# Towards Optimal Bounds for Implicit Factorization Problem

Yao Lu, Liqiang Peng, Rui Zhang and Dongdai Lin

***Abstract*—We propose a new algorithm to solve the Implicit Factorization Problem, which was introduced by May and Ritzenhofen at PKC'09. In 2011, Sarkar and Maitra (IEEE TIT 57(6): 4002-4013, 2011) improved May and Ritzenhofen's results by making use of the technique for solving multivariate approximate common divisors problem. In this paper, based on the observation that the desired root of the equations that derived by Sarkar and Maitra contains large prime factors, which are already determined by some known integers, we develop new techniques to acquire better bounds. We show that our attack is the best among all known attacks, and give experimental results to verify the correctness. Additionally, for the first time, we can experimentally handle the implicit factorization for the case of balanced RSA moduli.**

***Index Terms*—lattices, Implicit Factorization Problem, Coppersmith's method, LLL algorithm, Small root**

## I. Introduction

The RSA cryptosystem is the most widely used public-key cryptosystem in practice, and its security relies on the difficulty of Integer Factorization Problem. It is conjectured that factoring cannot be solved in polynomial-time without quantum computers.

In Eurocrypt'85, Rivest and Shamir [18] first studied the factoring with known bits problem. They showed that $N = pq$ ($p, q$ is of the same bit size) can be factored given $\frac{2}{3}$-fraction of the bits of $p$. In 1996, Coppersmith [2] improved [18]'s bound to $\frac{1}{2}$. Note that for the above results, the unknown bits are within one consecutive block. The case of $n$ blocks was later considered in [7], [13].

Motivated by the cold boot attack [4], in Crypto'09, Heninger and Shacham [6] considered the case of known bits are uniformly spread over the factors $p$ and $q$, they presented a polynomial-time attack that works when ever a $0.57$-fraction of the bits of $p$ and $q$ is given. As a follow-up work, Henecka et al. [5] focused on the attack scenario that allowed for error correction of secret factors, which called Noisy Factoring Problem. Recently, Kunihiro et al. [10] discussed secret key recovery from noisy secret key sequences with both errors and erasures.

### A. Implicit Factorization Problem (IFP)

The above works require the knowledge of explicitly bits of secret factor. In PKC'09, May and Ritzenhofen [16] introduced a new factoring problem with implicit information, called Implicit Factorization Problem (IFP). Consider that $N_1 = p_1q_1, \ldots, N_k = p_kq_k$ be $n$-bit RSA moduli, where $q_1, \ldots, q_k$ are $\alpha n (\alpha \in (0,1))$-bit primes: Given the implicit information that $p_1, \ldots, p_k$ share certain portions of bit pattern, under what condition is it possible to factorize $N_1, \ldots, N_k$ efficiently? This problem can be applied in the area of malicious generation of RSA moduli, i.e. the construction of backdoored RSA moduli. Besides, it also helps to understand the complexity of the underlying factorization problem better.

Since then, there have been many cryptanalysis results for this problem [16], [3], [20], [12], [17], [19]. Recently, Sarkar and Maitra [20] developed a new approach, they used the idea of [9], which is for the approximate common divisor problem (ACDP), to solve the IFP, and managed to improve the previous bounds significantly.

We now give a brief review of their method. Suppose that primes $p_1, \ldots, p_k$ share certain amount of most significant bits (MSBs). First, they notice that

$$\gcd(N_1, N_2 + (p_1 - p_2)q_2, \ldots, N_k + (p_1 - p_k)q_k) = p_1$$

Then they try to solve the simultaneous modular univariate linear equations

$$\begin{cases} N_2 + u_2 = 0 \mod p_1 \\ \vdots \\ N_k + u_k = 0 \mod p_1 \end{cases} \quad (1)$$

for some unknown divisor $p_1$ of known modulus $N_1$. Note that if the root $(u_2^{(0)}, \ldots, u_k^{(0)}) = ((p_1 - p_2)q_2, \ldots, (p_1 - p_k)q_k)$ is small enough, we can extract them efficiently. In [20], Sarkar and Maitra proposed an algorithm to find the small root of equations (1). Recently, Lu et al. [12] performed a more effective analysis by making use of Cohn and Heninger's algorithm [1].

### B. Our Contributions

In this paper, we present a new algorithm to obtain better bounds for solving the IFP. As far as we are aware, our attack is the best among all known attacks.

Technically, our algorithm is also to find small root of Equations (1). Concretely, our improvement is based on the observation that for $2 \le i \le k$, $u_i^{(0)}$ contains a large prime $q_i$, which already determined by $N_i$.

Therefore, we separate $u_i$ into two unknown variables $x_i$ and $y_i$ i.e. $u_i = x_iy_i$. Consider the following equations

$$\begin{cases} N_2 + x_2y_2 = 0 \mod p_1 \\ \vdots \\ N_k + x_ky_k = 0 \mod p_1 \end{cases}$$

Yao Lu, Liqiang Peng, Rui Zhang and Dongdai Lin are with the State Key Laboratory of Information Security (SKLOIS), Institute of Information Engineering (IIE), Chinese Academy of Sciences (CAS) (e-mail: lywhhit@gmail.com; r-zhang@iie.ac.cn).

