Finding Small Solutions of the Equation $Bx - Ay = z$ and Its Applications to Cryptanalysis of the RSA Cryptosystem*

Shixiong Wang[1], Longjiang Qu[1,2], Chao Li[1,3], Shaojing Fu[2,3], and Hao Chen[4]

1 College of Liberal Arts and Sciences, National University of Defense Technology,
Changsha 410073, China
2 State Key Laboratory of Cryptology, Beijing 100878, China
3 College of Computer, National University of Defense Technology,
Changsha 410073, China
4 College of Information Science and Technology/Collage of Cyber Security,
Jinan University, Guangzhou 510632, China
wsx09@foxmail.com,ljqu_happy@hotmail.com,lichao_nudt@sina.com,
shaojing1984@163.com,haochen@jnu.edu.cn

Abstract. In this paper, we study the condition of finding small solutions $(x, y, z) = (x_0, y_0, z_0)$ of the equation $Bx - Ay = z$. The framework is derived from Wiener's small private exponent attack on RSA and May-Ritzenhofen's investigation about the implicit factorization problem, both of which can be generalized to solve the above equation. We show that these two methods, together with Coppersmith's method, are equivalent for solving $Bx - Ay = z$ in the general case. Then based on Coppersmith's method, we present two improvements for solving $Bx - Ay = z$ in some special cases. The first improvement pays attention to the case where either $\gcd(x_0, z_0, A)$ or $\gcd(y_0, z_0, B)$ is large enough. As the applications of this improvement, we propose some new cryptanalysis of RSA, such as new results about the generalized implicit factorization problem, attacks with known bits of the prime factor, and so on. The motivation of these applications comes from oracle based complexity of factorization problems. The second improvement assumes that the value of $C \equiv z_0 \pmod{x_0}$ is known. We present two attacks on RSA as its applications. One focuses on the case with known bits of the private exponent together with the prime factor, and the other considers the case with a small difference of the two prime factors. Our new attacks on RSA improve the previous corresponding results respectively, and the correctness of the approach is verified by experiments.

Keywords: RSA, Cryptanalysis, Lattice, Coppersmith's method
Mathematics Subject Classification: 11Y05(Primary); 94A60(Secondary)

* The work in this paper is supported by the National Natural Science Foundation of China (Grant Nos. 11531002, 61572026 and 61722213), and the Open Foundation of State Key Laboratory of Cryptology (Grant No. MMKFKT201617).

1 Introduction

Background Since RSA was proposed by Rivest, Shamir, and Adleman [33], much effort has been made to evaluate the security of this public key cryptosystem due to its wide variety of applications. For example, RSA is vulnerable in the case of either a small public exponent [5, 7] or a small private exponent [3, 40]. Some attacks were also presented when a portion of the private exponent is exposed [1, 2, 4, 11, 34, 35, 38, 39], or some bits of the prime factor are known [6, 7, 16]. From the work of [9, 24], it was proved that recovering the private exponent and factoring the modulus are determinately equivalent in polynomial time. In addition, there were also some investigations about the implicit factorization problem (IFP) [12, 20, 21, 25, 31, 32, 37], which aims to factor two (or more) RSA moduli if unknown prime factors of these moduli share a certain number of bits.

In this paper, we show that some of the above cryptanalysis of RSA can be summarized into one framework, namely, finding small solutions $(x, y, z) = (x_0, y_0, z_0)$ of the equation $Bx - Ay = z$, where A, B, x_0, y_0, z_0 are integers and Bx_0, Ay_0 have the same bit size. Without loss of generality, we assume $A, B, x_0, y_0, z_0 \in \mathbb{Z}^+$ and $\gcd(A, B) = 1,\ \gcd(x_0, y_0) = 1$.

In 1990, based on approximations using continued fractions, Wiener [40] presented the first small private exponent attack. For an RSA modulus $N = pq$, suppose p, q are of the same bit size, and the public exponent $e \approx N$. Wiener showed that one can factor $N = pq$ if the private exponent $d < N^{0.25}$. Since $e \cdot d \equiv 1 (\bmod\ \varphi(N))$, there exists a positive integer k such that $k\varphi(N) = ed - 1$, which is equivalent to $Nk - ed = k(p + q - 1) - 1$. After taking $B = N, x_0 = k, A = e, y_0 = d, z_0 = k(p + q - 1) - 1$, we obtain $Bx_0 - Ay_0 = z_0$. And the RSA modulus $N = pq$ can be factored according to the knowledge of x_0, y_0, z_0. Therefore, Wiener's attack is essentially finding the solution $(x, y, z) = (x_0, y_0, z_0)$ of the equation $Bx - Ay = z$ if x_0, y_0, z_0 are small enough. The best small private exponent attack was later given by Boneh and Durfee [3] in 1999, which showed that it is possible to factor $N = pq$ if $d < N^{0.292}$. And the proof was simplified by Herrmann and May [14] in 2010.

