DETERMINISTIC FACTORIZATION OF SUMS AND DIFFERENCES OF POWERS

MARKUS HITTMEIR

ABSTRACT. Choose \(a, b \in \mathbb{N}\) and let \(N\) be a number of the form \(a^n \pm b^n\), \(n \in \mathbb{N}\). We will generalize a result of Bostan, Gaudry and Schost (2007) and prove that we may compute deterministically the prime factorization of \(N\) in
\[
\mathcal{O}\left( M_{\text{int}} \left( \frac{N^{1/4}}{\sqrt{\log N}} \right) \right),
\]
where \(M_{\text{int}}(k)\) denotes the cost for multiplying two \([k]\)-bit integers. This result is better than the currently best known general bound for the runtime complexity for deterministic integer factorization.

1. Introduction

In this paper, we consider runtime complexity bounds for integer factorization, where complexity always means bit-complexity in the model of the multitape Turing machine [P94].

The currently best known deterministic and unconditional runtime complexity bound for computing the prime factorization of any natural number \(N\) is due to E. Costa and D. Harvey [CH14] and given by
\[
\mathcal{O}\left( M_{\text{int}} \left( \frac{N^{1/4} \log N}{\sqrt{\log \log N}} \right) \right),
\]
where \(M_{\text{int}}(k)\) denotes the cost for multiplying two \([k]\)-bit integers. M. Fürer showed in [F09] that \(M_{\text{int}}(k)\) can be bounded by \(\mathcal{O}(k \log k^{2^{O(\log^* k)}})\), where
\[
\log^* k := \begin{cases} 
0 & \text{if } k \leq 1, \\
1 + \log^* (\log k) & \text{if } k > 1,
\end{cases}
\]
is the iterated logarithm. The proof in [CH14] improves the well-known approach of V. Strassen, which has been presented in [S77]. Both methods use fast polynomial evaluation for computing subproducts of the product \(\lfloor N^{1/2} \rfloor!\) to find a nontrivial factor of \(N\).

Our contribution is a further improvement of Strassen’s approach for numbers of certain shape, namely for sums and differences of powers. The main theorem of this paper is the following.
Theorem 1.1. Choose $a, b \in \mathbb{N}$ and let $N$ be a number of the form $a^n \pm b^n$, $n \in \mathbb{N}$. Then, we may compute deterministically the prime factorization of $N$ in

$$O\left(M_{\text{int}}\left(N^{1/4} \sqrt{\log N}\right)\right)$$

bit operations. The implied constant depends on $a$ and $b$.

The core idea is given in Lemma 2.5 and concerns the case that the prime divisors $p$ of $N$ satisfy $r = p \mod m$ for known $m, r \in \mathbb{N}_0$. We show that we may use this information and employ the well-known result of Lemma 2.4 on fast polynomial evaluation in order to speed up Strassen’s approach, essentially by getting rid of all the factors of $\lfloor N^{1/2}\rfloor$ which are not of this form. Theorem 2.8 is a generalization of a result due to A. Bostan, P. Gaudry and É. Schost [BGS07] and demonstrates how the size of $m$ affects the runtime complexity bound. In Algorithm 3.1, we finally combine this result with the observation that for $N = a^m \pm b^m$,

$$(ab^{-1})^{2m} \equiv 1 \mod p$$

holds for every prime divisor $p$ of $N$. Using Lagrange’s theorem on the order of $ab^{-1}$, we may extract the factors of $N$ step by step, until we are left with only those prime divisors $p$ satisfying $1 = p \mod m$.

We would like to point out that the theorem applies to some interesting subsets of \mathbb{N}, like Mersenne numbers or Fermat numbers. However, our deterministic method is of theoretical interest only and not appropriate for factorizing large numbers. In practice, probabilistic algorithms with much lower complexity are used for this task. For further information, we refer the reader to the survey [CP05].

2. Preliminaries

We briefly introduce the notions and results we will use in Section 3 to prove Theorem 1.1. The following definitions describe the invertibility conditions required for fast polynomial evaluation provided by Lemma 2.4. They have been first introduced in [BGS07]. Let $N \in \mathbb{N}$ and $\mathbb{Z}_N := \mathbb{Z}/N\mathbb{Z}$.

