left(\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\cdots\right)} = 2^2.$$

If we finally define $r_2 = 2$ then the estimated logarithmic norm after sieving by elements in the factor base is

$$\log W(x) - \sum_{p \leq B} r_p \frac{\log p}{p-1}.$$

Since the corresponding value for a random number y equals

$$\log y - \sum_{p \leq B} \frac{\log p}{p-1},$$

we assume that the numbers $W(x)$ are about as smooth as random integers with logarithmic norm $\alpha + \log W(x)$, where

$$\alpha = \sum_{p \leq B} (1 - r_p) \frac{\log p}{p-1}. \tag{4.2}$$

The probability that for a random integer y, with $y_i \leq y \leq y_{i+1}$, there exists an integer x such that (x, y) is a member of Γ_1 is approximately $1/s_i$. Hence, using the correction term α in (3.6), it follows that

$$t_{1,1}^{(y_{i+1})} - t_{1,1}^{(y_i)} \approx \left(1 - \frac{\log g_i}{\log B}\right) \frac{y_i}{s_i z} \left(\rho(v_i) + (1-\gamma)\frac{\rho(v_i - 1)}{g_i}\right) \left(1 + c_1 \frac{1}{z} + c_2 \frac{\log\log B}{\log B}\right), \tag{4.3}$$

where $g_i = \alpha + f_i$ and $v_i = g_i/\log B$. Note that $y_i/(s_i z) = x_{i+1} - x_i$. Combining this approximation with formula (4.1), we obtain an approximation of $t_{1,1}$.

To approximate $t_{1,2}$, we replace Γ_1 by Γ_2, S by R, y_i by $|y_i|$, $x_{i+1} - x_i$ by $x_i - x_{i+1}$ and proceed as above. Since $W(x)$ is almost symmetric around the y–axis by construction, we have $t_{1,3} \approx t_{1,1}$ and $t_{1,4} \approx t_{1,2}$.

5. Incomplete relations

We show how to approximate the number $t_{2,1}$ of incomplete relations from branch Γ_1. We have

$$t_{2,1} = \sum_{i=0}^{N-1} (t_{2,1}^{(y_{i+1})} - t_{2,1}^{(y_i)}). \tag{5.1}$$

Since the probability that a W–value is divisible by a prime q is r_q/q, we have

$$t_{2,1}^{(Y)} \approx \sum_{B<q\le L} \frac{r_q}{q}\, t_{1,1}^{(Y/q)},$$

where the summation ranges over all primes q between B and L. Write $g_{i,q} = \alpha + f_i - \log q$ and $v_{i,q} = g_{i,q}/\log B$. We then obtain

$$\begin{aligned} t_{2,1}^{(y_{i+1})} - t_{2,1}^{(y_i)} &\approx \sum_{B<q\le L} \frac{r_q}{q} \left(t_{1,1}^{(y_{i+1}/q)} - t_{1,1}^{(y_i/q)} \right) \\ \approx (x_{i+1} - x_i) \sum_{B<q\le L} \frac{r_q}{q} &\left(1 - \frac{\log g_{i,q}}{\log B}\right) \left(\rho(v_{i,q}) + (1-\gamma)\frac{\rho(v_{i,q}-1)}{g_{i,q}}\right) \\ &\times \left(1 + c_1 \frac{1}{z} + c_2 \frac{\log\log B}{\log B}\right), \end{aligned} \tag{5.2}$$

where we used (4.3) with f_i replaced by $f_i - \log q$ and $y_i/(s_i z)$ by $x_{i+1} - x_i$. Approximation (5.2) together with equation (5.1) yields an approximation of $t_{2,1}$. To compute approximations $t_{2,i}$ ($i = 2, 3, 4$) we apply similar adjustments as at the end of Section 4.

6. Numerical results

Below we list the results of our experiments. For each number to be factored we sieved thousands of polynomials and computed the average number of (in)complete relations obtained per polynomial. The number of polynomials we sieved is denoted by r, the estimated values of t_1 and t_2 are written as $\tilde{t}_1$ and $\tilde{t}_2$, respectively. By Cx we denote a composite number of x decimal digits and r.e. in the tables means relative error. We picked *one* polynomial to compute $\tilde{t}_1$ and $\tilde{t}_2$ as described above, since different polynomials in one experiment turned out to give almost the same values for $\tilde{t}_1$ and $\tilde{t}_2$.