TABLE I
COMPARISON OF OUR GENERALIZE BOUNDS AGAINST PREVIOUS BOUNDS

| | [16] | [3] | [20] | [12] | [17] | this paper |
|---|---|---|---|---|---|---|
| $\beta n$-bit LSBs case ($\beta > \cdot$) | $\frac{k}{k-1}\alpha$ | - | $F(\alpha,k)$ | $H(\alpha,k)$ | $G(\alpha,k)$ | $T(\alpha,k)$ |
| $\gamma n$-bit MSBs case ($\gamma > \cdot$) | - | $\frac{k}{k-1}\alpha + \frac{6}{n}$ | $F(\alpha,k)$ | $H(\alpha,k)$ | $G(\alpha,k)$ | $T(\alpha,k)$ |
| $\gamma n$-bit MSBs and $\beta n$-bit LSBs together case ($\gamma + \beta > \cdot$) | - | - | $F(\alpha,k)$ | $H(\alpha,k)$ | $G(\alpha,k)$ | $T(\alpha,k)$ |
| $\delta n$-bit in the Middle case ($\delta > \cdot$) | - | $\frac{2k}{k-1}\alpha + \frac{7}{n}$ | - | - | - | - |

[1] $F(\alpha,k) = \frac{\alpha k^2 - (2\alpha+1)k + 1 + \sqrt{k^2 + 2\alpha^2 k - \alpha^2 k^2 - 2k + 1}}{k^2 - 3k + 2}$

[2] $H(\alpha,k) = 1 - (1-\alpha)^{\frac{k}{k-1}}$

[3] $G(\alpha,k) = \frac{k}{k-1}\left(\alpha - 1 + (1-\alpha)^{\frac{k+1}{k}} + (k+1)(1-(1-\alpha)^{\frac{1}{k}})(1-\alpha)\right)$

[4] $T(\alpha,k) = k(1-\alpha)\left(1 - (1-\alpha)^{\frac{1}{k-1}}\right)$

[5] The symbol "-" means that this corresponding case has not been considered.

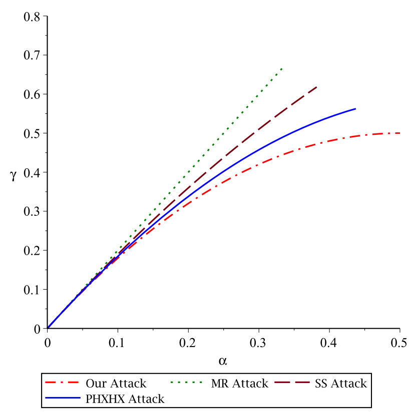


Fig. 1. Comparison with previous bounds on $\gamma$ with respect to $\alpha$: $k = 2$. MR Attack denotes May and Ritzenhofen's attack [16], SS Attack denotes Sarkar and Maitra's attack [20], PHXHX Attack denotes Peng et al.'s attack [17].

with the root $(x_2^{(0)}, \ldots, x_k^{(0)}, y_2^{(0)}, \ldots, y_k^{(0)}) = (q_2, \ldots, q_k, p_1 - p_2, \ldots, p_1 - p_k)$. Then we introduce $k-1$ new variables $z_i$ for the prime factor $p_i$ ($2 \le i \le k$), and use the equation $x_i z_i = N_i$ to decrease the determinate of the desired lattice. That is the key reason why get better results than [20].

In Fig 1, we give the comparison with previous bounds for the case $k = 2$. In Table I-B, we list the comparisons between our generalized bounds and the previous bounds.

Recently in [17], Peng et al. proposed another method for the IFP. Instead of applying Coppersmith's technique directly to the ACDP, Peng et al. utilized the lattice proposed by May and Ritzenhofen [16], and tried to find the coordinate of the desired vector which is not included in the reduced basis, namely they introduced a method to deal with the case when the number of shared bits is not large enough to satisfy the bound in [16].

In this paper, we also investigate Peng et al.'s method. Surprisingly, we get the same result with a different method. In Sec V, we give the experimental data for our two methods.

We organize the rest of the paper as follows. In Section II, we review the necessary background for our approaches. In Section III, based on new observations, we present our new analysis on the IFP. In Section IV, we revisit Peng et al.'s method [17]. Finally, in Sec V, we give the experimental data for the comparison with previous methods.

## II. Preliminaries

### A. Notations

Consider that $N_1 = p_1 q_1, \ldots, N_k = p_k q_k$ be $n$-bit RSA moduli, where $q_1, \ldots, q_k$ are $\alpha n (\alpha \in (0,1))$-bit primes. Three cases were considered in this paper, we list them below:

- $p_1, \ldots, p_k$ share $\beta n$ LSBs where $\beta \in (0,1)$;
- $p_1, \ldots, p_k$ share $\gamma n$ MSBs where $\gamma \in (0,1)$;
- $p_1, \ldots, p_k$ share $\gamma n$ MSBs and $\beta n$ LSBs together where $\gamma \in (0,1)$ and $\beta \in (0,1)$;

For simplicity, here we consider $\alpha n$, $\beta n$ and $\gamma n$ as integers.

### B. Lattice

Consider a set of linearly independent vectors $u_1, \ldots, u_w \in \mathbb{Z}^n$, with $w \leqslant n$. The lattice $\mathcal{L}$, spanned by $\{u_1, \ldots, u_w\}$, is the set of all integer linear combinations of the vectors $u_1, \ldots, u_w$. The number of vectors is the dimension of the lattice. The set $u_1, \ldots, u_w$ is called a basis of $\mathcal{L}$. In lattices with arbitrary dimension, finding the shortest vector is a very hard problem, however, approximations of a shortest vector can be obtained in polynomial-time by applying the well-known *LLL* basis reduction algorithm [11].