Definition 2.1. Let $\alpha, \beta \in \mathbb{Z}_N$ and $d \in \mathbb{N}$. We say that $h(\alpha, \beta, d)$ is satisfied if

$$\beta, 2, \ldots, d, (\alpha - d\beta), (\alpha - (d - 1)\beta), \ldots, (\alpha + d\beta)$$

are invertible in $\mathbb{Z}_N$, and we define

$$d(\alpha, \beta, d) = \beta \cdot d(\alpha - d\beta)(\alpha - (d - 1)\beta) \cdots (\alpha + d\beta).$$

$h(\alpha, \beta, d)$ holds if and only if $d(\alpha, \beta, d)$ is invertible.

Definition 2.2. Let $\beta \in \mathbb{Z}_N$ and $e \in \mathbb{N}$. We say that $H(2^e, \beta)$ is satisfied if $h(2^i, \beta, 2^i)$ and $h((2^i + 1)\beta, \beta, 2^i)$ hold for each $0 \leq i < e$. We define

$$D(2^e, \beta) = \prod_{i=0}^{e-1} d(2^i, \beta, 2^i)d((2^i + 1)\beta, \beta, 2^i).$$

$H(2^e, \beta)$ holds if and only if $D(2^e, \beta)$ is invertible.

Lemma 2.3. Let $f_0, \ldots, f_{k-1} \in \mathbb{Z}_N$. Then we can decide if all $f_i$ are invertible in $\mathbb{Z}_N$ and, if not, find a noninvertible $f_i$ in

$$O(kM_{\text{int}}(\log N) + \log kM_{\text{int}}(\log N) \log \log N)$$

bit operations.
Proof. See Lemma 12 in [BGS07] for a proof. The idea is to form the subproduct tree associated with the values \( f_i \) and compute the gcd of its elements and \( N \). At most \( k \) multiplications are necessary to build up the tree, which has depth \( \log k \). Hence, if there is a noninvertible element, it can be found by computing \( \log k \) gcds.

Now let \( H \in \mathbb{Z}_N[X] \) with \( \deg H = 1 \). We define

\[
H_k(X) = H(X)H(X+1)\cdots H(X+k-1)
\]

and consider the problem of evaluating \( H_k \) on an arithmetic progression.

**Lemma 2.4.** Let \( \beta \in \mathbb{Z}_N \), \( e \in \mathbb{N} \) and \( k = 2^e \). Assume that \( H(k, \beta) \) holds and that the inverse of \( \mathcal{D}(k, \beta) \) is known. We may compute

\[
H_k(0), H_k(\beta), H_k(2\beta), \ldots, H_k(k\beta)
\]

in \( O(M_{\text{int}}(k \log(kN))) + M_{\text{int}}(\log N)) \) bit operations.

Proof. We apply Proposition 7 in [CH14] with \( \rho = 1 \). Starting with \( H(0) \) and \( H(\beta) \), the proof is based on the observation that we may use the values \( H_j(0), \ldots, H_j(j\beta) \) to efficiently compute \( H_{2j}(0), \ldots, H_{2j}(2j\beta), j \in \mathbb{N} \). The invertibility conditions are necessary because the core of this computation uses Lagrange’s interpolation formula.

The following lemma contains the key ingredient for the proof of Theorem 1.1 and is based on the idea to use information about a prime divisor \( p \) of \( N \) to compute two sets which are disjoint modulo \( N \), but not modulo \( p \). In Corollary 2.6 we show that the result of Lemma 2.5 can be translated to fit the setting of Lemma 2.4.

**Lemma 2.5.** Let \( N \in \mathbb{N} \) be composite and \( p \) a prime factor of \( N \) with \( p \leq b \) for some \( b \leq N/5 \). If \( r, m \in \mathbb{N}_0 \) such that \( \gcd(N, m) = 1 \), \( m < p \) and \( r = p \mod m \), then the sets

\[
\{m^{-1}r - i \mod N : 0 \leq i \leq k-1\},
\{-j k \mod N : 1 \leq j \leq k\},
\]

with \( k = \lceil (b/m)^{1/2} \rceil \) are disjoint and there exist \( u \in \{1, \ldots, k\} \) and \( v \in \{0, \ldots, k-1\} \) such that \( m^{-1}r - v \equiv -uk \mod p \).