We selected polynomials such that the leading coefficient a of each polynomial was the square of an integer co–prime with primes in the factor base.

We computed the Taylor series of ρ up to degree 55. Since we had to compute many values of the Dickman–De Bruijn function we precomputed a table of values $\rho(u)$ from $u = 2$ to $u = 10$ using a step size of $1/2^{11}$. Using linear interpolation we then approximated $\rho(u)$ for a particular value of u. The number N of subintervals of one branch was chosen te be 100.

To determine numbers c_1 and c_2 in approximations (4.3) and (5.2) we chose sample numbers to be factored (listed in the appendix), determined the actual number of complete and incomplete relations per polynomial and used the least squares method to minimize the sum of the squares of the relative errors. We found $c_1 = -0.4813$ and $c_2 = 1.344$ for approximation (4.3) and $c_1 = 1.688$ and $c_2 = -2.372$ for approximation (5.2).

The program for the approximation of complete relations was written in Maple V Release 3; for the incomplete relations we wrote a program in Fortran.

A. Complete relations

We did experiments for ten values of $(n/k, \sqrt{a}, b)$. (Note that coefficient c is determined by a and b since the discriminant of each polynomial is equal to $4n$.) These values can be found in the appendix. Tables 1 and 2 contain the chosen values of the parameters, the resulting value of α and the actually found value of t_1 compared to the calculated estimate $\tilde{t}_1$.

Number:	1a	2a	3a	4a	5a
k	1	41	1	47	71
$B/10^5$	2	4	3	4	3
$M/10^6$	1	0.5	1	1	0.5
T	76.55	80.32	82.42	84.74	87.46
r	3400	13 000	25 400	16 400	117 100
α	−2.605	−2.150	−0.5899	−2.373	−1.881
$\tilde{t}_1$	2.148	1.241	0.5849	0.8715	0.1099
t_1	2.604	1.206	0.5089	0.9684	0.1004
r.e. (%)	−17.5	2.89	14.9	−10.0	9.41

Table 1: Estimated and actual number of complete relations per polynomial

B. Incomplete relations

We did experiments for six tuples $(n/k, \sqrt{a}, b)$. Again, see the appendix for the actual values. Table 3 contains the parameter values and the results of the experiments.

Number:	6a	7a	8a	9a	10a
k	41	79	13	1	29
$B/10^5$	7	5	5	8	9.5
$M/10^6$	2	2.5	2.5	3	5
T	99.49	84.87	86.07	104.6	113.6
r	22 314	292 280	287 312	65 260	36 785
α	−2.652	−1.972	−1.293	−2.145	−1.592
$\tilde{t}_1$	0.1332	0.02 646	0.02 702	0.06 321	0.01 360
t_1	0.1423	0.03 058	0.03 009	0.06 281	0.01 279
r.e. (%)	−6.41	−13.5	−10.2	0.638	6.31

Table 2: Estimated and actual number of complete relations per polynomial

Number:	1b	2b	3b	4b	5b	6b
k	47	79	13	1	23	1
$B/10^5$	4	5	5	8	9.5	10
$L/10^6$	6	10	10	16	14.25	10
$M/10^6$	1	2.5	2.5	2	10	10
T	88.02	84.87	86.07	97.64	107.1	113.8
r	34 911	292 280	287 312	36 400	17 842	23 087
α	−1.865	−1.972	−1.293	−2.145	−2.072	−0.9389
$\tilde{t}_2$	0.3908	0.2366	0.2296	0.2796	0.1807	0.03 370
t_2	0.5052	0.2571	0.2569	0.3063	0.1876	0.03 180
r.e. (%)	−22.6	−7.97	−10.6	−8.71	−3.66	6.08