**Lemma 1** (LLL [11])**.** *Let $\mathcal{L}$ be a lattice of dimension $w$. In polynomial-time, the LLL algorithm outputs reduced basis vector $v_i$, $1 \leqslant i \leqslant w$ that satisfy*

$$\| v_1 \| \leqslant \| v_2 \| \leqslant \cdots \leqslant \| v_i \| \leqslant 2^{\frac{w(w-1)}{4(w+1-i)}} \det(\mathcal{L})^{\frac{1}{w+1-i}}$$

We also state a useful lemma from Howgrave-Graham [8]. Let $g(x_1, \ldots, x_k) = \sum_{i_1, \ldots, i_k} a_{i_1, \ldots, i_k} x_1^{i_1} \cdots x_k^{i_k}$. We define the norm of $g$ by the Euclidean norm of its coefficient vector: $||g||^2 = \sum_{i_1, \ldots, i_k} a_{i_1, \ldots, i_k}^2$.

**Lemma 2** (Howgrave-Graham [8]). *Let $g(x_1,\ldots,x_k) \in \mathbb{Z}[x_1,\ldots,x_k]$ be an integer polynomial that consists of at most $w$ monomials. Suppose that*

1) $g(y_1,\ldots,y_k) = 0 \mod p^m$ *for* $|\, y_1 \,| \leqslant X_1,\ldots,|\, y_k \,| \leqslant X_k$ *and*
2) $\|\, g(x_1X_1,\ldots,x_kX_k) \,\| < \frac{p^m}{\sqrt{w}}$

*Then $g(y_1,\cdots,y_k) = 0$ holds over the integers.*

The approach we used in the rest of the paper relies on the following heuristic assumption [15][7] for computing multivariate polynomials.

**Assumption 1.** *The lattice-based construction in this work yields algebraically independent polynomials, this common roots of these polynomials can be computed using techniques like calculation of the resultants or finding a* Gröbner basis.

## III. Our New Analysis for Implicit Factorization

As described in the previous section, we will use the fact the desired root of target equations contains large prime factors $q_i$ $(2 \leq i \leq k)$ which are already determined by $N_i$ to improve Sarkar's results.

### A. Analysis for Two RSA Moduli: the MSBs Case

**Theorem 1.** *Let $N_1 = p_1q_1, N_2 = p_2q_2$ be two different $n$-bit RSA moduli with $\alpha n$-bit $q_1, q_2$ where $\alpha \in (0,1)$. Suppose that $p_1, p_2$ share $\gamma n$ MSBs where $\gamma \in (0,1)$. Then under Assumption 1, $N_1$ and $N_2$ can be factored in polynomial-time if*

$$\gamma > 2\alpha(1-\alpha)$$

*Proof:* Let $\widetilde{p_2} = p_1 - p_2$. We have $N_1 = p_1q_1$, $N_2 = p_2q_2 = p_1q_2 - \widetilde{p_2}q_2$, and $\gcd(N_1, N_2 + \widetilde{p_2}q_2) = p_1$. Then we want to recover $q_2, \widetilde{p_2}$ from $N_1, N_2$. We focus on a bivariate polynomial $f(x,y) = N_2 + xy$ with the root $(x^{(0)}, y^{(0)}) = (q_2, \widetilde{p_2})$ modulo $p_1$. Let $X = N^{\alpha}, Y = N^{1-\alpha-\gamma}, Z = N^{1-\alpha}$ be the upper bounds of $q_2, \widetilde{p_2}, p_2$. Following we will use the fact that the small root $q_2$ is already determined by $N_2$ to improve Sarkar's results.

First let us introduce a new variable $z$ for $p_2$. We multiply the polynomial $f(x,y)$ by a power $z^s$ for some $s$ that has to be optimized. Additionally, we can replace every occurence of the monomial $xz$ by $N_2$. Define two integers $m$ and $t$, let us look at the following collection of trivariate polynomials that all have the root $(x_0, y_0)$ modulo $p_1^t$.

$$g_k(x,y,z) = z^s f^k N_1^{\max\{t-k,0\}} \qquad \text{for } k = 0,\ldots,m$$

For $g_k(x,y,z)$, we replace every occurrence of the monomial $xz$ by $N_2$ because $N_2 = p_2q_2$. Therefore, every monomial $x^ky^kz^s (k \geq s)$ with coefficient $a_k$ is transformed into a monomial $x^{k-s}y^k$ with coefficient $a_kN_2^s$. And every monomial $x^ky^kz^s (k < s)$ with coefficient $a_k$ is transformed into a monomial $y^kz^{s-k}$ with coefficient $a_kN_2^k$.

To keep the lattice determinant as small as possible, we try to eliminate the factor of $N_2^i$ in the coefficient of diagonal entry. Since $\gcd(N_1, N_2) = 1$, we only need multiplying the corresponding polynomial with the inverse of $N_2^i$ modulo $N_1^t$.