Proof. From \( m \leq p \leq b \leq mk^2 \) it follows that there exists \( x \in \{1, 2, \ldots, k^2\} \) with \( p = mx + r \). We write \( x = ku - v \) for some \( u \in \{1, 2, \ldots, k\} \) and \( v \in \{0, 1, \ldots, k-1\} \). Note that \( p = m(ku - v) + r \) implies \( mku + r \equiv mv \mod p \). Define

\[
b_j := mkj + r, \text{ where } j \in \{1, \ldots, k\},
\]

\[
s_i := mi, \text{ where } i \in \{0, \ldots, k-1\}.
\]

We derive \( 0 \leq s_i \leq m(k-1) < mk + r \leq b_j \leq mk^2 + r < N \) for every \( i, j \), since

\[
mk^2 + r = m((b/m)^{1/2})^2 + r < m((b/m)^{1/2} + 1)^2 + m
\]

\[
= b + 2(bm)^{1/2} + 2m < 5b \leq N.
\]

As a consequence, the sets \( \{b_j : 1 \leq j \leq k\} \) and \( \{s_i : 1 \leq i \leq k-1\} \) are disjoint and we have also seen that there exist \( u, v \) such that \( b_u \equiv s_v \mod p \). It is easy to see that this implies the statement.
Corollary 2.6. With the notation of Lemma 2.5 let furthermore \( H \) be defined as \( X - m^{-1}r \in \mathbb{Z}_N[X] \) and set \( k = \lceil (b/m)^{1/2} \rceil \). Then at least one of the elements
\[
H_k(-k), H_k(-2k), \ldots, H_k(-k^2)
\]
is noninvertible in \( \mathbb{Z}_N \). Let \( u \in \{1, \ldots, k\} \) such that \( H_k(-uk) \) is noninvertible. Then \( \gcd(-uk - m^{-1}r + v \mod N, N) \) yields a nontrivial factor of \( N \) for some \( v \in \{0, \ldots, k - 1\} \).

Proof. Lemma 2.5 yields that there exist \( u \in \{1, \ldots, k\} \) and \( v \in \{0, \ldots, k - 1\} \) such that \( m^{-1}r - v \equiv -uk \mod p \). This implies that \(-uk - m^{-1}r + v \) is noninvertible in \( \mathbb{Z}_N \). Clearly, the same holds for \( H_k(-uk) = \prod_{i=0}^{k-1}(-uk - m^{-1}r + i) \). Since the sets in Lemma 2.5 are disjoint, we deduce
\[
p \mid \gcd(-uk - m^{-1}r + v \mod N, N) \neq N,
\]
which proves the claim. \( \square \)

The following two results are the technical core of the proof of Theorem 1.1. Remark 2.9 explains how they may be considered as generalizations of similar statements in \[ \text{[BGS07]} \].

Lemma 2.7. Let \( N \) be a natural number and \( r, m \in \mathbb{N}_0 \) such that \( r = p \mod m \) for every prime divisor \( p \) of \( N \). Let \( e \in \mathbb{N} \) such that \( b := 4^e m \leq N/5 \). Knowing \( r \) and \( m \), one can compute a prime divisor \( p \) of \( N \) with \( p \leq b \) or prove that no such divisor exists in
\[
O(M_{\text{int}}(2^e \log(N)) + eM_{\text{int}}(\log N) \log \log N)
\]
bit operations.