Table 3: Estimated and actual number of incomplete relations per polynomial

7. Complete relations from incomplete relations

What eventually matters is the total number of complete relations derived from the t_1 complete relations and the t_2 incomplete relations. In order to estimate this number, we have to compute the expected number, $E(s)$, of complete relations descendent from a given number s of incomplete relations. Lenstra and Manasse [LM94] explained how to approximate $E(s)$. In [BR95] it was shown that $E(s)$ can be estimated with, in almost all cases, a relative error of less than 5 %. For the sake of completeness we state the result here and we refer to [LM94] and [BR95] for a full explanation and numerical results. We have

$$E(s) \approx \frac{1}{2} \sum_{i=2}^{5} (-2)^i \binom{s}{i} \tau^{-i}(B, L, \beta)\ \tau(B, L, \beta i),$$

where τ is defined by

$$\tau(x,y,v) = \int_x^y t^{-v} d\frac{t}{\log t} = \frac{y^{1-v}}{\log y} - \frac{x^{1-v}}{\log x}$$
$$+ v(\mathrm{Ei}((1-v)\log y) - \mathrm{Ei}((1-v)\log x)) \quad (0 < x < y,\ v > 0),$$

Ei is the exponential integral defined by

$$\mathrm{Ei}(x) = \int_{-\infty}^{x} \frac{e^t}{t}\, dt,$$

and β is a positive constant smaller than 1. On the basis of our experiments we took $\beta = 0.73$, cf. [LM94].

8. The amount of work

In practice the sieve phase dominates the run time of the algorithm; it takes more than 90 % of the total time. Therefore we only consider the amount of work done in the sieve part of the algorithm. The determination of good parameter values depends heavily on the computer used and the implementation of the algorithm.

The amount of work is approximately proportional to the number of sieve updates (additions of logarithms to the elements of the sieve array). Per root and per factor base element q we have to apply the sieve updates on the $2M+1$ cells of the sieve interval, using stride q. This means that the number of sieve updates per polynomial is approximately equal to

$$4M \sum_{q \in \mathcal{F}} \frac{1}{q}.$$

The total number of complete relations after processing r polynomials is approximately $rt_1 + E(rt_2)$. Since we have to generate at least $1 + |\mathcal{F}|$ complete relations, an approximation of the minimal number of polynomials needed is the solution r_0 of the equation

$$r_0 t_1 + E(r_0 t_2) = 1 + |\mathcal{F}|. \tag{8.1}$$

We determine r_0 by using binary search in some interval. The total amount of work is approximately

$$4M r_0 \sum_{q \in \mathcal{F}} \frac{1}{q} \tag{8.2}$$

sieve updates and the expression is dependent on B, L and M. By varying the parameters in some interval, one can compute the total amount of work and thus determine good parameters.

On a computer without hierarchical memory (cache or virtual memory), for example the Cray C90 supercomputer, the CPU–time is a linear function in (8.2). Of course the same holds for implementations on computers with virtual memory, as long as page–faults do not occur during the sieving process.

Most people do not have access to a supercomputer and use an implementation on workstations (or PCs) with cache. In this case the CPU–time is dominated by cache effects [WW95]. Simply stated, cache is a small and expensive amount of memory that is accessible very fast by the processor(s). Physical memory is larger and cheaper but it takes much more time to fetch data from it. Therefore it makes sense to write programs such that the data used most frequently stays in cache as long as possible. In our case it is a good idea to split up the sieve array in blocks such that each block fits in cache. The elements of the sieve array under consideration are then manipulated very fast. Of course, at a certain moment the cache has to be refreshed. In general different computers use different cache policies (refreshment strategies). We conclude that it is difficult to give one formula that gives the CPU–time in terms of the parameters.

