SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2018, Vol. 61 032109:1–032109:10
doi: 10.1007/s11432-017-9176-5

A better bound for implicit factorization problem with shared middle bits

Shixiong WANG[1], Longjiang QU[2,3*], Chao LI[1,3] & Shaojing FU[1,2]

[1]*College of Computer, National University of Defense Technology, Changsha* 410073*, China;*
[2]*State Key Laboratory of Cryptology, Beijing* 100878*, China;*
[3]*College of Science, National University of Defense Technology, Changsha* 410073*, China*

Received 7 February 2017/Accepted 21 June 2017/Published online 26 October 2017

Abstract This paper presents our investigation of the implicit factorization problem, where unknown prime factors of two RSA moduli share a certain number of middle bits. The problem is described as follows. Let $N_1 = p_1q_1, N_2 = p_2q_2$ be two different n-bit RSA moduli, where q_1, q_2 are both αn-bit prime integers. Suppose that p_1, p_2 share tn bits at positions from t_1n to $t_2n = (t_1 + t)n$. Then this problem focuses on the condition about t, α to factor N_1, N_2 efficiently. At PKC 2010, Faugère et al. showed that N_1, N_2 can be factored when $t > 4\alpha$. Subsequently, in 2015, Peng et al. improved this bound to $t > 4\alpha - 3\alpha^2$. In this paper, we directly apply Coppersmith's method to the implicit factorization problem with shared middle bits, and a better bound $t > 4\alpha - 4\alpha^{\frac{3}{2}}$ is obtained. The correctness of our approach is verified by experiments.

Keywords RSA, implicit factorization problem, middle bits, lattice, Coppersmith's method

Citation Wang S X, Qu L J, Li C, et al. A better bound for implicit factorization problem with shared middle bits. Sci China Inf Sci, 2018, 61(3): 032109, doi: 10.1007/s11432-017-9176-5

1 Introduction

In 1978, Rivest et al. [1] proposed the well-known RSA public key cryptosystem. Since then, much effort has been made to evaluate the security of RSA because of its wide variety of applications. For example, RSA is vulnerable in the case of either a small public exponent [2, 3] or a small private exponent [4, 5]. Some attacks were also presented when a portion of the private key is exposed [6–11]. From the work of [12, 13], it was proved that recovering the private key and factoring the modulus are determinately equivalent in polynomial time. In addition, both Luo et al. [14] and Zheng et al. [15], in 2009 and 2016, respectively, found some weak keys for the RSA public key cryptosystem.

In this paper, we concentrate on the implicit factorization problem (IFP). Suppose that for two n-bit RSA moduli $N_1 = p_1q_1$ and $N_2 = p_2q_2$, q_1, q_2 are both αn-bit prime integers, and p_1, p_2 share tn bits. Then the IFP becomes the problem of determining the conditions under which there exists an efficient algorithm to factor N_1, N_2. It also includes the generalized case of more RSA moduli.

At PKC 2009, May and Ritzenhofen [16] firstly introduced the IFP. They considered the case where p_1, p_2 share the least significant bits (LSBs), and claimed that the two RSA moduli N_1, N_2 can be factored when $t \geqslant 2\alpha + \frac{3}{n}$. It was proved in [16] that under the condition $t \geqslant 2\alpha + \frac{3}{n}$, (q_1, q_2) is the shortest vector in a two-dimensional lattice they constructed. Then by some lattice basis reduction method, q_1, q_2 were recovered and N_1, N_2 were factored. Subsequently, at PKC 2010, Faugère et al. [17] analyzed two

[*] Corresponding author (email: ljqu_happy@hotmail.com)

Table 1 Previous bounds and our contribution to the IFP

Case	[16]	[17]	[21, 22]	[23]	[24]	[25]	This paper
LSBs ($t > \cdot$)	2α	–	$2\alpha - \alpha^2$	$4 - 4\alpha - 4(1-\alpha)^{\frac{3}{2}}$	$2\alpha - 2\alpha^2$	–	–
MSBs ($t > \cdot$)	–	2α	$2\alpha - \alpha^2$	$4 - 4\alpha - 4(1-\alpha)^{\frac{3}{2}}$	$2\alpha - 2\alpha^2$	–	–
Middle bits ($t > \cdot$)	–	$\boldsymbol{4\alpha}$	–	–	–	$\boldsymbol{4\alpha - 3\alpha^2}$	$\boldsymbol{4\alpha - 4\alpha^{\frac{3}{2}}}$

cases where p_1, p_2 share the most significant bits (MSBs) or the middle bits. Similarly, for the case of shared MSBs they got the bound $t \geqslant 2\alpha + \frac{3}{n}$ using a two-dimensional lattice. For the case of shared middle bits, they obtained a heuristic result that q_1, q_2 can be found from a three-dimensional lattice when $t \geqslant 4\alpha + \frac{7}{n}$.

Both [16, 17] used lattice-based methods involving the construction of either a two-dimensional lattice or a three-dimensional lattice. Different from them, another lattice-based method widely adopted by researchers for cryptanalysis is Coppersmith's method. Coppersmith's method is used to find small roots of v-variate modular polynomial equations or $(v+1)$-variate integer polynomial equations in polynomial time based on lattice basis reduction. Initially in 1996, Coppersmith [2, 18] obtained results for the case of $v = 1$. Later the methods of [2, 18] were reformulated by Howgrave-Graham [19] and Coron [20] respectively in simpler ways. The aforementioned two reformulations can also be extended to the case of $v \geqslant 2$, in which the results are based on an assumption and are thus heuristic. In general, the reformulations are used when we refer to Coppersmith's method.

From [16, 17] we know that the IFP mainly includes three cases where p_1, p_2 share MSBs, LSBs, or middle bits. For the case of shared MSBs and shared LSBs, respectively, the bounds have been simultaneously improved several times by means of lattice-based methods. In 2011, Sarkar and Maitra [21] related the Approximate Common Divisor Problem (ACDP) to the IFP, and a better bound $t \geqslant 2\alpha - \alpha^2 + \varepsilon$ was obtained for these two cases. Here "ε" is a small constant that depends on the bit length n and the dimension of the lattice constructed in [21]. The same applies for the following results. The bound obtained by Lu et al. [22] is also $t \geqslant 2\alpha - \alpha^2 + \varepsilon$. However, when generalized to k ($k > 3$) RSA moduli, the result in [22] is better than that of [21]. Both the methods used in [21, 22] are essentially Coppersmith's method. In 2014, Peng et al. [23] combined Coppersmith's method with the method in [16, 17], and thus improved the bound to $t \geqslant 4 - 4\alpha - 4(1-\alpha)^{\frac{3}{2}} + \varepsilon$. Finally, in 2015, Lu et al. [24] investigated all the above methods and acquired the best bound $t \geqslant 2\alpha - 2\alpha^2 + \varepsilon$ among all known attacks.