Compare to Sarkar's lattice, the coefficient vectors $g_k(xX, yY, zZ)$ of our lattice contain less powers of $X$, which decreases the determinant of the lattice spanned by these vectors, however, on the other hand, the coefficient vectors contain powers of $Z$, which in turn increases the determinant. Hence, there is a trade-off and one has to optimize the parameter $s$ subject to a minimization of the lattice determinant. That is the key reason why we can get better result than Sarkar's results.

We have to find two short vectors in lattice $\mathcal{L}$. Suppose that these two vectors are the coefficient vectors of two trivariate polynomial $f_1(xX, yY, zZ)$ and $f_2(xX, yY, zZ)$. There two polynomials have the root $(q_2, \widetilde{p_2})$ over the integers. Then we can eliminate the variable $z$ from these polynomials by setting $z = \frac{N_2}{x}$. Finally, we can extract the desired root $(q_2, \widetilde{p_2})$ from the new two polynomials if these polynomials are algebraically independent. Therefore, our attack relies on Assumption 1.

We are able to confirm Assumption 1 by various experiments later. This shows that our attack works very well in practice.

Now we give the details of the condition which we can find two sufficiently short vectors in the lattice $\mathcal{L}$. The determinate of the lattice $\mathcal{L}$ is

$$\det(\mathcal{L}) = N_1^{\frac{t(t+1)}{2}} X^{\frac{(m-s)(m-s+1)}{2}} Y^{\frac{m(m+1)}{2}} Z^{\frac{s(s+1)}{2}}$$

The dimension of the lattice is $w = m+1$.

To get two polynomials which sharing the root $q_2, \widetilde{p_2}, p_2$, we get the condition

$$2^{\frac{w(w-1)}{4w}} \det(\mathcal{L})^{\frac{1}{w}} < \frac{p_1^t}{\sqrt{w}}$$

Substituting the values of the $\det(\mathcal{L})$ and neglecting low-order term, we obtain the new condition

$$\frac{t^2}{2} + \alpha\frac{(m-s)^2}{2} + (1-\alpha-\gamma)\frac{m^2}{2} + (1-\alpha)\frac{s^2}{2} < (1-\alpha)tm$$

Let $t = \tau m, s = \sigma m$, the optimized values of parameters $\tau$ and $\sigma$ were given by

$$\tau = 1-\alpha \qquad \sigma = \alpha$$

Plugging in this values, we finally end up with the condition

$$\gamma > 2\alpha(1-\alpha)$$

■

One can refer to Fig. 1 for the comparison with previous theoretical results.

### B. Extension to $k$ RSA Moduli

In this section, we give an analysis for $k$ $(k > 2)$ RSA moduli.

**Theorem 2.** *Let $N_1 = p_1q_1,\ldots,N_k = p_kq_k$ be $k$ different $n$-bit RSA moduli with $\alpha n$-bit $q_1,\ldots,q_k$ where $\alpha \in (0,1)$. Suppose that $p_1,\ldots,p_k$ share $\gamma n$ MSBs where $\gamma \in (0,1)$. Then under Assumption 1, $N_1,\ldots,N_k$ can be factored in polynomial-time if*

$$\gamma > k(1-\alpha)\left(1-(1-\alpha)^{\frac{1}{k-1}}\right)$$

*Proof:* Let $\widetilde{p_i} = p_1 - p_i$. We have $N_1 = p_1 q_1$ and $N_i = p_i q_i = p_1 q_i - \widetilde{p_i} q_i$ $(2 \le i \le k)$. We have $\gcd(N_1, N_2 + \widetilde{p_2} q_2, \ldots, N_k + \widetilde{p_k} q_k) = p_1$. Then we want to recover $q_i, \widetilde{p_i}$ $(2 \le i \le k)$ from $N_1, \ldots, N_k$. We construct a system of $k-1$ polynomials

$$\begin{cases} f_2(x_2, y_2) = N_2 + x_2 y_2 \\ \qquad\vdots \\ f_k(x_k, y_k) = N_k + x_k y_k \end{cases}$$

with the root $(x_2^{(0)}, y_2^{(0)}, \ldots, x_k^{(0)}, y_k^{(0)}) = (q_2, \widetilde{p_2}, \ldots, q_k, \widetilde{p_k})$ modulo $p_1$. Using the similar technique of Theorem 1, and introducing $k-1$ new variable $z_i$ for $p_i$ $(2 \le i \le k)$, we define the following collection of trivariate polynomials.

$$\begin{aligned} &g_{i_2,\ldots,i_k}(x_2, \ldots, x_k, y_2, \ldots, y_k, z_2, \ldots, z_k) \\ &= (z_2 \cdots z_k)^s f_2^{i_2} \cdots f_k^{i_k} N_1^{\max\{t - i_2 - \cdots - i_k, 0\}} \end{aligned}$$

with $0 \le i_2 + \cdots + i_k \le m$ (Because of the asymmetric nature of the unknown variables $x_2, \ldots, x_k$, we use the same parameter $s$).