We give an example that illustrates how to determine good parameters. For a given choice of the parameters we estimate the number of complete and incomplete relations generated by one polynomial by formulae (4.3) and (5.2) and we estimate the total number of polynomials needed by solving equation (8.1). Next we do the actual sieve run for *one* polynomial and measure the CPU–time. Thus we have an estimation of the total sieve time. To verify our estimates we also carry out the complete sieve run for each choice of the parameters. In practice of course, this is only done for the final choice of parameters derived from our estimates. The sample number is the 62 decimal digit number

$$10\ 5783259093\ 2620060454\ 1346963019\ 3620363971\ 1810100364\ 0923795313$$

with $k = 1$, $M = 2.5 \cdot 10^5$, $T = 68.12$, $\alpha = -1.522$, $\sqrt{a} = 429\ 1273798937$ and
$b = 27443\ 6107325508\ 5474186145$.
We vary the most important parameter B in the range $[100\ 000, 500\ 000]$ with step size $50\ 000$ and take $L = 10 \cdot B$ for each choice of B. For simplicity the other parameters are kept fixed for each choice of B. The table below contains columns for the estimated and actual number r_0 of polynomials needed and the total sieve time in seconds. The experiments were carried out on a Silicon Graphics Indy workstation with one 100 MHz R4000 processor and 8 Kb data cache size. (Since we used an implementation written for the Cray vector computer, the code was not optimal for usage on workstations so a better performance should be possible.)

$B/10^5$	r_0 (est.)	r_0 (act.)	sieve time (est.)	sieve time (act.)
1	2377	2055	16 654	14 398
1.5	1641	1425	11 974	10 395
2	1317	1155	10 119	8874
2.5	1127	990	9140	8033
3	1003	885	8671	7655
3.5	918	825	8529	7674
4	855	765	8589	7689
4.5	804	720	8730	7825
5	764	690	9000	8135

The relative error in the estimation of the total sieve time is less than 16 %. We have plotted the estimated and actual sieve time, see Figure 8. The estimated times are systematically higher than the actual times, but since the *shape* of the two graphs is the same, the minimum of the estimated time graph lies close to that of the actual time graph. The graphs show that $B = 300\,000$ is a good choice, but some larger values are also acceptable since from there the sieve time does not fluctuate too much. If B becomes smaller than 200 000 then the sieve time increases considerably. Therefore, to be sure, one might choose a B that is somewhat larger than the estimated optimal choice.

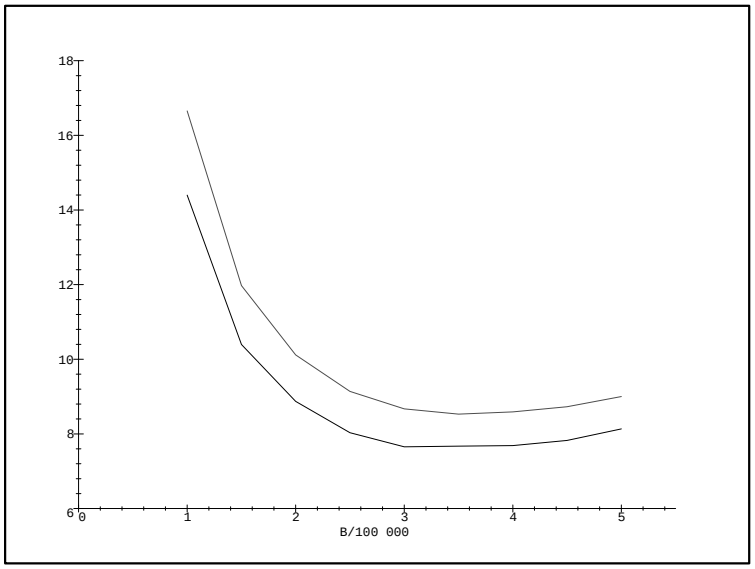

Figure 1: Estimated and actual sieve times (/1000 s)

9. Conclusions

We have given expressions to approximate the average number of complete and incomplete relations per polynomial we find using the single large prime variation of MPQS.

Next, the number of complete relations descendent from a given number of incomplete relations is estimated by using a known formula. From these results we can derive a good approximation of the total number of polynomials needed and hence we are able to estimate the total sieve time after processing one polynomial. The prediction formulae can be used to determine good parameters.

For numbers with 65–100 decimal digits our experiments indicate that the average relative error of the estimations of the number of (in)complete relations per polynomial is about 10 %.

An example illustrates that we can estimate the total sieve time with a relative error of less than 16 % only on the basis of a test run on *one* polynomial. The example, used to find a good value for B, yields a range of B-values where the actual sieve time is close to minimal.