For the case of shared middle bits, Peng et al. [25] recently obtained a new bound $t \geqslant 4\alpha - 3\alpha^2 + \varepsilon$, which improved the bound $t \geqslant 4\alpha + \frac{7}{n}$ in [17]. Similar to [23], Ref. [25] first utilized the lattice proposed in [17] to obtain a reduced basis, and then acquired the desired vector from this reduced basis by using Coppersmith's method. In this paper, we directly apply Coppersmith's method to the IFP with shared middle bits, and obtain a better bound $t \geqslant 4\alpha - 4\alpha^{\frac{3}{2}} + \varepsilon$ than [25].

Table 1 summarizes all the above existing bounds and our contribution. For the sake of simplicity, the small constants "$\frac{3}{n}, \frac{7}{n}, \varepsilon$" are omitted from Table 1. In this paper, we focus on the case of shared middle bits. In this regard, Figure 1 illustrates the comparison between our result and previous work in [17, 25]. Since $(1-\alpha)n$-bit p_1, p_2 share tn bits, we have $t < 1 - \alpha$. Thus, any valid range of t with respect to α lies below the diagonal line in Figure 1. According to Figure 1, we know the result given in [17] is only valid when $0 < \alpha < 0.2$, whereas that obtained in [25] applies for $0 < \alpha < \frac{5-\sqrt{13}}{6} \approx 0.2324$. In this paper, we extend the range of α to $0 < \alpha < \frac{9+\sqrt{17}}{32} \approx 0.4101$, and our new improvement is denoted by the red area in Figure 1. From Figure 1, one can see that our bound is better than those in [17, 25].

Our result is based on Coppersmith's method for finding the small roots of multivariate modular polynomial equations. Thus, it relies on Assumption 1, which is introduced in Section 2 and is examined through experiments in Section 4. Ignoring "ε" just as [25], we conclude our contribution as follows.

Theorem 1. Let $N_1 = p_1 q_1, N_2 = p_2 q_2$ be two different n-bit RSA moduli, where q_1, q_2 are both αn-bit prime integers. Suppose that p_1, p_2 share tn bits at positions from $t_1 n$ to $t_2 n = (t_1 + t)n$. Then under Assumption 1, N_1 and N_2 can be factored in polynomial time if

$$t > 4\alpha - 4\alpha^{\frac{3}{2}}.$$

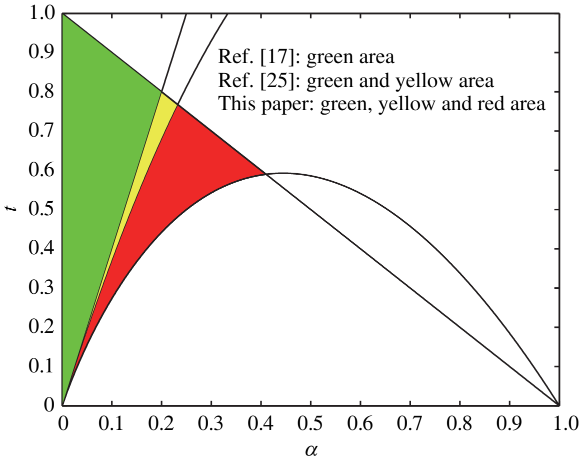

Figure 1 Comparison between our result and previous work in [17, 25].

The remainder of this paper is organized as follows. In Section 2, we introduce lattice-based Coppersmith's method, mainly including Howgrave-Graham's lemma and the LLL algorithm. Section 3 presents our new analysis for the IFP with shared middle bits. A new lattice for Coppersmith's method is constructed and the proof of Theorem 1 is provided in this section. In Section 4, we examine the justification of our approaches through some experiments. Section 5 is the conclusion.

2 Preliminaries

In this section, we introduce Coppersmith's method for finding the small roots of multivariate modular polynomial equations, which will be used in the proof of Theorem 1 in Section 3. First of all, let us recall the definition of lattice.

Definition 1. Let $b_1, b_2, \ldots, b_\omega \in \mathbb{R}^s$ be linearly independent row vectors for $\omega \leqslant s$. A lattice Λ generated by $b_1, b_2, \ldots, b_\omega$ is the set of all integral linear combinations of these vectors:

$$\Lambda = \mathrm{span}_{\mathbb{Z}}(b_1, b_2, \ldots, b_\omega) = \left\{ \sum_{i=1}^{\omega} x_i b_i \mid x_i \in \mathbb{Z}, i = 1, 2, \ldots, \omega \right\}.$$

We call s the dimension of Λ and ω its rank. The row vectors $b_1, b_2, \ldots, b_\omega$ are a basis of Λ, and we denote the basis as a matrix, termed the basis matrix of Λ:

$$\mathcal{B} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_\omega \end{pmatrix} \in \mathbb{R}^{\omega \times s}.$$

The determinant of Λ is defined as $\det(\Lambda) = \sqrt{\det(\mathcal{B}\mathcal{B}^{\mathrm{T}})}$, which is independent of the choice of basis and only determined by Λ. This paper only considers lattices for the case of $\omega = s$. Thus, $\mathcal{B}$ is a square matrix and $\det(\Lambda) = |\det \mathcal{B}|$.

In 1982, Lenstra et al. [26] proposed the well-known LLL algorithm for lattice basis reduction. It allows one to find a short vector in a lattice in polynomial time. The proof of the following fact can be found in [27]. The norm of a vector $v_i = (v_{i1}, v_{i2}, \ldots, v_{is})$ is defined as $\|v_i\| = \sqrt{v_{i1}^2 + v_{i2}^2 + \cdots + v_{is}^2}$.

Proposition 1 (LLL). Let s be the dimension (and the rank) of Λ. Given a basis (square) matrix $\mathcal{B}$ of Λ, the LLL algorithm outputs an LLL-reduced basis $v_1, v_2, \ldots, v_s$ satisfying

$$\|v_1\|, \|v_2\|, \ldots, \|v_i\| \leqslant 2^{\frac{s(s-1)}{4(s-i+1)}} \det(\Lambda)^{\frac{1}{s-i+1}}, \quad 1 \leqslant i \leqslant s$$

in polynomial time in s and in the bit sizes of the entries of the basis matrix $\mathcal{B}$.