Proof. If \( m \geq q \) for the smallest prime factor \( q \) of \( N \), then \( m = q \) or \( r = q \) and we have already found a prime divisor. Since it is easy to determine whether this is the case or not, we may assume \( m < q \). Set \( k := 2^e = \sqrt{b/m} \) and define \( H = X - m^{-1}r \in \mathbb{Z}_N[X] \). We want to apply Lemma 2.4 with \( \beta = -k \) to compute the values
\[
H_k(-k), H_k(-2k), \ldots, H_k(-k^2).
\]
In order to do this, we have to check if \( H(k, -k) \) holds. It is easy to see that this is the case if and only if \( 2, 3, \ldots, 2^e + 1 \) and
\[
(2^i - 2^i2^e), (2^i - (2^i - 1)2^e), \ldots, (2^i + 2^i2^e)
\]
are invertible in \( \mathbb{Z}_N \) for each \( 0 \leq i < e \). This list consists of \( O(k) \) easily computable elements in \( \mathbb{Z}_N \). By assuming that there are \( 0 \leq i < e \) and \( j \in \mathbb{Z} \) such that \( 2^i - (2^i - j)2^e = 0 \), we deduce the contradiction that \( 2^e-i\cdot j = 2^e - 1 \). We conclude that the absolute values of the elements in the list are lower bounded by \( 1 \) and upper bounded by
\[
2^e-1 + 2^{e-1}2^e < 4^e < b \leq N/5.
\]
Hence, they are all nonzero modulo \( N \). By Lemma 2.3 we are able to decide if all of them are invertible in \( \mathbb{Z}_N \) or, if not, find a noninvertible one in
\[
O(kM_{\text{int}}(\log N) + \log kM_{\text{int}}(\log N) \log \log N)
\]
bit operations. Assume that we have found a noninvertible element; then we have also deduced a nontrivial factor of \( N \) bounded by \( k^2 = 4^e \). By using trial division, we are able to compute a prime divisor of \( N \) in \( O(kM_{\text{int}}(\log N)) \) bit operations.
In this case, the result is proven. Now assume that all of the elements above are noninvertible. We are able to compute $\mathcal{D}(k, -k) \in \mathbb{Z}_N$ in $O(kM_{\text{int}}(\log N))$ bit operations. The cost for computing its inverse is negligible. We now apply Lemma 2.4 to compute $H_k(-jk)$ for $j = 1, \ldots, k$, and since $k < N$, we can do this in

$$O(M_{\text{int}}(k \log(N)) + M_{\text{int}}(\log N))$$

bit operations. Suppose that $N$ has a prime factor $p \leq b$ with $r = p \mod m$. Then by Corollary 2.6, there exists at least one $u \in \{1, \ldots, k\}$ such that $H_k(-uk)$ is noninvertible in $\mathbb{Z}_N$. We use Lemma 2.3 to find such an element. Let $H_k(-uk)$ be noninvertible in $\mathbb{Z}_N$. Then Corollary 2.6 also yields that $\gcd(-uk - m^{-1}r + v \mod N, N)$ is nontrivial for some $v \in \{0, \ldots, k - 1\}$. We may apply Lemma 2.3 again to determine such an element $-uk - m^{-1}r + v \mod N$. Note that all these computations can be done in

$$O(kM_{\text{int}}(\log N) + \log kM_{\text{int}}(\log N) \log \log N)$$

bit operations. Now we know that $\gcd(-uk - m^{-1}r + v \mod N, N)$ is divisible by a prime divisor $p$ of $N$. This implies $-uk - m^{-1}r + v \equiv 0 \mod p$, hence $0 \equiv muk + r - mv \mod p$. Therefore, $\gcd(muk + r - mv, N)$ is nontrivial. The value $muk + r - mv$ is bounded by $mk^2 + r < mk^2 + m < 2b$. Again, we use trial division to find a prime divisor of $N$. There are less than $\lceil \sqrt{2k} \rceil$ primes $p$ smaller than $\lceil \sqrt{2b} \rceil$ satisfying $r = p \mod m$, since they have to be of the form $mx + r$ for some $x \in \{0, \ldots, \lceil \sqrt{2k}/\sqrt{m} \rceil - 1\}$. Therefore, the trial division can be done by $O(kM_{\text{int}}(\log N))$ bit operations. This proves the claim.