Along the way we tested approximation formula (3.6) for the *total* number of smooth integers in an interval and observed that the approximation worked well in the range of our interest, even if conditions (3.5) were not satisfied completely. In the experiments with more than 500 sieved numbers the error of the estimate was less than 5 %. From this it follows that the classical formula (3.3) for the approximation of the total number of smooth numbers below some bound is also useful in practice.

Acknowledgements

I thank A.K. Lenstra, W.M. Lioen, P.L. Montgomery, H.J.J. te Riele and R. Tijdeman for their helpful remarks.

Appendix

The following 18 numbers Number 1–18 are used to determine c_1 and c_2 in approximations (4.3) and (5.2). The numbers do not interfere with those of Tables 1, 2 and 3. We list the square root $\sqrt{a}$ of the leading coefficient of the chosen polynomial and b. In the Tables 4, 5, 6 below we list the chosen parameters, the number r of sieved polynomials, α, t_1 and t_2.

Number 1 C62
$n/k = 11\ 9418190562\ 3383706600\ 5776102904\ 8038688463\ 4734397795\ 0205279043\backslash$
$\sqrt{a} = 1137\ 7985757553$, $b = -122533\ 9205727893\ 5806928333$

Number 2 C62
$n/k = 12\ 4960434331\ 2369771507\ 1881486773\ 5798764534\ 8378146406\ 8316835097\backslash$
$\sqrt{a} = 317\ 7594878677$, $b = -36769\ 2049127053\ 7109069372$

Number 3 C67
$n/k = 2795300\,4544218809\,0098428471\,3449908979\,6930327677\,1193962889\,0773152249\backslash$
$\sqrt{a} = 69049\ 8676367814$, $b = 6141119\ 7980268423\ 3761996924$

Number 4 C70
$n/k = 3452074587\ 1532036897\ 1663724885\ 9085645491\ 3776024644\ 1573043843\backslash$
3590687149
$\sqrt{a} = 61164\ 1107106217$, $b = -684001154\ 8751383787\ 6228906623$

Number 5 C71
$n/k = 1\ 1111111111\ 1111111111\ 1111111111\ 1111111111\ 1111111111\backslash$
$1111111111\ 1111111111$
$\sqrt{a} = 59982\ 2833488191$, $b = 1265942212\ 7495946598\ 6178939644$

Number 6 C73
$n/k = 125\ 8308688150\ 5179142348\ 1385763915\ 7371927998\ 2504149545\backslash$
$2167806649\ 6312946019$
$\sqrt{a} = 495384\ 4251317617$, $b = 9\ 2578266926\ 6479862115\ 5975230049$

Number 7 C73
$n/k = 374\ 6281167477\ 1257926128\ 5180095995\ 9020537316\ 7454521437\backslash$
$5261287497\ 4743890821$
$\sqrt{a} = 351266\ 6308597937$, $b = 1\ 7699066017\ 4448675718\ 5749361392$

Number 8 C76
$n/k = 901440\ 4467263718\ 8992383413\ 6656698645\ 6858019702\ 1805964137\backslash$
$0498292646\ 0417171821$
$\sqrt{a} = 5301075\ 2767899037$, $b = -963\ 8357898726\ 9109565464\ 7268570179$

Number 9 C81
$n/k = 1\ 2084979326\ 5793320196\ 7285818566\ 0539963519\ 8873106157\backslash$
$9813709116\ 9898443228\ 8871633489$
$\sqrt{a} = 9071728\ 9289477089$, $b = -13159\ 6978618865\ 5546891188\ 6370038398$

Number 10 C81
$n/k = 1\ 2756160780\ 8611099305\ 8711238270\ 6682428104\ 3584769443\backslash$
$1177445428\ 2344210861\ 0440705197$
$\sqrt{a} = 13501794\ 5809631273$, $b = -55856\ 6477642561\ 2204513394\ 1344005728$