Next, we introduce the following useful lemma due to Howgrave-Graham [19]. The norm of a polynomial $h(x_1,\ldots,x_n)=\sum a_{t_1,\ldots,t_n}x_1^{t_1}\cdots x_n^{t_n}$ is defined as $\|h(x_1,\ldots,x_n)\|=\sqrt{\sum|a_{t_1,\ldots,t_n}|^2}$.

Lemma 1 (Howgrave-Graham). Let $h(x_1,\ldots,x_v)\in\mathbb{Z}[x_1,\ldots,x_v]$ be a polynomial that consists of at most s monomials. Suppose that there exists $(x_1^{(0)},\ldots,x_v^{(0)})\in\mathbb{Z}^v$ satisfying

$$h(x_1^{(0)},\ldots,x_v^{(0)})\equiv 0 \mod V,\quad |x_1^{(0)}|<X_1,\ldots,|x_v^{(0)}|<X_v,$$

and we have

$$\|h(X_1x_1,\ldots,X_vx_v)\|<V/\sqrt{s}.$$

Then $h(x_1^{(0)},\ldots,x_v^{(0)})=0$ holds over the integers.

Combining Proposition 1 with Lemma 1, one can analyse the bounds for the small roots. The method is called Coppersmith's method, which has been widely adopted by researchers for the lattice-based cryptanalysis of RSA. Taking the case of trivariate modular polynomial equations as an example, we summarize Coppersmith's method as follows.

Finding small roots of a trivariate modular polynomial equation can be described as finding each root $(x_1^{(0)},x_2^{(0)},x_3^{(0)})\in\mathbb{Z}^3$ of

$$h_0(x_1,x_2,x_3)\equiv 0 \mod W,\quad |x_1|<X_1,\quad |x_2|<X_2,\quad |x_3|<X_3.$$

Let τ be a positive integer, and find a subset Λ^* of $\mathbb{Z}[x_1,x_2,x_3]$ satisfying

$$h(x_1^{(0)},x_2^{(0)},x_3^{(0)})\equiv 0 \mod W^\tau,\quad \forall\ h(x_1,x_2,x_3)\in\Lambda^*.$$

Given an order of some monomials of $\mathbb{Z}[x_1,x_2,x_3]$, there is a one-to-one correspondence between a polynomial $h(x_1,x_2,x_3)$ in Λ^* and a vector in a subset Λ of $\mathbb{R}^s$ for some s, and the components of the vector are coefficients of $h(X_1x_1,X_2x_2,X_3x_3)$ in the order of the corresponding monomials. For example, if $s=4$ and the monomial order "$\prec$" is defined such that $1\prec x_1\prec x_1x_2\prec x_1x_3$, we know that the polynomial $2+x_1x_2+6x_1x_3$ corresponds to the vector $(2,0,X_1X_2,6X_1X_3)$. Λ is also required to be a lattice of dimension s. Combining Proposition 1 for $i=3$ with Lemma 1 for $v=3, V=W^\tau$, if

$$2^{\frac{s(s-1)}{4(s-2)}}\det(\Lambda)^{\frac{1}{s-2}}<W^\tau/\sqrt{s} \tag{1}$$

is satisfied, by running the LLL algorithm one can get three polynomials $g_1(x_1,x_2,x_3), g_2(x_1,x_2,x_3), g_3(x_1,x_2,x_3)$, all of which share the desired root $(x_1^{(0)},x_2^{(0)},x_3^{(0)})$ as a common root over the integers. Then Coppersmith's method needs the following assumption.

Assumption 1. The polynomials obtained by our lattice-based method are algebraically independent, and the common roots of these polynomials can be efficiently computed using techniques such as the calculation of the resultants or finding a Gröbner basis.

Since Assumption 1 is heuristic, we need to perform experiments to examine it in our attacks, which is done in Section 4. In our experiments, we choose to extract the desired root $(x_1^{(0)},x_2^{(0)},x_3^{(0)})$ by computing the resultants of $g_1(x_1,x_2,x_3), g_2(x_1,x_2,x_3), g_3(x_1,x_2,x_3)$. One can see Section 4 for these details.

Notice that inequality (1) is equivalent to $2^{\frac{s(s-1)}{4}}s^{\frac{s-2}{2}}\det(\Lambda)<W^{\tau(s-2)}$, and researchers often ignore terms that do not depend on W. Thus, we obtain

$$\det(\Lambda)<W^{\tau(s-2)}. \tag{2}$$

According to the analysis above, under Assumption 1, using Coppersmith's method to find the small roots of trivariate modular polynomial equations just requires the condition inequality (2).

3 Our new analysis

In this section, we present our new analysis for the IFP with shared middle bits. Similar to [21, 22, 24], our approach is based on Coppersmith's method for finding the small roots of multivariate modular polynomial equations. We optimize the lattice construction for Coppersmith's method and then prove Theorem 1.

Recall that for two n-bit RSA moduli $N_1 = p_1q_1$ and $N_2 = p_2q_2$, q_1, q_2 are both αn-bit prime integers, and p_1, p_2 share tn bits at positions from t_1n to $t_2n = (t_1 + t)n$. Thus, we write

$$p_1 = p_{1_2}2^{t_2n} + p2^{t_1n} + p_{1_0}, \quad p_2 = p_{2_2}2^{t_2n} + p2^{t_1n} + p_{2_0},$$

and it is obtained that

$$p_1 - p_2 = (p_{1_2} - p_{2_2})2^{t_2n} + (p_{1_0} - p_{2_0}).$$

Together with

$$N_2 + (p_1 - p_2)q_2 \equiv 0 \mod p_1,$$

we obtain

$$N_2 + [(p_{1_2} - p_{2_2})2^{t_2n} + (p_{1_0} - p_{2_0})]q_2 \equiv 0 \mod p_1,$$

or

$$N_2 + 2^{t_2n}(p_{1_2} - p_{2_2})q_2 + (p_{1_0} - p_{2_0})q_2 \equiv 0 \mod p_1. \tag{3}$$

Set $N := 2^n$, then we have $N_1 \approx N_2 \approx N$, and roughly

$$|p_{1_2} - p_{2_2}| < 2^{(1-\alpha-t_2)n} = N^{1-\alpha-t_2}, \quad |p_{1_0} - p_{2_0}| < 2^{t_1n} = N^{t_1}, \quad |q_2| < 2^{\alpha n} = N^{\alpha}. \tag{4}$$