In 2009, May and Ritzenhofen [25] firstly introduced the IFP, namely, the implicit factorization problem. The motivation of this problem comes from oracle based complexity of factorization problems. Namely, it allows for an oracle that on input an RSA modulus $N_1 = p_1q_1$ outputs another different RSA modulus $N_2 = p_2q_2$ such that p_1, p_2 share some bits. Note that the oracle only gives an implicit information about p_1, p_2, since the actual values of the shared bits are not known. Denote "most significant bits" by MSBs, and "least significant bits" by LSBs. Let $N_1 = p_1q_1, N_2 = p_2q_2$ be n-bit RSA moduli, and p_1, p_2 be αn-bit prime integers. Then suppose that p_1, p_2 share tn LSBs. By finding the shortest non-zero vector in a two-dimensional lattice, May and Ritzenhofen [25] claimed that N_1, N_2 can be factored if $t > 2(1 - \alpha)$. Later in 2010, Faugère et al. [12] analyzed the case where αn-bit p_1, p_2 share tn MSBs, and they got the same bound $t > 2(1-\alpha)$. For simplicity, here we only consider the latter case, namely, we have $p_1 - p_2 = \widetilde{p},\ 0 < \widetilde{p} < 2^{\alpha n - tn}$, where $\widetilde{p} > 0$ is assumed without loss of generality. It is easy to obtain $N_1q_2 - N_2q_1 = \widetilde{p}q_1q_2$. Similarly, after taking

$B = N_1, x_0 = q_2, A = N_2, y_0 = q_1, z_0 = \widetilde{p} q_1 q_2$, one can also get $Bx_0 - Ay_0 = z_0$. And the knowledge of x_0, y_0, z_0 is sufficient to factor $N_1 = p_1 q_1, N_2 = p_2 q_2$. Therefore, the above IFP can be also regarded as finding the solution $(x, y, z) = (x_0, y_0, z_0)$ of the equation $Bx - Ay = z$ if x_0, y_0, z_0 are small enough. Since the methods in [25] and [12] are similar, in this paper we only consider the method in [25] for simplicity. Besides the work in [25] and [12], later the bounds for the cases of shared MSBs and shared LSBs were simultaneously improved several times [20, 21, 32, 37] by means of lattice-based methods. And the optimal bound was given by Lu et al. [20] in 2015, which claimed that N_1, N_2 can be factored if $t > 2\alpha(1 - \alpha)$.

Both Wiener's attack in [40] and May-Ritzenhofen's attack in [25], can be generalized to solve the equation $Bx - Ay = z$. Besides, there is another lattice-based method, called Coppersmith's method, which is widely adopted by researchers for cryptanalysis of RSA. 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 [5, 6] obtained results for the case of $v = 1$. Later the methods of [5] and [6] were reformulated by Howgrave-Graham[15] and Coron[8] respectively in simpler ways. The aforementioned two reformulations can also be extended to the case of $v \geqslant 2$. In general, the reformulations are used when we refer to Coppersmith's method.

Our Contributions In this paper, we are devoted to making improvements for solving the equation $Bx - Ay = z$ in some special cases, together with obtaining some new applications to cryptanalysis of RSA.

First of all, we present the condition for solving the equation $Bx - Ay = z$ in the general case as Result 1 in Section 2. We regard it as a known result, which can be obtained by generalizing either Wiener's small private exponent attack in [40] or May-Ritzenhofen's investigation about implicit factorization problem in [25]. Besides, Coppersmith's method can also be used to prove Result 1. As a conclusion, these three methods are equivalent for solving $Bx - Ay = z$ in the general case.

Moreover, Coppersmith's method is much powerful. It can not only obtain the same result in the general case, but also perform better than the two methods in [40] and [25] under some circumstance. The optimal bound for the IFP [20] and the best small private exponent attack [3], are both obtained according to Coppersmith's method. Based on Coppersmith's method, this paper then presents two improvements for solving the equation $Bx - Ay = z$ in some special cases.