Number 11 C81
$n/k = 1\ 2804626196\ 3485075791\ 3316233394\ 7732460978\ 8865639874\backslash$
$5232331043\ 0957142640\ 4937004817$
$\sqrt{a} = 9145180\ 1150415749$, $b = 8968\ 6153120320\ 6925962274\ 5273510003$

Number 12 C81

$n/k = 5\ 0637629921\ 9632522549\ 0667694055\ 0900653300\ 3228780546\backslash$
$9758645025\ 0364355475\ 3465666097$
$\sqrt{a} = 25252227\ 2969049649,\ b = 3421\ 9602490260\ 7990556819\ 7711369718$

Number 13 C82
$n/k = 54\ 1397729081\ 4966784940\ 0566002598\ 8468945940\ 6325764580\backslash$
$6017604383\ 9674763457\ 6574474851$
$\sqrt{a} = 58758139\ 0028964029,\ b = -27533\ 1111357320\ 1577238353\ 6387564611$

Number 14 C86
$n/k = 389079\ 4216696731\ 3164779897\ 1098709133\ 3423459224\ 7603987842\backslash$
$9842778661\ 3756224933\ 3353028281$
$\sqrt{a} = 297038425\ 8314072489,\ b = -2352011\ 2511694002\ 4248915047\ 5760387015$

Number 15 C88
$n/k = 11175709\ 2730877850\ 7566337550\ 9268349948\ 6353284691\ 8694497998\backslash$
$8537535212\ 0764988950\ 2631975827$
$\sqrt{a} = 1760793650\ 3778546661,\ b = 142666646\ 7066789181\ 4484201414\ 6991688541$

Number 16 C88
$n/k = 21635556\ 5990646589\ 4381332995\ 5809413444\ 1212020197\ 1957079131\backslash$
$3709426842\ 5925590441\ 2221060057$
$\sqrt{a} = 1646945457\ 7510843897,\ b = 55968176\ 2595691602\ 3311711198\ 9664685591$

Number 17 C91
$n/k = 1\ 3049963709\ 6764364819\ 9568631191\ 9613777969\ 9609425129\backslash$
$6224051097\ 7317881302\ 8992552947\ 8108284379$
$\sqrt{a} = 8391974343\ 4446932021,\ b = 1799938559\ 2054466830\ 2724667288\ 2065532161$

Number 18 C95
$n/k = 35497\ 2695591791\ 3798837415\ 1428489573\ 4961682843\ 8902745455\backslash$
$7880927844\ 3957786766\ 7167167895\ 2104479787$
$\sqrt{a} = 9\,4010239651\,0674712937,\ b = 11\,7351838585\,6995091351\,3581526051\,1685001862$

Number 1a up to Number 10a are examples used to compare the estimated and actual number of complete relations per polynomial (see Tables 1 and 2).

Number 1a C65
$n/k = 27158\ 0560707763\ 8170285090\ 4822402606\ 4919324961\ 0446180354\ 8064706241$
$\sqrt{a} = 1538\ 3080127953,\ b = -672943\ 6594969807\ 7121164793$

Number:	1	2	3	4	5	6
k	43	1	1	5	23	59
$B/10^5$	3	3.5	4	4.5	5	4.5
$L/10^6$	3	3.5	6	6.75	10	6.75
$M/10^6$	0.25	0.5	0.5	0.5	2	0.5
T	58.27	56.95	62.57	61.82	67.33	66.00
r	2475	1200	4000	9465	4964	25 000
α	−2.569	−1.397	−1.629	−0.6330	−1.797	−2.035
t_1	2.947	7.389	1.963	0.5008	2.057	0.2990
t_2	12.37	29.90	10.67	3.475	13.60	1.865

Table 4: Actual number of (in)complete relations per polynomial

Number:	7	8	9	10	11	12
k	5	109	1	5	1	1
$B/10^5$	5	5	6	5	4	7
$L/10^6$	5	5	180	50	200	7
$M/10^6$	0.5	0.5	2	2	2	0.5
T	65.61	71.04	81.49	76.96	79.67	73.83
r	24 312	25 385	69 432	79 328	87 904	43 970
α	−0.6977	−3.186	−0.9103	−1.318	−0.9949	−0.4083
t_1	0.2040	0.1916	0.1717	0.09 899	0.08 221	0.05 388
t_2	1.251	0.9779	2.305	1.676	1.651	0.2994