From (3) and inequality (4), we know that $(x_1^{(0)}, x_2^{(0)}, y^{(0)}) := (p_{1_2} - p_{2_2}, p_{1_0} - p_{2_0}, q_2)$ is a small root of the following modular equation:

$$f(x_1, x_2, y) := N_2 + 2^{t_2n}x_1y + x_2y \equiv 0 \mod p_1, \tag{5}$$

$$|x_1| < X_1 := N^{1-\alpha-t_2}, \quad |x_2| < X_2 := N^{t_1}, \quad |y| < Y := N^{\alpha}.$$

If we recover $(x_1^{(0)}, x_2^{(0)}, y^{(0)}) = (p_{1_2} - p_{2_2}, p_{1_0} - p_{2_0}, q_2)$, N_2 is factored, and we can also easily factor N_1 from $p_1 = \frac{N_2}{q_2} + (p_{1_2} - p_{2_2})2^{t_2n} + (p_{1_0} - p_{2_0}) = \frac{N_2}{y^{(0)}} + x_1^{(0)} \cdot 2^{t_2n} + x_2^{(0)}$.

Just as [24], we introduce a new variable z for p_2, and multiply the polynomial $f(x_1, x_2, y)$ by a power z^i for some i, which is optimized later. Set $z^{(0)} := p_2$, and roughly we have $|z^{(0)}| < Z := N^{1-\alpha}$. Since $N_2 = q_2p_2$, it allows us to replace each occurrence of the monomial yz by N_2, which optimizes our lattice construction. Let positive integers m, τ, together with i, be undetermined parameters. Then for two non-negative integers j, k, define

$$g_{j,k}(x_1, x_2, y, z) := z^i(x_1y)^j[f(x_1, x_2, y)]^kN_1^{\max\{\tau-k,0\}}, \quad \text{where } j + k = 0, 1, 2, \ldots, m.$$

Recall that $f(x_1^{(0)}, x_2^{(0)}, y^{(0)}) \equiv 0 \mod p_1$ and $N_1 \equiv 0 \mod p_1$. Thus, every polynomial $g_{j,k}(x_1, x_2, y, z)$ has the root $(x_1^{(0)}, x_2^{(0)}, y^{(0)}, z^{(0)}) = (p_{1_2} - p_{2_2}, p_{1_0} - p_{2_0}, q_2, p_2)$ modulo p_1^{τ}.

For every polynomial $g_{j,k}(x_1, x_2, y, z)$, we replace each occurrence of the monomial yz by N_2 as mentioned before. Therefore, for some j_1, j_2, the monomial $x_1^{j_1}x_2^{j_2}y^{j_1+j_2}z^i$ $(j_1 + j_2 \leqslant i)$ with coefficient a_{j_1,j_2} is transformed into a monomial $x_1^{j_1}x_2^{j_2}z^{i-(j_1+j_2)}$ with coefficient $a_{j_1,j_2}N_2^{j_1+j_2}$, and the monomial $x_1^{j_1}x_2^{j_2}y^{j_1+j_2}z^i$ $(j_1 + j_2 > i)$ with coefficient a_{j_1,j_2} is transformed into a monomial $x_1^{j_1}x_2^{j_2}y^{(j_1+j_2)-i}$ with coefficient $a_{j_1,j_2}N_2^i$.

Let E be the inverse of N_2 modulo N_1^{τ}, namely, $EN_2 \equiv 1 \mod N_1^{\tau}$. If E does not exist, one can easily factor N_1, N_2 by computing $\gcd(N_1, N_2)$. Next, we define

$$h_{j,k}(x_1, x_2, y, z) := E^{\min\{j+k,i\}}g_{j,k}(x_1, x_2, y, z), \quad \text{where } j + k = 0, 1, 2, \ldots, m.$$

Table 2 The basis matrix $\mathcal{B}$ when $m = 3, \tau = 2, i = 2$[a)]

$h_{j,k}$	z^2	x_1z	x_2z	x_1^2	x_1x_2	x_2^2	x_1^3y	$x_1^2x_2y$	$x_1x_2^2y$	x_2^3y
$h_{0,0}$	$N_1^2Z^2$									
$h_{1,0}$		$N_1^2X_1Z$								
$h_{0,1}$	$*$	$*$	N_1X_2Z							
$h_{2,0}$				$N_1^2X_1^2$						
$h_{1,1}$		$*$		$*$	$N_1X_1X_2$					
$h_{0,2}$	$*$	$*$	$*$	$*$	$*$	X_2^2				
$h_{3,0}$							$N_1^2X_1^3Y$			
$h_{2,1}$				$*$			$*$	$N_1X_1^2X_2Y$		
$h_{1,2}$		$*$		$*$	$*$		$*$	$*$	$X_1X_2^2Y$	
$h_{0,3}$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	$*$	X_2^3Y

a) Non-zero off-diagonal entries are denoted by *, and blank elements mean zero entries.

Here $(x_1^{(0)}, x_2^{(0)}, y^{(0)}, z^{(0)})$ is also the root of every $h_{j,k}(x_1, x_2, y, z)$ modulo p_1^τ. Notice that we are only concerned with $h_{j,k}(x_1, x_2, y, z)$ modulo p_1^τ. From $p_1|N_1$ one obtains $EN_2 \equiv 1 \mod p_1^\tau$. Thus, we can replace each occurrence of EN_2 by 1 in $h_{j,k}(x_1, x_2, y, z)$. If $j + k \leqslant i$, there is a monomial $x_1^jx_2^kz^{i-(j+k)}$ with coefficient $N_1^{\max\{\tau-k,0\}}(EN_2)^{j+k}$ in $h_{j,k}(x_1, x_2, y, z)$. If $j + k > i$, there is a monomial $x_1^jx_2^ky^{(j+k)-i}$ with coefficient $N_1^{\max\{\tau-k,0\}}(EN_2)^i$ in $h_{j,k}(x_1, x_2, y, z)$. By substituting $EN_2 \equiv 1 \mod p_1^\tau$, we are able to minimize the coefficient of the monomial $x_1^jx_2^kz^{i-(j+k)}$ or the monomial $x_1^jx_2^ky^{(j+k)-i}$, which is related to the determinant of the lattice Λ proposed in this paper.