(1) The first improvement considers the case where either $\gcd(x_0, z_0, A)$ or $\gcd(y_0, z_0, B)$ is large enough. It is stated as our Theorem 1 together with Theorem 2 in Section 4. Based on the first improvement, we present some applications to cryptanalysis of RSA as follows.

(1.1) In 2015, Nitaj and Ariffin [28] proposed a generalization of the IFP, which is also related to oracle based complexity of factorization problems. The generalization allows for an oracle that on input an RSA modulus $N_1 = p_1q_1$ outputs another different RSA modulus $N_2 = p_2q_2$ such that some unknown multiples a_1p_1 and a_2p_2 of the prime factors p_1 and p_2 share an amount of MSBs or LSBs. When $a_1 = a_2 = 1$, it is exactly the case of the IFP introduced by May and Ritzenhofen in [25]. Moreover, since $\gcd(p_1, p_2) = 1$, there must exist unknown $a_1^*, a_2^* \in \mathbb{Z}^+$ such that $a_1^*p_1 - a_2^*p_2 = 1$ (or $a_2^*p_2 - a_1^*p_1 = 1$). It implies that $a_1^*p_1$ and $a_2^*p_2$ share nearly all of their bits beginning from the most significant bit, which may apply for the generalized IFP. Applying our Theorem 1 to this generalized IFP for the case of shared MSBs, we can get a better result than [28]. While our Theorem 2 can be used to improve the attack in [28] for the case of shared LSBs.

(1.2) In 1996, Coppersmith [6] claimed that given $0.25n$ MSBs of p, one can factor n-bit $N = pq$ for $0.5n$-bit p and q. Later [7, 16] showed that N can also be factored if one knows $0.25n$ LSBs of p. The motivation of their attacks is exactly from the original oracle based complexity of factorization problems. As opposed to the (generalized) IFP where the oracle only gives an implicit information, their attacks [6, 7, 16] allow for an oracle that explicitly outputs the bits of the prime factor p. Considering implementations in practice, the bits of p may be obtained via side channel attacks. As an application of the theorems in Section 4, our new attacks improve the results in [6, 7, 16] if some greatest common divisor is large enough, and we also consider the case of unbalanced p, q.

(1.3) Similarly, according to Theorem 1, we can improve the result of partially approximate common divisor problem (PACDP) in [16] under some circumstance. And the same improvement also applies to the attacks in [29, 22], which focus on solving $ex + y \equiv 0 \pmod{p}$ to factor the RSA modulus $N = pq$.

(2) The second improvement assumes that the value of $C \equiv z_0 \pmod{x_0}$ is known. It is stated as our Theorem 3 in Section 5. And its applications to cryptanalysis of RSA are as follows.

(2.1) In 2005, Ernst et al. [11] proposed the attack on RSA when a portion of the private exponent d is exposed due to side channel attacks. Their result is the first one that works up to full size public or private exponent. Later in 2008, Sarkar and Maitra [36] improved the result of [11] by guessing a few MSBs of the prime factor p of the RSA modulus N. The total amount of bits of d, p to be known as presented in [36], is less than the number of bits of d to be known as reported in [11]. According to our Theorem 3, we present a new attack on RSA when some LSBs of d together with some MSBs of p are exposed. Similar to the motivation of [36], we may obtain only a few bits of p by exhaustive search to reduce the requirement of more bits of d to be known. Our attack combines the results in [30] and [39], and is better than Sarkar and Maitra [36] under some circumstance.

(2.2) In 2002, Weger [10] showed that choosing an RSA modulus with a small difference of its prime factors yields improvements on the small private exponent attacks of Wiener [40] and Boneh, Durfee [3]. Among the results presented in [10], one is obtained by generalizing the best small private exponent attack result $d < N^{0.292}$ [3]. Another proof of this result can also be found in [18].

As the application of our second improvement, our new attack needs a weaker condition, which implies that our result is better than that in [10, 18].