For $g_{i_2,\ldots,i_k}$, we replace every occurrence of the monomial $x_i z_i$ by $N_i$. We can eliminate the factor of $N_2^{j_2} \cdots N_k^{j_k}$ in the coefficient of diagonal entry. The determinate of the lattice $\mathcal{L}$ is

$$\det(\mathcal{L}) = N_1^{s_N} \prod_{i=2}^{k} X_i^{s_{X_i}} Y_i^{s_{Y_i}} Z_i^{s_{Z_i}}$$

where

$$\begin{aligned} s_N &= \sum_{j=0}^{t} j \binom{t-j+k-2}{k-2} \\ &= \binom{t+k-1}{k-1} \frac{t}{k} \\ s_{X_2} = \cdots = s_{X_k} &= \sum_{j=0}^{m-s} j \binom{m-s-j+k-2}{k-2} \\ &= \binom{m-s+k-1}{k-1} \frac{m-s}{k} \\ s_{Y_2} = \cdots = s_{Y_k} &= \sum_{j=0}^{m} j \binom{m-j+k-2}{k-2} \\ &= \binom{m+k-1}{k-1} \frac{m}{k} \\ s_{Z_2} = \cdots = s_{Z_k} &= \sum_{j=0}^{s} j \binom{m-s+j+k-2}{k-2} \\ &= \binom{m+k-1}{k} \frac{ks-m}{m} \\ &+ \binom{m-s-1+k-1}{k} \frac{k+m-s-1}{m-s-1} \end{aligned}$$

Here $X_i = N^{\alpha}, Y_i = N^{1-\alpha-\gamma}, Z_i = N^{1-\alpha}$ are the upper bounds of $q_i, \widetilde{p_i}, p_i$. The dimension of the lattice is

$$w = \dim(\mathcal{L}) = \sum_{j=0}^{m} \binom{j+k-2}{j} = \binom{m+k-1}{m}$$

To get $2k-2$ polynomials which sharing the root $q_2, \widetilde{p_2}, p_2$, we get the condition

$$2^{\frac{w(w-1)}{4(w+4-2k)}} \det(\mathcal{L})^{\frac{1}{w+4-2k}} < \frac{p_1^t}{\sqrt{w}}$$

Substituting the values of the $\det(\mathcal{L})$ and neglecting low-order term, we obtain the new condition

$$\begin{aligned} &\binom{t+k-1}{k-1} \frac{t}{k} + (k-1)\alpha \binom{m-s+k-1}{k-1} \frac{m-s}{k} \\ &+ (k-1)(1-\alpha-\gamma) \binom{m+k-1}{k-1} \frac{m}{k} \\ &+ (k-1)(1-\alpha) \binom{m+k-1}{k} \frac{ks-m}{m} \\ &+ (k-1)(1-\alpha) \binom{m-s-1+k-1}{k} \frac{k+m-s-1}{m-s-1} \\ &< (1-\alpha)t \binom{m+k-1}{m} \end{aligned}$$

Let $t = \tau m, s = \sigma m$, the optimized values of parameters $\tau$ and $\sigma$ were given by

$$\tau = (1-\alpha)^{\frac{1}{k-1}} \qquad \sigma = 1 - (1-\alpha)^{\frac{1}{k-1}}$$

Plugging in this values, we finally end up with the condition

$$\gamma > k(1-\alpha)\left(1 - (1-\alpha)^{\frac{1}{k-1}}\right)$$

■

One can refer to Table I-B for the comparison with previous theoretical results.

### C. Extension to the LSBs Case

Following we show a similar result in the case of $p_1, \ldots, p_k$ share some MSBs and LSBs together. This also takes care of the case when only LSBs are shared.

**Theorem 3.** *Let $N_1 = p_1 q_1, \ldots, N_k = p_k q_k$ be $k$ different $n$-bit RSA moduli with $\alpha n$-bit $q_i$ $(\alpha \in \{0,1\})$. Suppose that $p_1, \cdots, p_k$ share $\gamma n$ MSBs $(\gamma \in \{0,1\})$ and $\beta n$ LSBs $(\beta \in \{0,1\})$ together. Then under Assumption 1, $N_1, \cdots, N_k$ can be factored in polynomial-time if*

$$\gamma + \beta > k(1-\alpha)\left(1 - (1-\alpha)^{\frac{1}{k-1}}\right)$$

*Proof:* Suppose that $p_1, \ldots, p_k$ share $\gamma n$ MSBs and $\beta n$ LSBs together. Then we have the following equations:

$$\begin{cases} p_2 = p_1 + 2^{\beta n} \tilde{p}_2 \\ \qquad\vdots \\ p_k = p_1 + 2^{\beta n} \tilde{p}_k \end{cases}$$

We can write as follows

$$N_i q_1 - N_1 q_i = 2^{\beta n} \tilde{p}_i q_1 q_i \quad \text{for } 2 \le i \le k$$

Then we get

$$(2^{\beta n})^{-1} N_i q_1 - \tilde{p}_i q_1 q_i \equiv 0 \mod N_1 \quad \text{for } 2 \le i \le k$$

Let $A_i \equiv (2^{\beta n})^{-1} N_i \mod N_1$ for $2 \le i \le k$. Thus, we have

$$\begin{cases} A_2 - q_2 \tilde{p_2} \equiv 0 \mod p_1 \\ \vdots \\ A_k - q_k \tilde{p_k} \equiv 0 \mod p_1 \end{cases}$$

Then we can construct a system of $k-1$ polynomials

$$\begin{cases} f_2(x_2, \cdots, x_k) = A_2 + x_2 y_2 \\ \vdots \\ f_k(x_2, \cdots, x_k) = A_k + x_k y_k \end{cases}$$

with the root $(x_2^{(0)}, y_2^{(0)}, \ldots, x_k^{(0)}, y_k^{(0)}) = (q_2, \widetilde{p_2}, \ldots, q_k, \widetilde{p_k})$ modulo $p_1$. The rest of the proof follows the similar technique as in the proof of Theorem 2. We omit the details here. ■

## IV. Revisiting Peng et al.'s Method [17]

In [17], Peng et al. gave a new idea for IFP. Recall the method proposed by May and Ritzenhofen in [16], the lower bound on the number of shared LSBs has been determined, which can ensure the vector $(q_1, \cdots, q_k)$ is shortest in the lattice, namely the desired factorization can be obtained by lattice basis reduction algorithm.