Before constructing the basis matrix $\mathcal{B}$ of our lattice Λ, we need to define the monomial order "$\prec$". For convenience, here we use $x_1^{j_1}x_2^{j_2}y^{j_1+j_2}z^i$ to denote $x_1^{j_1}x_2^{j_2}z^{i-(j_1+j_2)}$ if $j_1+j_2 \leqslant i$ and denote $x_1^{j_1}x_2^{j_2}y^{(j_1+j_2)-i}$ if $j_1+j_2 > i$. Then, "$\prec$" is defined such that $x_1^{j_1}x_2^{j_2}y^{j_1+j_2}z^i \prec x_1^{k_1}x_2^{k_2}y^{k_1+k_2}z^i$ if and only if $j_1+j_2 < k_1+k_2$ or $j_1+j_2 = k_1+k_2, j_2 < k_2$. Now, the coefficient vectors of $h_{j,k}(X_1x_1, X_2x_2, Yy, Zz)$ $(j+k = 0, 1, 2, \ldots, m)$ are determined according to "$\prec$", and thus we know the basis matrix $\mathcal{B}$ that consists of these coefficient vectors.

A simple example of $\mathcal{B}$ when $m = 3, \tau = 2, i = 2$ is shown in Table 2, where other non-zero off-diagonal entries are denoted by *, and blank elements mean zero entries. Here we use the polynomial $h_{2,1}(x_1, x_2, y, z)$ in Table 2 to illustrate the above substitution and the one-to-one correspondence between a polynomial and a row vector. According to the definition, we know

$$\begin{aligned} h_{2,1}(x_1, x_2, y, z) &= E^{\min\{2+1,2\}}z^2(x_1y)^2(N_2 + 2^{t_2n}x_1y + x_2y)N_1^{\max\{2-1,0\}} \\ &= N_1E^2(N_2x_1^2y^2z^2 + 2^{t_2n}x_1^3y^3z^2 + x_1^2x_2y^3z^2). \end{aligned}$$

Replace each occurrence of the monomial yz by N_2, and get

$$\begin{aligned} h_{2,1}(x_1, x_2, y, z) &= N_1E^2(N_2x_1^2 \cdot N_2^2 + 2^{t_2n}x_1^3y \cdot N_2^2 + x_1^2x_2y \cdot N_2^2) \\ &= N_1N_2(EN_2)^2x_1^2 + 2^{t_2n}N_1(EN_2)^2x_1^3y + N_1(EN_2)^2x_1^2x_2y. \end{aligned}$$

Next, replace each occurrence of EN_2 by 1, and obtain

$$h_{2,1}(x_1, x_2, y, z) = N_1N_2x_1^2 + 2^{t_2n}N_1x_1^3y + N_1x_1^2x_2y.$$

Thus, $h_{2,1}(X_1x_1, X_2x_2, Yy, Zz) = N_1N_2X_1^2x_1^2 + 2^{t_2n}N_1X_1^3Yx_1^3y + N_1X_1^2X_2Yx_1^2x_2y$. According to the definition of the monomial order "$\prec$", the corresponding coefficient vector is

$$(0,\ 0,\ 0,\ N_1N_2X_1^2,\ 0,\ 0,\ 2^{t_2n}N_1X_1^3Y,\ N_1X_1^2X_2Y,\ 0,\ 0).$$

Define $\sigma := \frac{i}{m}$, $\xi := \frac{\tau}{m}$, and we only consider the case of $0 < \sigma < 1$, $0 < \xi < 1$. Now one gets $i = \sigma m$, $\tau = \xi m$, which is used in the following calculation. As seen in Table 2, it is easy to make $\mathcal{B}$ a lower triangular square matrix; thus, we can easily compute the value of $\det(\Lambda) = |\det \mathcal{B}|$. Let

$\det(\Lambda)=N_1^{s_{N_1}}X_1^{s_{X_1}}X_2^{s_{X_2}}Y^{s_Y}Z^{s_Z}$, and then we have

$$\begin{aligned}
s_{N_1}&=\sum_{k=0}^{\tau}(\tau-k)(m-k+1)=-\frac{1}{6}\tau^3+\frac{1}{2}\tau^2m+o(m^3)=\Big(-\frac{1}{6}\xi^3+\frac{1}{2}\xi^2\Big)m^3+o(m^3),\\
s_{X_1}&=s_{X_2}=\frac{1}{2}\sum_{j=1}^{m}j(j+1)=\frac{1}{6}m^3+o(m^3),\\
s_Y&=\sum_{j=i+1}^{m}(j-i)(j+1)=\frac{1}{6}i^3-\frac{1}{2}im^2+\frac{1}{3}m^3+o(m^3)=\Big(\frac{1}{6}\sigma^3-\frac{1}{2}\sigma+\frac{1}{3}\Big)m^3+o(m^3),\\
s_Z&=\sum_{j=0}^{i}(i-j)(j+1)=\frac{1}{6}i^3+o(m^3)=\frac{1}{6}\sigma^3m^3+o(m^3).
\end{aligned}$$

Let s denote the dimension of Λ, and we can calculate

$$s=\sum_{j=0}^{m}(j+1)=\frac{1}{2}m^2+o(m^2).$$

As described in Section 2, under Assumption 1, using Coppersmith's method to find the small common root $(x_1^{(0)},x_2^{(0)},y^{(0)})$ simply requires the condition inequality (2), namely, $\det(\Lambda)<W^{\tau(s-2)}$, where $W=p_1$. Both N_1 and N_2 can be successfully factored after we recover $(x_1^{(0)},x_2^{(0)},y^{(0)})$. Here we note that there are four variables x_1,x_2,y,z with the relation $yz=N_2$, which is essentially the case of three variables x_1,x_2,y. One can see Section 4 for details.