**Theorem 2.8.** Let $N$ be a natural number and $r, m \in \mathbb{N}_0$ such that $r = p \mod m$ for every prime divisor $p$ of $N$. Knowing $r$ and $m$, one can compute the prime factorization of $N$ in

$$O\left(M_{\text{int}}\left(\frac{N^{1/4}}{\sqrt{m}} \log N\right)\right)$$

bit operations.

**Proof.** We apply Lemma 2.7 with $b = 4^e m$ for $e \in \mathbb{N}$, starting with $e = 1$. Lemma 2.7 is applied with the same value of $b$ until no prime divisor of $N$ smaller than $b$ is found. Then we increase $e$ by 1 and repeat. We do this until $b \geq \sqrt{N}$. If $N \geq 400$, $b$ is always bounded by $4\sqrt{N} \leq N/5$. If $N$ is smaller, we may find its prime factorization in $O(1)$ by using trial division.

If we run Lemma 2.7 with value $b$, all prime divisors smaller than $b/4$ have already been detected. Since their product is bounded by $N$, we derive that the number of prime divisors between $b/4$ and $b$ and hence the number of runs of Lemma 2.7 with the same value $b$ is bounded by $O(\log N/\log b)$. The sum of all the terms of the form $eM_{\text{int}}(\log N) \log N$ in the runtime complexity of Lemma 2.7 is bounded by a polynomial in $\log N$ and hence negligible. We consider the sum of the other terms. Since $4^e m \geq \sqrt{N}$ implies $e \geq (\log N)/4 - (\log m)/2$, we define $e_0 := \lceil (\log N)/4 - (\log m)/2 \rceil$ and get

$$\sum_{i=1}^{e_0} \frac{\log N}{\log (4^i m)} M_{\text{int}}(2^i \log(N)) \leq M_{\text{int}}\left(\log N \sum_{i=1}^{e_0} \left\lceil \frac{\log N}{2i + \log m} \right\rceil 2^i\right).$$

Note that the inequality is a simple consequence of the facts that $k \leq M_{\text{int}}(k)$ and $M_{\text{int}}(k) + M_{\text{int}}(k') \leq M_{\text{int}}(k + k')$. 
Now we split the sum on the right side into \( i \leq e_0/2 \) and \( i > e_0/2 \). For \( i \leq e_0/2 \) we have \( 2^i \leq 2^{e_0/2} \in \mathcal{O}(N^{1/8}/m^{1/4}) \), hence the first part of the sum is bounded by \( \mathcal{O}((\log N)^2(N^{1/8}/m^{1/4})) \) and therefore negligible. We consider the main contribution by the summands with \( i > e_0/2 \). In these cases we have \( 2i + \log m > (\log N)/4 - (\log m)/2 + \log m > (\log N)/4 \), hence the terms \( \lfloor \log N/(2i + \log m) \rfloor \) are in \( \mathcal{O}(1) \). We conclude that this part of the sum can be bounded by
\[
\mathcal{O}(2^{e_0}) = \mathcal{O}(2^{(\lfloor \log N/4 \rfloor - (\log m)/2)}) = \mathcal{O}(N^{1/4}/\sqrt{m}),
\]
which proves the claim. \( \square \)

Remark 2.9. If we apply Lemma 2.7 and Theorem 2.8 with \( m = 1 \) and \( r = 0 \), we get the results of Lemma 13 and Theorem 11 in [BGS07].

3. Algorithm and Proof

Let \( a, b \in \mathbb{N} \) be fixed. We first define the sets
\[
P_{a,b}^+ := \{a^n + b^n : n \in \mathbb{N}\}, \quad P_{a,b}^- := \{a^n - b^n : n \in \mathbb{N}\}, \quad P_{a,b} := P_{a,b}^+ \cup P_{a,b}^-.
\]
Since we are interested in computing the prime factorization of nonnegative numbers, we suppose that \( a \geq b \).

Furthermore, if \( g := \gcd(a, b) > 1 \), we compute the prime factorization of \( g \) in \( \mathcal{O}(1) \) and then continue by considering \( (a/g)^n \pm (b/g)^n \).