Table 5: Actual number of (in)complete relations per polynomial

Number 2a C67
$n/k = 4999228\ 1439980547\ 1200119698\ 6062426712\ 9417318768\ 0936258414\ 0274996129$
$\sqrt{a} = 20276\ 9767993829$, $b = -5103108\ 6608932988\ 4245660016$

Number 3a C70
$n/k = 3407462558\ 0870149333\ 5159834239\ 1305240184\ 2217017985\ 5972598052\backslash$
9382492473
$\sqrt{a} = 29115\ 0507704933$, $b = -200353646\ 9616791195\ 2118089548$

Number 4a C70
$n/k = 7446263828\ 5084664090\ 2304588883\ 2293771397\ 0280869969\ 0700499067\backslash$
2643660783
$\sqrt{a} = 91942\ 6170405581$, $b = -362914086\ 3289343194\ 0606822182$

Number:	13	14	15	16	17	18
k	11	1	43	17	19	11
$B/10^5$	7.5	8	8.5	8.5	9	8
$L/10^6$	11.25	8	8.5	8.5	9	12
$M/10^6$	1	1	1	1	1	1
T	76.43	80.01	83.51	83.38	86.58	106.1
r	13 500	16 068	15 000	16 000	27 500	113 000
α	−1.000	−1.251	−2.116	−1.862	−2.745	−0.8066
t_1	0.06 178	0.04 363	0.02 687	0.02 631	0.01 778	0.001 699
t_2	0.4461	0.2610	0.1559	0.1500	0.1044	0.0105

Table 6: Actual number of (in)complete relations per polynomial

Number 5a C73
$n/k = 452\ 2421093387\ 6231088785\ 6962118423\ 9709711935\ 6087407616\ 9382896713\backslash$
5633689471
$\sqrt{a} = 718570\ 3169703073$, $b = 4\ 9938242915\ 8672800079\ 0845023439$

Number 6a C83
$n/k = 138\ 1842483120\ 1804965895\ 4919777751\ 4146329101\ 5347888233\ 4496837656\backslash$
$5549227846\ 1653455089$
$\sqrt{a} = 72975948\ 9819948209$, $b = 194279\ 0662870352\ 6047288876\ 4531567349$

Number 7a C85
$n/k = 44884\ 3150792924\ 5691032960\ 4491477299\ 0585267717\ 7194205247\ 3840148537\backslash$
$2743386385\ 7714147711$
$\sqrt{a} = 339613268\ 9863474013$, $b = -22113030\ 0646978737\ 5654873872\ 8057541679$

Number 8a C85
$n/k = 63707\ 3407732464\ 4027659439\ 0221020431\ 2154608342\ 1294091436\ 7818192696\backslash$
$6398299023\ 2113191301$
$\sqrt{a} = 233339860\ 4569422437$, $b = -8866157\ 1010598397\ 3774112477\ 9900638768$

Number 9a C88
$n/k = 74819830\ 5013400188\ 9590279365\ 2111113412\ 8394311143\ 9946201421\backslash$
$9624982641\ 5188333977\ 2212945401$
$\sqrt{a} = 638716555\ 3933850141$, $b = 8422796\ 5151088504\ 2033019053\ 5854508860$

Number 10a C94

$n/k = 5820\ 8670385704\ 7731277020\ 3646825636\ 8097924769\ 4045734980\ 3172591869\backslash$
$2043944029\ 4338517568\ 2700432109$
$\sqrt{a} = 3\ 4090217783\ 4280797121$, $b = 4\ 0593839263\ 3466719333\ 7596276279\ 3179399737$

Number 1b up to Number 6b are examples used to compare the estimated and actual number of incomplete relations per polynomial (see Table 3).