From condition $\det(\Lambda)<{p_1}^{\tau(s-2)}$ and calculation of $\det(\Lambda)$ and s, we have

$$N_1^{-\frac{1}{6}\xi^3+\frac{1}{2}\xi^2+\frac{o(m^3)}{m^3}}(X_1X_2)^{\frac{1}{6}+\frac{o(m^3)}{m^3}}Y^{\frac{1}{6}\sigma^3-\frac{1}{2}\sigma+\frac{1}{3}+\frac{o(m^3)}{m^3}}Z^{\frac{1}{6}\sigma^3+\frac{o(m^3)}{m^3}}<{p_1}^{\frac{1}{2}\xi+\frac{o(m^3)}{m^3}}.$$

Substitute $N_1\approx N$, $X_1=N^{1-\alpha-t_2}$, $X_2=N^{t_1}$, $Y=N^{\alpha}$, $Z=N^{1-\alpha}$, $p_1\approx N^{1-\alpha}$, and then it is obtained that

$$N^{-\frac{1}{6}\xi^3+\frac{1}{2}\xi^2+\frac{o(m^3)}{m^3}}N^{\frac{1}{6}[1-\alpha-(t_2-t_1)]+\frac{o(m^3)}{m^3}}N^{(\frac{1}{6}\sigma^3-\frac{1}{2}\sigma+\frac{1}{3})\alpha+\frac{o(m^3)}{m^3}}N^{\frac{1}{6}\sigma^3(1-\alpha)+\frac{o(m^3)}{m^3}}<N^{\frac{1}{2}\xi(1-\alpha)+\frac{o(m^3)}{m^3}}.$$

Take $m\to\infty$ and omit term $\frac{o(m^3)}{m^3}$, then we have

$$-\frac{1}{6}\xi^3+\frac{1}{2}\xi^2+\frac{1}{6}(1-\alpha-t)+\left(\frac{1}{6}\sigma^3-\frac{1}{2}\sigma+\frac{1}{3}\right)\alpha+\frac{1}{6}\sigma^3(1-\alpha)<\frac{1}{2}\xi(1-\alpha),$$

which reduces to

$$\sigma^3-3\alpha\sigma+2\alpha+(1-\alpha-t)<\xi^3-3\xi^2+3(1-\alpha)\xi. \tag{6}$$

In order to minimize the left-hand side of inequality (6) and maximize the right-hand side of inequality (6), the optimized values of σ and ξ are given by $\sigma=\sqrt{\alpha}$, $\xi=1-\sqrt{\alpha}$. After substituting $\sigma=\sqrt{\alpha}$, $\xi=1-\sqrt{\alpha}$ in inequality (6), we acquire

$$-2\alpha^{\frac{3}{2}}+2\alpha+(1-\alpha-t)<2\alpha^{\frac{3}{2}}-3\alpha+1,$$

which finally ends up with

$$t>4\alpha-4\alpha^{\frac{3}{2}},$$

and thus Theorem 1 follows.

Theorem 1 ignores "ε" for the bound $t\geqslant 4\alpha-4\alpha^{\frac{3}{2}}+\varepsilon$. From the above proof, we know that "ε" is dependent on N (or the bit length n) and the parameter m (or the parameter s, the dimension of Λ). The value of "ε" can be made arbitrarily small by ensuring that n and s are sufficiently large.

Technique for introducing a new variable *z*. Note that a new variable z is introduced for p_2. In order to optimize our lattice construction, we multiply the polynomial $f(x_1,x_2,y)$ by a power z^i, and then replace each occurrence of the monomial yz by N_2 later. After optimizing the value of i, we obtain the final result $t > 4\alpha - 4\alpha^{\frac{3}{2}}$.

If this technique is not used, we can only consider $f(x_1, x_2, y) = N_2 + 2^{t_2 n}x_1 y + x_2 y \equiv 0 \mod p_1$ (or $f^*(u_1, u_2) := N_2 + 2^{t_2 n}u_1 + u_2 \equiv 0 \mod p_1$ for simplicity, where $u_1 := x_1 y, u_2 := x_2 y$). The situation is equivalent to multiplying $f(x_1, x_2, y)$ by z^i and then taking $i = 0$. Since $\sigma = \frac{i}{m}$, we know $i = 0$ implies $\sigma = 0$. After substituting $\sigma = 0,\ \xi = 1 - \sqrt{\alpha}$ (instead of $\sigma = \sqrt{\alpha},\ \xi = 1 - \sqrt{\alpha}$) in inequality (6), finally we can only obtain the result $t > 4\alpha - 2\alpha^{\frac{3}{2}}$. Although this result is better than $t > 4\alpha - 3\alpha^2$ (the result of [25]) for $0 < \alpha < \frac{4}{9}$, it is not as good as $t > 4\alpha - 4\alpha^{\frac{3}{2}}$ (the result of Theorem 1).

From the analysis above, one can see how the technique contributes to an optimal result in our paper. Finally, we note that this technique was not only used by [24] for the IFP with shared LSBs or shared MSBs, but also used earlier in 2006 by [28] for attacks on RSA with small secret CRT-exponents.

Extension to more RSA moduli. Similar to [25], our new analysis can also be generalized from two RSA moduli to an arbitrary number of RSA moduli. The key sketch can be described as follows.

Suppose there are k ($\geqslant 2$) moduli $N_j = p_j q_j$ for $j = 1, 2, \ldots, k$, where all the N_j have n bits, all the q_j have αn bits, and all the p_j share tn bits at the positions from $t_1 n$ to $t_2 n = (t_1 + t)n$. Thus, we write

$$p_j = p_{j_2} 2^{t_2 n} + p 2^{t_1 n} + p_{j_0}, \quad j = 1, 2, \ldots, k,$$

and it is obtained that

$$p_1 - p_j = (p_{1_2} - p_{j_2})2^{t_2 n} + (p_{1_0} - p_{j_0}), \quad j = 2, \ldots, k.$$

Together with $N_j + (p_1 - p_j)q_j \equiv 0 \mod p_1$ for $j = 2, \ldots, k$, we get

$$N_j + 2^{t_2 n}(p_{1_2} - p_{j_2})q_j + (p_{1_0} - p_{j_0})q_j \equiv 0 \mod p_1, \quad j = 2, \ldots, k.$$

Namely, we obtain the following modular equations:

$$f_j(x_{j_2}, x_{j_0}, y_j) := N_j + 2^{t_2 n}x_{j_2} y_j + x_{j_0} y_j \equiv 0 \mod p_1, \quad j = 2, \ldots, k,$$

with the small roots $(x_{j_2}^{(0)}, x_{j_0}^{(0)}, y_j^{(0)}) := (p_{1_2} - p_{j_2}, p_{1_0} - p_{j_0}, q_j)$ for $j = 2, \ldots, k$.