As a result, we may assume that \( a \) and \( b \) are coprime and \( a > b \). We use the following algorithm to factorize sums and differences of powers:

Algorithm 3.1. Let \( N \in P_{a,b} \). We can write either \( N = a^m + b^m \in P_{a,b}^+ \) or \( N = a^m - b^m \in P_{a,b}^- \) for some \( m \in \mathbb{N} \). Set \( N_1 := N \), \( v := 1 \) and take the following steps to compute the prime factorization of \( N \):

1. Apply trial division to compute all divisors of \( m \). If \( N \in P_{a,b}^+ \), define \( \mathcal{D} := \{2d : d \mid m\} \). If \( N \in P_{a,b}^- \), define \( \mathcal{D} := \{d : d \mid m\} \). For \( l \geq 2 \), let \( d_1 < d_2 < \cdots < d_l \) be the ordered list of all elements in \( \mathcal{D} \).

2. Set \( j = v \).

3. Compute \( G_j = \gcd((ab^{-1})d_j - 1 \mod N_j, N_j) \). If \( G_j = 1 \), set \( N_{j+1} = N_j \).

If \( 1 \leq G_j \leq N \), apply Theorem 2.8 for \( m = d_j \) and \( r = 1 \) to compute the prime factorization of \( G_j \), remove all prime factors dividing \( G_j \) from \( N_j \) and denote the resulting number by \( N_{j+1} \). If \( N_{j+1} = 1 \), stop. If not, set \( v = j + 1 \) and go to Step 2.

Proof of Theorem 1.1 We show that Algorithm 3.1 is correct. First note that \( N \in P_{a,b}^+ \) implies \( (ab^{-1})2m \equiv 1 \mod N \) and \( N \in P_{a,b}^- \) implies \( (ab^{-1})m \equiv 1 \mod N \). Hence, we have \( N_{i+1} = 1 \) in any case and the algorithm always terminates.

To prove correctness, it remains to show that the conditions in Theorem 2.8 are always satisfied. Let \( i \in \{1, \ldots, l\} \) be arbitrary and let \( p \) be any prime factor of \( G_i \). We have to prove that \( 1 = p \mod d_i \). If \( N \in P_{a,b}^- \), then \( (ab^{-1})^m \equiv 1 \mod p \). Hence, the order \( o \) of the element \( ab^{-1} \) modulo \( p \) is a divisor of \( m \). We derive \( o \in \mathcal{D} \). If \( N \in P_{a,b}^+ \), then \( (ab^{-1})^m \equiv -1 \mod p \) and \( (ab^{-1})2m \equiv 1 \mod p \). Hence, the order \( o \) of the element \( ab^{-1} \) modulo \( p \) is of the form \( 2d \) for some divisor \( d \) of \( m \). Again, we derive \( o \in \mathcal{D} \).

Since \( p \) divides \( G_i \), we deduce \( (ab^{-1})d_j \equiv 1 \mod p \). Furthermore, \( p \) divides \( N_i \) and therefore has not been removed as the prime factor in the previous runs of the loop. But this implies \( (ab^{-1})d_j \not\equiv 1 \mod p \) for \( 1 \leq j < i \), and we conclude that \( o = d_i \). Since the order of any element is a divisor of the group order \( p - 1 \), we derive \( p \equiv 1 \mod d_i \) and the claim follows.
We now consider the runtime of Algorithm 3.1.

**Step 1.** Note that $N \geq a^m - b^m \geq (b + 1)^m - b^m \geq b^{m-1}$. This implies $m \leq \log_b N + 1 = \log N/\log b + 1 \in \mathcal{O}(\log N)$ for $b \neq 1$. For $b = 1$, it is also easy to see that $m$ is bounded by $\mathcal{O}(\log N)$. Hence, the cost to compute all divisors of $m$ and to bring them into the right order is negligible.