Organization The rest of this paper is organized as follows. In Section 2, the condition of solving the equation $Bx - Ay = z$ in the general case is given as a known result. Section 3 introduces Coppersmith's method for finding the small roots of modular polynomial equations, which is used to prove the results in this paper. In Section 4, we present our first improvement. And based on this improvement, we obtain many applications to cryptanalysis of RSA, such as new results about the generalized IFP, attacks with known bits of the prime factor, analysis of the PACDP, and so on. Section 5 is our second improvement. And in this section, we propose two attacks on RSA, which consider either the case with known bits of the private exponent together with the prime factor, or the case with a small difference of the two prime factors. In Section 6, we implement several experiments to examine the justification of our approach and thus verify the correctness of our results. Finally we conclude this paper in Section 7.

2 Known Result for Solving $Bx - Ay = z$

In this section, we show the condition of finding the small solutions $(x, y, z) = (x_0, y_0, z_0)$ of the equation $Bx - Ay = z$ in the general case. We present it as the following Result 1, and regard it as a known result. It can be obtained by generalizing either Wiener's attack in [40] or May-Ritzenhofen's attack in [25]. We briefly present the proof using Wiener's method [40] in Appendix A, and the proof using May-Ritzenhofen's method [25] in Appendix B. Section 3 will introduce Coppersmith's method by showing another proof of Result 1 as an example. As a conclusion, these three methods are equivalent for solving $Bx - Ay = z$ in the general case.

Result 1 *Suppose there exists unknown* $(x_0, y_0, z_0) \in (\mathbb{Z}^+)^3$ *satisfying*

$$Bx_0 - Ay_0 = z_0, \quad |x_0| < X,\ |y_0| < Y,\ |z_0| < Z,$$

where $\gcd(A, B) = 1,\ \gcd(x_0, y_0) = 1$. *Here* A, B, X, Y, Z *are known large positive integers, and for some known large positive integer* M *we have*

$$X = M^{\alpha_1},\ A = M^{\alpha_2},\ Y = M^{\beta_1},\ B = M^{\beta_2},\ Z = M^{\gamma}.$$

X, Y, Z *are usually selected to satisfy* $x_0 \approx X,\ y_0 \approx Y,\ z_0 \approx Z$, *thus we also assume* $\beta_2 + \alpha_1 \approx \alpha_2 + \beta_1$ *holds such that* $z_0 \ll Bx_0, Ay_0$. *Then one can find all such solutions* $(x, y, z) = (x_0, y_0, z_0)$ *of the equation* $Bx - Ay = z$ *in polynomial time if*

$$\gamma + \alpha_1 - \alpha_2\ (\approx \gamma + \beta_1 - \beta_2) < 0. \tag{1}$$

We note that there should be a term "ε" with $\varepsilon > 0$ in the left-hand side of Inequality (1). This term is negligible, since A, B, X, Y, Z, M are large positive integers and ε usually equals to a small value such as $\log_M 2$. Considering applications to cryptanalysis of RSA, these large positive integers may have the magnitudes such as that of an RSA modulus. We usually have known the rough bit sizes of desired $x_0, y_0, z_0 \in \mathbb{Z}^+$, thus we are able to select X, Y, Z satisfying $x_0 \approx X,\ y_0 \approx Y,\ z_0 \approx Z$. Besides, $\beta_2 + \alpha_1 \approx \alpha_2 + \beta_1$ implies that Bx_0 and Ay_0 have the same bit size. And one can check that Inequality (1) also implies $z_0 \ll Bx_0, Ay_0$. Finally, we assume $x_0, y_0, z_0 \in \mathbb{Z}^+$ just for simplicity, and the result also holds for the case of $x_0, y_0, z_0 \in \mathbb{Z}$. This is why we use $|x_0|, |y_0|, |z_0|$ instead of x_0, y_0, z_0 in Result 1. The same applies to our two improvements in Sections 4, 5.

3 Coppersmith's Method

In this section, we introduce Coppersmith's method for finding the small roots of modular polynomial equations, which will be used in the proofs of our two improvements in Sections 4, 5 in this paper. We will first give some preliminaries about lattice in Section 3.1, and then present Coppersmith's method by showing a new proof of Result 1 as an example in Section 3.2. The proof can be regarded as a preparation for our proofs of the two improvements in Sections 4, 5.

3.1 Preliminaries about Lattice

First of all, let us recall the definition of (integer) lattice.