Again we introduce new variables z_j for p_j, where $j = 2, \ldots, k$. Then we construct the basis matrix of the lattice used for attack according to the following polynomials:

$$(z_2 z_3 \cdots z_k)^i (x_{2_2} y_2)^{l_2^*} f_2^{l_2} (x_{3_2} y_3)^{l_3^*} f_3^{l_3} \cdots (x_{k_2} y_k)^{l_k^*} f_k^{l_k} N_1^{\max\{\tau - l_2 - l_3 - \cdots - l_k, 0\}},$$

where $(l_2^* + l_2) + (l_3^* + l_3) + \cdots + (l_k^* + l_k) = 0, 1, 2, \ldots, m$, and the positive integers m, τ, i are three undetermined parameters just like before. Since $N_j = q_j p_j$, it allows us to replace each occurrence of the monomial $y_j z_j$ by N_j for $j = 2, \ldots, k$, which again optimizes our lattice construction.

For this situation, there are $3(k-1)$ variables. Thus, we need to generalize the condition for attack from $\det(\Lambda) < {p_1}^{\tau(s-2)}$ to $\det(\Lambda) < {p_1}^{\tau(s-3(k-1)+1)}$. The computation for the dimension s and the determinant $\det(\Lambda)$ of the new lattice Λ may be very intricate and complicated, which also involves the optimization of $\sigma := \frac{i}{m},\ \xi := \frac{\tau}{m}$. Take $m \to \infty$ and substitute the values of s and $\det(\Lambda)$ into $\det(\Lambda) < {p_1}^{\tau(s-3(k-1)+1)}$, and one can obtain the generalized result of the IFP with shared middle bits for $k (\geqslant 2)$ RSA moduli.

4 Experiments

Similar to other cryptanalyses of RSA based on Coppersmith's method, our approach is heuristic due to Assumption 1 as stated before. In order to show the correctness of our results, we have implemented several experiments in SAGE 5.0 over Linux Fedora 16 on a laptop with 2.80 GHz Intel Core2 CPU and 4 GB RAM.

We choose to use the calculation of the resultants to examine Assumption 1 in our experiments. According to our approach, if the bound $t > 4\alpha - 4\alpha^{\frac{3}{2}}$ is satisfied, after running the LLL algorithm, we

Table 3 Some experimental results for Theorem 1

n	α	t	t_1	t_2	m	τ	i	$\dim(\Lambda)$	$\log_2(\det(\Lambda))$	Time for LLL algorithm (s)
1000	0.250	0.650	0.050	0.700	6	3	3	28	6.042×10^4	1.826
1500	0.230	0.670	0.020	0.690	6	4	2	28	1.234×10^5	14.13
2000	0.270	0.630	0.075	0.705	6	3	3	28	1.221×10^5	9.975
1000	0.310	0.660	0.010	0.670	8	4	4	45	1.217×10^5	22.89
1500	0.290	0.640	0.020	0.660	9	4	5	55	2.284×10^5	197.5
2000	0.300	0.650	0.030	0.680	7	3	4	36	1.506×10^5	26.13

can finally obtain three polynomials $\widetilde{g}_1(x_1, x_2, y, z), \widetilde{g}_2(x_1, x_2, y, z), \widetilde{g}_3(x_1, x_2, y, z)$, all of which share the desired root $(x_1^{(0)}, x_2^{(0)}, y^{(0)}, z^{(0)})$ as a common root over the integers.

The case of four variables x_1, x_2, y, z with the relation $yz = N_2$ is essentially the case of three variables x_1, x_2, y. This is why we need only three polynomials $\widetilde{g}_1, \widetilde{g}_2, \widetilde{g}_3$. The variable z can be eliminated by substituting $z = \frac{N_2}{y}$. Namely, for $j = 1, 2, 3$ we set $g_j(x_1, x_2, y) := y^i \widetilde{g}_j(x_1, x_2, y, \frac{N_2}{y})$, where $i = \sigma m$ is a parameter defined and optimized before. Now we have three polynomials $g_1(x_1, x_2, y), g_2(x_1, x_2, y), g_3(x_1, x_2, y)$ with the desired root $(x_1^{(0)}, x_2^{(0)}, y^{(0)})$ as a common root over the integers.

By computing resultants we can eliminate x_1 and x_2. Namely, we obtain $g_{12}(x_2, y) = \mathrm{Res}_{x_1}(g_1, g_2)$, $g_{13}(x_2, y) = \mathrm{Res}_{x_1}(g_1, g_3)$ and then $g_{12,13}(y) = \mathrm{Res}_{x_2}(g_{12}, g_{13})$. If Assumption 1 holds, $g_{12,13}(y) \not\equiv 0$. Thus, one can use any standard root-finding algorithm to recover $y^{(0)} \in \mathbb{Z}$ from $g_{12,13}(y)$. Similarly, $x_2^{(0)} \in \mathbb{Z}$ can be computed from $g_{12}(x_2, y^{(0)})$ or $g_{13}(x_2, y^{(0)})$, and $x_1^{(0)} \in \mathbb{Z}$ is also obtained from $g_1(x_1, x_2^{(0)}, y^{(0)})$ or $g_2(x_1, x_2^{(0)}, y^{(0)})$ or $g_3(x_1, x_2^{(0)}, y^{(0)})$.

In Table 3, we show some experimental results for Theorem 1. Assumption 1 holds for these experimental results, and the desired root $(x_1^{(0)}, x_2^{(0)}, y^{(0)})$ is successfully acquired.

Assumption 1 may fail for other experimental results on a few occasions. In this case, we give another method to factor N_1, N_2. Still suppose that after running the LLL algorithm we obtain $\widetilde{g}_j(x_1, x_2, y, z)$, $j = 1, 2, 3$, and then set $g_j(x_1, x_2, y) := y^i \widetilde{g}_j(x_1, x_2, y, \frac{N_2}{y})$, $j = 1, 2, 3$. According to our lattice construction, it can be proved that every monomial (neglect the corresponding coefficient) of $g_j(x_1, x_2, y)$ must have the form of $x_1^{j_1} x_2^{j_2} y^{j_1+j_2}$. Therefore, for $j = 1, 2, 3$ it is reasonable to define $g_j^*(u_1, u_2) := g_j(x_1, x_2, y)$ by substituting $u_1 := x_1 y, u_2 := x_2 y$. Consequently, the case of three variables x_1, x_2, y is changed into the case of two variables u_1, u_2. Then we can recover the small root $(u_1^{(0)}, u_2^{(0)}) := (x_1^{(0)} y^{(0)}, x_2^{(0)} y^{(0)})$ by computing the resultants of $g_1^*(u_1, u_2), g_2^*(u_1, u_2)$ and by running the standard root-finding algorithm just as before. Here we also need Assumption 1, which actually holds in all of our experiments. Next, since $p_1 q_2 = p_2 q_2 + (p_1 - p_2) q_2 = N_2 + (p_{1_2} - p_{2_2}) 2^{t_2 n} \cdot q_2 + (p_{1_0} - p_{2_0}) q_2 = N_2 + x_1^{(0)} y^{(0)} \cdot 2^{t_2 n} + x_2^{(0)} y^{(0)} = N_2 + u_1^{(0)} \cdot 2^{t_2 n} + u_2^{(0)}$, we are able to obtain the value of $p_1 q_2$ from the knowledge of $u_1^{(0)}, u_2^{(0)}$. Thus, after computing $\gcd(N_1, p_1 q_2)$ and $\gcd(N_2, p_1 q_2)$, finally we can also factor N_1, N_2 in this way.