**Loop in Steps 2 and 3.** The cardinality of $\mathcal{D}$ can be bounded by $\mathcal{O}(\log N)$. The cost for computing the greatest common divisors is negligible. Assume the computational worst case, in which we have to apply Theorem 2.8 for every cost for computing the greatest common divisors is negligible. Assume the computational worst case, in which we have to apply Theorem 2.8 for every $j \in \{1, \ldots, l\}$. We consider $1 \leq j < l$. Then we have $a^{d_j} \equiv b^{d_j} \mod G_j$, hence $G_j \mid a^{d_j} - b^{d_j}$. If $N \in P_{a,b}^-$, we can write $d_j = m/k$ for some $k \geq 2$, and we deduce that

$$G_j \leq a^{d_j} - b^{d_j} = a^{m/k} - b^{m/k} \leq (a^m - b^m)^{1/k} \leq (a^m - b^m)^{1/2} = N^{1/2}.$$  

As a consequence, the runtime of all the applications of Theorem 2.8 for $1 \leq j < l$ can be bounded by $\mathcal{O}(M_{\text{int}}(N^{1/8}(\log N)) \log N)$ and is negligible. If $N \in P_{a,b}^+$, then we have $d_j = 2m/k$ for some $k \geq 2$. Note that $G_j > 1$ implies $d_j \neq m$, since $a^m \equiv -b^m \not\equiv b^m \mod p$ for every prime factor $p$ of $N$. We deduce that $k > 2$ and therefore

$$G_j \leq a^{d_j} - b^{d_j} = a^{2m/k} - b^{2m/k} \leq (a^m - b^m)^{2/k} \leq (a^m - b^m)^{2} < (a^m + b^m)^{2} = N^{2/3}.$$

We hence conclude that in this case the runtime of all the applications of Theorem 2.8 for $1 \leq j < l$ can be bounded by $\mathcal{O}(M_{\text{int}}(N^{1/6}(\log N)) \log N)$ and is negligible.

We now consider the runtime of Theorem 2.8 for the case $j = l$. First note that $N \leq a^m + b^m < 2a^m$ implies $m > \log_a N - \log_a 2 = \log N/\log a - \log_a 2$, which is in $\mathcal{O}(\log N)$. Hence, $m$ and $2m$ are both lower bounded by $\mathcal{O}(\log N)$. Assume the computational worst case, in which we have $G_l = N$. Then, the runtime is in

$$\mathcal{O}(M_{\text{int}}(N^{1/4}/\sqrt{\log N})) = \mathcal{O}(M_{\text{int}}(N^{1/4} \sqrt{\log N})).$$

This proves the result. \qed

We may apply Theorem 1.1 to interesting subsets of $\mathbb{N}$, namely to Mersenne numbers $M_n = 2^n - 1$ and Fermat numbers $F_n = 2^{2^n} + 1$, $n \in \mathbb{N}$.

**Corollary 3.2.** Let $N \in \mathbb{N}$ be a Mersenne number or a Fermat number. Then $N$ is in $P_{2,1}$ and we may compute its prime factorization in $\mathcal{O}(M_{\text{int}}(N^{1/4} \sqrt{\log N}))$.

**References**

[BGS07] A. Bostan, P. Gaudry, and É. Schost, *Linear recurrences with polynomial coefficients and application to integer factorization and Cartier-Manin operator*, SIAM J. Comput. 36 (2007), no. 6, 1777–1806, DOI 10.1137/S009753970443793. MR2290425

[CH14] E. Costa and D. Harvey, *Faster deterministic integer factorization*, Math. Comp. 83 (2014), no. 285, 339–345, DOI 10.1090/S0025-5718-2013-02707-X. MR3120593

[CP05] R. Crandall and C. Pomerance, *Prime Numbers: A Computational Perspective*, 2nd ed., Springer, New York, 2005. MR2156291

[F09] M. Fürer, *Faster integer multiplication*, SIAM J. Comput. 39 (2009), no. 3, 979–1005, DOI 10.1137/070711761. MR2538847
[P94] C. H. Papadimitriou, *Computational Complexity*, Addison-Wesley Publishing Company, Reading, MA, 1994. MR1251285

[S77] V. Strassen, *Einige Resultate über Berechnungskomplexität*, Jber. Deutsch. Math.-Verein. 78 (1976/77), no. 1, 1–8. MR0438807

Hellbrunnerstrasse 34, A-5020 Salzburg, Austria

E-mail address: markus.hittmeir@sbg.ac.at