Peng et al. took into consideration the lattice introduced in [16] and discussed a method which can deal with the case when the number of shared LSBs is not enough to ensure that the desired factorization cannot be solved out by applying reduction algorithms to the lattice. More narrowly, since $(q_1, \cdots, q_k)$ is in the lattice, it can be represented as a linear combination of reduced lattice basis. Hence the problem of finding $(q_1, \cdots, q_k)$ is transformed into solving a homogeneous linear equation with unknown moduli. Peng et al. utilized the result from Herrmann and May [7] to solve out the linear modulo equation and obtain a better result.

In this section, we revisit Peng et al.'s method and modify the construction of lattice which is used to solve the homogeneous linear modulo equation. Therefore, a further improved bound on the shared LSBs and MSBs is obtained.

Firstly, we recall the case of primes shared LSBs. Assume that there are $k$ different $n$-bit RSA moduli $N_1 = p_1 q_1, \cdots, N_k = p_k q_k$, where $p_1, \cdots, p_k$ share $\gamma n$ LSBs and $q_1, \cdots, q_k$ are $\alpha n$-bit primes. The moduli can be represented as

$$\begin{cases} N_1 = (p + 2^{\gamma n} \widetilde{p_1}) q_1 \\ \vdots \\ N_k = (p + 2^{\gamma n} \widetilde{p_k}) q_k \end{cases}$$

Furthermore, we can get following modular equations

$$\begin{cases} N_1^{-1} N_2 q_1 - q_2 \equiv 0 \mod 2^{\gamma n} \\ \vdots \\ N_1^{-1} N_k q_1 - q_k \equiv 0 \mod 2^{\gamma n} \end{cases} \tag{2}$$

In [16], May and Ritzenhofen introduced a $k$-dimensional lattice $\mathcal{L}_1$ which is generated by the row vectors of following matrix

$$\begin{pmatrix} 1 & N_1^{-1}N_2 & N_1^{-1}N_3 & \cdots & N_1^{-1}N_k \\ 0 & 2^{\gamma n} & 0 & \cdots & 0 \\ 0 & 0 & 2^{\gamma n} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 2^{\gamma n} \end{pmatrix}.$$

Since (2) holds, the vector $(q_1, \cdots, q_k)$ is the shortest vector in $\mathcal{L}_1$ with a good possibility when $\gamma \ge \frac{k}{k-1}\alpha$. Then by applying the $LLL$ reduction algorithm to the lattice, the vector $(q_1, \cdots, q_k)$ can be solved out. Conversely, when $\gamma < \frac{k}{k-1}\alpha$ the reduced basis $(\lambda_1, \cdots, \lambda_k)$ doesn't contain vector $(q_1, \cdots, q_k)$, nevertheless, we can represent the vector $(q_1, \cdots, q_k)$ into the form with a linear combination of reduced basis. Namely, there exist integers $x_1, x_2, \cdots, x_k$ such that $(q_1, \cdots, q_k) = x_1\lambda_1 + \cdots + x_k\lambda_k$. Moreover, the following system of modular equations can be obtained,

$$\begin{cases} x_1 l_{11} + x_2 l_{21} + \cdots + x_k l_{k1} = q_1 \equiv 0 \mod q_1 \\ \vdots \\ x_1 l_{1k} + x_2 l_{2k} + \cdots + x_k l_{kk} = q_k \equiv 0 \mod q_k \end{cases} \tag{3}$$

where $\lambda_i = (l_{i1}, l_{i2}, \cdots, l_{ik})$, $i = 1, 2, \cdots, k$.

Based on the Gaussian heuristic, we have a rough estimation on the size of the reduced basis. We estimate the length of $\lambda_i$ and the size of $l_{ij}$ as $\det(L_2)^{\frac{1}{k}} = 2^{\frac{nt(k-1)}{k}}$, hence the solution of (3) is $|x_i| \approx \frac{q_i}{k l_{ij}} \approx 2^{\alpha n - \frac{nt(k-1)}{k} - \log_2 k} \le 2^{\alpha n - \frac{nt(k-1)}{k}}$.

Then using the Chinese Remainder Theorem, from (3) we can get the following homogeneous modular equation

$$a_1 x_1 + a_2 x_2 + \cdots + a_k x_k \equiv 0 \mod q_1 q_2 \cdots q_k \tag{4}$$

where $a_i$ is an integer satisfying $a_i \equiv l_{ij} \bmod N_j$ for $1 \le j \le k$ and it can be calculated from the $l_{ij}$ and $N_j$.

For this linear modular equation, Peng et al. directly utilized the method of Herrmann and May [7] to solve it and obtain that when

$$\gamma \ge \frac{k}{k-1}(\alpha - 1 + (1-\alpha)^{\frac{k+1}{k}} + (k+1)(1-(1-\alpha)^{\frac{1}{k}})(1-\alpha)$$

the desired solution can be solved out.