5 Conclusion

In this paper, we revisit the implicit factorization problem for the case where the unknown prime factors of two RSA moduli share a certain number of middle bits. We present the best bound among all known attacks in this case. It is for the first time that Coppersmith's method is directly applied to the IFP with shared middle bits. Besides, we give almost optimal lattice construction for Coppersmith's method in our new analysis. The justification of our approach is also examined through experiments.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 11531002, 61572026), Basic Research Fund of National University of Defense Technology (Grant No. CJ 13-02-01), Open Foundation of State Key Laboratory of Cryptology, and Program for New Century Excellent Talents in University (NCET).

Conflict of interest The authors declare that they have no conflict of interest.

References

1 Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Commun ACM, 1978, 21: 120–126

2 Coppersmith D. Finding a small root of a univariate modular equation. In: Advances in Cryptology-EUROCRYPT 1996. Berlin-Heidelberg: Springer, 1996. 155–165

3 Coppersmith D. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. J Cryptol, 1997, 10: 233–260

4 Wiener M J. Cryptanalysis of short RSA secret exponents. IEEE Trans Inform Theory, 1990, 36: 553–558

5 Boneh D, Durfee G. Cryptanalysis of RSA with private key d less than $N^{0.292}$. In: Advances in Cryptology-EUROCRYPT 1999. Berlin-Heidelberg: Springer, 1999. 1–11

6 Boneh D, Durfee G, Frankel Y. An attack on RSA given a small fraction of the private key bits. In: Advances in Cryptology-ASIACRYPT 1998. Berlin-Heidelberg: Springer, 1998. 25–34

7 Blömer J, May A. New partial key exposure attacks on RSA. In: Advances in Cryptology-CRYPTO 2003. Berlin-Heidelberg: Springer, 2003. 27–43

8 Ernst M, Jochemsz E, May A, et al. Partial key exposure attacks on RSA up to full size exponents. In: Advances in Cryptology-EUROCRYPT 2005. Berlin-Heidelberg: Springer, 2005. 371–386

9 Aono Y. A new lattice construction for partial key exposure attack for RSA. In: Public Key Cryptography-PKC 2009. Berlin-Heidelberg: Springer, 2009. 34–53

10 Sarkar S, Gupta S S, Maitra S. Partial key exposure attack on RSA-improvements for limited lattice dimensions. In: Progress in Cryptology-INDOCRYPT 2010. Berlin-Heidelberg: Springer, 2010. 2–16

11 Sarkar S. Partial key exposure: generalized framework to attack RSA. In: Progress in Cryptology-INDOCRYPT 2011. Berlin-Heidelberg: Springer, 2011. 76–92

12 May A. Computing the RSA secret key is deterministic polynomial time equivalent to factoring. In: Advances in Cryptology-CRYPTO 2004. Berlin-Heidelberg: Springer, 2004. 213–219

13 Coron J S, May A. Deterministic polynomial-time equivalence of computing the RSA secret key and factoring. J Cryptol, 2007, 20: 39–50

14 Luo P, Zhou H J, Wang D S, et al. Cryptanalysis of RSA for a special case with $d > e$. Sci China Ser F-Inf Sci, 2009, 52: 609–616

15 Zheng M, Hu H, Wang Z. Generalized cryptanalysis of RSA with small public exponent. Sci China Inf Sci, 2016, 59: 032108

16 May A, Ritzenhofen M. Implicit factoring: on polynomial time factoring given only an implicit hint. In: Public Key Cryptography-PKC 2009. Berlin-Heidelberg: Springer, 2009. 1–14

17 Faugère J C, Marinier R, Renault G. Implicit factoring with shared most significant and middle bits. In: Public Key Cryptography-PKC 2010. Berlin-Heidelberg: Springer, 2010. 70–87

18 Coppersmith D. Finding a small root of a bivariate integer equation; factoring with high bits known. In: Advances in Cryptology-EUROCRYPT 1996. Berlin-Heidelberg: Springer, 1996. 178–189

19 Howgrave-Graham N. Finding small roots of univariate modular equations revisited. In: Darnell M, ed. Crytography and Coding. Berlin: Springer, 1997. 131–142

20 Coron J S. Finding small roots of bivariate integer polynomial equations revisited. In: Advances in Cryptology-EUROCRYPT 2004. Berlin-Heidelberg: Springer, 2004. 492–505

21 Sarkar S, Maitra S. Approximate integer common divisor problem relates to implicit factorization. IEEE Trans Inform Theory, 2011, 57: 4002–4013

22 Lu Y, Zhang R, Lin D. Improved bounds for the implicit factorization problem. Adv Math Commun, 2013, 7: 243–251

23 Peng L Q, Hu L, Xu J, et al. Further improvement of factoring RSA moduli with implicit hint. In: Progress in Cryptology-AFRICACRYPT 2014. Berlin: Springer, 2014. 165–177

24 Lu Y, Peng L Q, Zhang R, et al. Towards optimal bounds for implicit factorization problem. In: Selected Areas in Cryptography-SAC 2015. Berlin: Springer, 2015. 462–476

25 Peng L Q, Hu L, Lu Y, et al. Implicit factorization of RSA moduli revisited (short paper). In: Advances in Information and Computer Security. Berlin: Springer, 2015. 67–76

26 Lenstra A K, Lenstra H W, Lovász L. Factoring polynomials with rational coefficients. Math Ann, 1982, 261: 515–534

27 May A. New RSA vulnerabilities using lattice reduction methods. Dissertation for Ph.D. Degree. Paderborn: University of Paderborn, 2003

28 Bleichenbacher D, May A. New attacks on RSA with small secret CRT-exponents. In: Public Key Cryptography-PKC 2006. Berlin-Heidelberg: Springer, 2006. 1–13