Number 1b C78
$n/k = 39431014\ 3497913309\ 9512682779\ 8377912751\ 2464722783\ 4573671137\backslash$
$3104823631\ 3132467031$
$\sqrt{a} = 7812839\ 5401083641$, $b = 1623\ 8764058964\ 5821009161\ 0356527005$

Number 2b C85
$n/k = 44884\ 3150792924\ 5691032960\ 4491477299\ 0585267717\ 7194205247\ 3840148537\backslash$
$2743386385\ 7714147711$
$\sqrt{a} = 339613268\ 9863474013$, $b = -22113030\ 0646978737\ 5654873872\ 8057541679$

Number 3b C85
$n/k = 63707\ 3407732464\ 4027659439\ 0221020431\ 2154608342\ 1294091436\ 7818192696\backslash$
$6398299023\ 2113191301$
$\sqrt{a} = 233339860\ 4569422437$, $b = -8866157\ 1010598397\ 3774112477\ 9900638768$

Number 4b C88
$n/k = 74819830\ 5013400188\ 9590279365\ 2111113412\ 8394311143\ 9946201421\backslash$
$9624982641\ 5188333977\ 2212945401$
$\sqrt{a} = 782193202\ 5824032009$, $b = 9613471\ 1095933128\ 1036342979\ 5711992830$

Number 5b C94
$n/k = 1592\ 0897086948\ 0220278154\ 1767196127\ 6633096702\ 4136090439\ 3052330290\backslash$
$7248179613\ 9299053383\ 8706754807$
$\sqrt{a} = 1\ 6450910750\ 5595962661$, $b = 7895009666\ 7907417812\ 2472377015\ 3992652248$

Number 6b C100
$n/k = 1193079720\ 2615798693\ 9665343380\ 4125664465\ 3472413608\ 6808267215\backslash$
$2071152844\ 3608270987\ 3992085756\ 0778854537$
$\sqrt{a} = 22\ 1018067486\ 2769424693$
$b = -70\ 1459496438\ 2136923254\ 1327637222\ 2556389907$

References

[BP92] E. Bach and R. Peralta. Asymptotic semi–smoothness probabilities. Technical Report 1115, University of Wisconsin, Computer Sciences Department,

Madison, October 1992.

[BR95] H. Boender and H. J. J. te Riele. Factoring integers with large prime variations of the quadratic sieve. *Experimental Mathematics*, 1995. To appear.

[Bre89] D. M. Bressoud. *Factorization and Primality Testing*. Springer–Verlag, New York, NY, 1989. Undergraduate Texts in Mathematics.

[HT86] A. Hildebrand and G. Tenenbaum. On integers free of large prime factors. *Transactions of the American Mathematical Society*, 296(1):265–290, July 1986.

[LM94] A. K. Lenstra and M. S. Manasse. Factoring with two large primes. *Mathematics of Computation*, 63:785–798, 1994.

[Mon95] P. L. Montgomery, December 1995. Private communication.

[Nor71] K. K. Norton. *Numbers with small prime factors, and the least kth power non–residue*. AMS, Department of Mathematics, University of Michigan, Ann Arbor, Michigan 48104, 1971.

[Pom85] C. Pomerance. The quadratic sieve factoring algorithm. In T. Beth, N. Cot, and I. Ingemarsson, editors, *Advances in Cryptology, Proceedings of EUROCRYPT 84*, volume 209 of *Lecture Notes in Computer Science*, pages 169–182, Springer–Verlag, New York, 1985.

[PST88] C. Pomerance, J. W. Smith, and R. Tuler. A pipeline architecture for factoring large integers with the quadratic sieve algorithm. *SIAM Journal on Computing*, 17:387–403, 1988.

[RLW89] H. J. J. te Riele, W. M. Lioen, and D. T. Winter. Factoring with the quadratic sieve on large vector computers. *Journal of Computational and Applied Mathematics*, 27:267–278, 1989.

[Sil87] R. D. Silverman. The multiple polynomial quadratic sieve. *Mathematics of Computation*, 48:329–339, 1987.

[vdLW69] J. van de Lune and E. Wattel. On the numerical solution of a differential–difference equation arising in analytic number theory. *Mathematics of Computation*, 23:417–421, 1969.

[WW95] G. Wambach and H. Wettig. Block sieving algorithms. University of Cologne. Manuscript, May 1995.