In this paper, we notice that the linear modular equation is homogeneous which is a variant of Herrmann and May's equation, hence we utilize the following theorem which is proposed by Lu et al. in [14] to modify the construction of lattice used in [17].

**Theorem 4.** *Let $N$ be a sufficiently large composite integer (of unknown factorization) with a divisor $p$ ($p \ge N^\beta$). Furthermore, let $f(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ be a homogenous linear polynomial in $n (n \ge 2)$ variables. Under Assumption 1, we can find all the solutions $(y_1, \ldots, y_n)$ of the equation $f(x_1, \ldots, x_n) = 0 \pmod p$ with $\gcd(y_1, \ldots, y_n) = 1$, $|y_1| \le N^{\gamma_1}, \ldots |y_n| \le N^{\gamma_n}$ if*

$$\sum_{i=1}^{n} \gamma_i < \left(1 - (1-\beta)^{\frac{n}{n-1}} - n(1-\beta)\left(1 - \sqrt[n-1]{1-\beta}\right)\right)$$

*The running time of the algorithm is polynomial in $\log N$ but exponential in $n$.*

TABLE II
THEORETICAL AND EXPERIMENTAL DATA OF THE NUMBER OF SHARED MSBS IN [20] AND SHARED MSBS IN OUR METHOD IN SEC. III

| $k$ | bitsize of $(p_i, q_i)$, i.e., $((1-\alpha)\log_2 N_i, \alpha\log_2 N_i)$ | No. of shared MSBs in $p_i$ [20] | | | | No. of shared MSBs in $p_i$ (Sec. III) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | theo. | expt. | dim | time(sec) | theo. | expt. | dim | time(sec) |
| 2 | (874,150) | 278 | 289 | 16 | 1.38 | 257 | 265 | 46 | 5572.75 |
| 2 | (824,200) | 361 | 372 | 16 | 1.51 | 322 | 330 | 46 | 4967.07 |
| 2 | (774,250) | 439 | 453 | 16 | 1.78 | 378 | 390 | 46 | 4273.77 |
| 2 | (724,300) | 513 | 527 | 16 | 2.14 | 425 | 435 | 46 | 2117.31 |
| 3 | (874,150) | 217 | 230 | 56 | 29.24 | 200 | 225 | 136 | 6898.65 |
| 3 | (824,200) | 286 | 304 | 56 | 36.28 | 255 | 280 | 136 | 10613.38 |
| 3 | (774,250) | 352 | 375 | 56 | 51.04 | 304 | 335 | 136 | 18757.73 |
| 3 | (724,300) | 417 | 441 | 56 | 70.55 | 346 | 375 | 136 | 6559.34 |
| 3 | (674,350) | 480 | 505 | 56 | 87.18 | 382 | 415 | 136 | 12340.21 |
| 3 | (624,400) | 540 | 569 | 56 | 117.14 | 410 | 450 | 136 | 14823.92 |
| 3 | (512,512) | - | - | - | - | 450 | 480 | 136 | 7326.63 |

TABLE III
THEORETICAL AND EXPERIMENTAL DATA OF THE NUMBER OF SHARED MSBS IN [17] AND SHARED MSBS IN OUR METHOD IN SEC. IV

| $k$ | bitsize of $(p_i, q_i)$, i.e., $((1-\alpha)\log_2 N_i, \alpha\log_2 N_i)$ | No. of shared MSBs in $p_i$ [17] | | | | No. of shared MSBs in $p_i$ (Sec. IV) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | theo. | expt. | dim | time(sec) | theo. | expt. | dim | time(sec) |
| 2 | (874,150) | 267 | 278 | 190 | 1880.10 | 257 | 265 | 46 | 498.17 |
| 2 | (824,200) | 340 | 357 | 190 | 1899.21 | 322 | 333 | 46 | 771.78 |
| 2 | (774,250) | 405 | 412 | 190 | 2814.84 | 378 | 390 | 46 | 1248.98 |
| 2 | (724,300) | 461 | 470 | 190 | 2964.74 | 425 | 435 | 46 | 2016.00 |
| 3 | (874,150) | 203 | 225 | 220 | 5770.99 | 200 | 218 | 120 | 5802.06 |
| 3 | (824,200) | 260 | 288 | 220 | 6719.03 | 255 | 280 | 120 | 8688.47 |
| 3 | (774,250) | 311 | 343 | 220 | 6773.48 | 304 | 340 | 120 | 10856.42 |
| 3 | (724,300) | 356 | 395 | 220 | 7510.86 | 346 | 375 | 120 | 31364.93 |
| 3 | (674,350) | 395 | 442 | 220 | 8403.91 | 382 | 420 | 120 | 39123.82 |
| 3 | (624,400) | 428 | 483 | 220 | 9244.42 | 410 | 450 | 120 | 83035.58 |
| 3 | (512,512) | 476 | - | - | - | 450 | 490 | 120 | 166932.36 |

For this homogeneous linear equation (4) in $k$ variables modulo $q_1q_2\cdots q_k \approx (N_1N_2\cdots N_k)^\alpha$, by Theorem 4 with the variables $x_i < (N_1N_2\cdots N_k)^{\delta_i} \approx 2^{k\delta_i n}$, $i = 1, 2, \cdots, k$, we can solve out the variables when

$$\sum_{i=1}^{k}\delta_i \approx k\delta_i \le 1-(1-\alpha)^{\frac{k}{k-1}} - k(1-\alpha)\left(1-(1-\alpha)^{\frac{1}{k-1}}\right)$$

where $\delta_1 \approx \delta_2 \approx \cdots \approx \delta_k$.

Hence, when

$$\alpha - \frac{\gamma(k-1)}{k} \le 1-(1-\alpha)^{\frac{k}{k-1}} - k(1-\alpha)\left(1-(1-\alpha)^{\frac{1}{k-1}}\right)$$

Namely,

$$\gamma \ge \frac{k}{k-1}\left(\alpha - 1 + (1-\alpha)^{\frac{k}{k-1}} + k(1-(1-\alpha)^{\frac{1}{k-1}})(1-\alpha)\right)$$
$$= (1-\alpha)\left(1-(1-\alpha)^{\frac{1}{k-1}}\right)$$

the desired vector can be found out.

The above result can be easily extend to MSBs case using the technique in [17]. Surprisingly we get the same result as Theorem 2 by modifying Peng et al.'s technique.

## V. EXPERIMENTAL RESULTS

We implemented our analysis in Sage 5.12 computer algebra system on a laptop with Intel(R) Core(TM) Duo CPU (2.53GHz, with1.9GB RAM in the guest OS Ubuntu 13.10 with Windows 7 as the host OS).

Since our method of Sec. III is based on an optimized method of [20], we present some numerical values for comparisons between these two methods in Table IV.

Meanwhile our method of Sec. IV is based on an improved method of [17], we present some numerical values for comparison with these two methods in Table IV.

In particular, for the first time, we can experimentally handle the IFP for the case of balanced RSA moduli.

## REFERENCES


[1] H. Cohn and N. Heninger. Approximate common divisors via lattices. *Algorithmic Number Theory Symposium*, 2012. 1

[2] D. Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *Journal of Cryptology*, 10(4):233–260, 1997. 1

[3] J.C. Faugère, R. Marinier, and G. Renault. Implicit factoring with shared most significant and middle bits. *Public Key Cryptography–PKC 2010*, pages 70–87, 2010. 1, 2

[4] J.A. Halderman, S.D. Schoen, N. Heninger, W. Clarkson, W. Paul, J.A. Calandrino, A.J. Feldman, J. Appelbaum, and E.W. Felten. Lest we remember: cold-boot attacks on encryption keys. *Communications of the ACM*, 52(5):91–98, 2009. 1

[5] W. Henecka, A. May, and A. Meurer. Correcting errors in RSA private keys. *Advances in Cryptology–CRYPTO 2010*, pages 351–369, 2010. 1

[6] N. Heninger and H. Shacham. Reconstructing RSA private keys from random key bits. *Advances in Cryptology-CRYPTO 2009*, pages 1–17, 2009. 1

[7] M. Herrmann and A. May. Solving linear equations modulo divisors: On factoring given any bits. *Advances in Cryptology-ASIACRYPT 2008*, pages 406–424, 2008. 1, 3, 5

[8] N. Howgrave-Graham. Finding small roots of univariate modular equations revisited. *Crytography and Coding*, pages 131–142, 1997. 2, 3

[9] N. Howgrave-Graham. Approximate integer common divisors. *Cryptography and Lattices*, pages 51–66, 2001. 1

[10] N. Kunihiro, N. Shinohara, and T. Izu. Recovering RSA secret keys from noisy key bits with erasures and errors. In *Public-Key Cryptography–PKC 2013*, pages 180–197. Springer, 2013. 1

[11] A.K. Lenstra, H.W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982. 2

[12] Y. Lu, R. Zhang, and D. Lin. Improved bounds for the implicit factorization problem. *Adv. in Math. of Comm.*, 7(3):243–251, 2013. 1, 2

[13] Y. Lu, R. Zhang, and D. Lin. Factoring multi-power RSA modulus $N = p^r q$ with partial known bits. In *Information Security and Privacy*, pages 57–71. Springer, 2013. 1

[14] Y. Lu and R. Zhang and D. Lin. New results on solving linear equations modulo unknown divisors and its applications. *Cryptology ePrint Archive, Report 2014/343*, 2014. 5

[15] A. May. *New RSA vulnerabilities using lattice reduction methods*. PhD thesis, 2003. 3

[16] A. May and M. Ritzenhofen. Implicit factoring: On polynomial time factoring given only an implicit hint. *Public Key Cryptography–PKC 2009*, pages 1–14, 2009. 1, 2, 5

[17] L. Peng, L. Hu, J. Xu, Z. Huang, and Y. Xie. Further improvement of factoring RSA moduli with implicit hint. *AFRICACRYPT 2014*, pages 165-177, 2014. 1, 2, 5, 6

[18] R. Rivest and A. Shamir. Efficient factoring based on partial information. In *Advances in Cryptology–EUROCRYPT'85*, pages 31–34. Springer, 1986. 1

[19] S. Sarkar and S. Maitra. Further results on implicit factoring in polynomial time. *Adv. in Math. of Comm.*, 3(2):205–217, 2009. 1

[20] S. Sarkar and S. Maitra. Approximate integer common divisor problem relates to implicit factorization. *IEEE Transactions on Information Theory*, 57(6):4002–4013, 2011. 1, 2, 6