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

$$\Lambda = \operatorname{span}_{\mathbb{Z}}(\boldsymbol{b_1}, \boldsymbol{b_2}, \cdots, \boldsymbol{b_\omega}) = \left\{ \sum_{i=1}^{\omega} x_i \boldsymbol{b_i} \;\middle|\; x_1, x_2, \cdots, x_\omega \in \mathbb{Z} \right\}.$$

We call s the dimension of Λ and ω its rank. Row vectors $\boldsymbol{b_1}, \boldsymbol{b_2}, \cdots, \boldsymbol{b_\omega}$ are a basis of Λ, and we denote the basis as a matrix, called the basis matrix of Λ:

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

The determinant of Λ is defined as $\det(\Lambda) = \sqrt{\det(\mathcal{B}\mathcal{B}^T)}$, which is independent of the choice of the basis and only determined by Λ. In this paper, we only consider 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. [19] proposed the famous LLL algorithm for lattice basis reduction, which allows one to find short vectors in polynomial time. The proof of the following lemma can be found in [23]. The norm of a vector $\boldsymbol{v_i} = (v_{i1}, v_{i2}, \cdots, v_{is})$ is defined as $\|\boldsymbol{v_i}\| = \sqrt{v_{i1}^2 + v_{i2}^2 + \cdots + v_{is}^2}$.

Lemma 1. ***(LLL)*** *Let s be the dimension (and the rank) of the lattice $\varLambda$. Given a basis (square) matrix $\mathcal{B}$ of $\varLambda$, the LLL algorithm outputs a LLL-reduced basis $\boldsymbol{v_1}, \boldsymbol{v_2}, \cdots, \boldsymbol{v_s}$ satisfying*

$$\|\boldsymbol{v_1}\|, \|\boldsymbol{v_2}\|, \cdots, \|\boldsymbol{v_i}\| \leqslant 2^{s(s-1)/4(s-i+1)} \det(\varLambda)^{1/(s-i+1)}, \quad 1 \leqslant i \leqslant s$$

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

Finally we introduce the following useful lemma due to Howgrave-Graham [15]. The norm of a polynomial $h(x_1, \cdots, x_n) = \sum a_{t_1,\cdots,t_n} x_1^{t_1} \cdots x_n^{t_n}$ is defined as $\|h(x_1, \cdots, x_n)\| = \sqrt{\sum |a_{t_1,\cdots,t_n}|^2}$.

Lemma 2. ***(Howgrave-Graham)*** *Let $h(x_1, \cdots, x_v) \in \mathbb{Z}[x_1, \cdots, x_v]$ be a polynomial that consists of at most s monomials. Suppose that there exists $(x_1^{(0)}, \cdots, x_v^{(0)}) \in \mathbb{Z}^v$ satisfying*

$$h(x_1^{(0)}, \cdots, x_v^{(0)}) \equiv 0 \pmod V, \quad |x_1^{(0)}| < X_1, \cdots, |x_v^{(0)}| < X_v,$$
$$\|h(X_1x_1, \cdots, X_vx_v)\| < V/\sqrt{s}.$$

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

3.2 Coppersmith's Method to Prove Result 1

Here we present Coppersmith's method by showing a new proof of Result 1, which is a preparation for our proofs of the two improvements in Sections 4, 5.

Recall the notations in Result 1. Define $f(x, z) := -Bx + z$, and from $Bx_0 - Ay_0 = z_0$ we obtain

$$f(x_0, z_0) \equiv 0 \pmod A.$$

Let m be a positive integer, and then define

$$g_k(x, z) := x^{m-k}[f(x, z)]^k A^{m-k}, \quad k = 0, 1, \cdots, m,$$
$$\varLambda^* := \left\{ \sum_{k=0}^{m} l_k g_k(x, z) \;\middle|\; l_0,\ l_1,\ \cdots,\ l_m \in \mathbb{Z} \right\}.$$

For two monomials $x^{m-k_1}z^{k_1}$ and $x^{m-k_2}z^{k_2}$, the monomial order "$\prec$" is defined such that $x^{m-k_1}z^{k_1} \prec x^{m-k_2}z^{k_2}$ if and only if $k_1 < k_2$. Then there is a one-to-one correspondence between a polynomial $g(x, z)$ in $\varLambda^*$ and a vector $\boldsymbol{g}$ in a subset $\varLambda$ of $\mathbb{Z}^{m+1}$, where the components of $\boldsymbol{g}$ are the coefficients of $g(Xx, Zz)$ in the order of "$\prec$". Denote the corresponding vector of $g_k(x, z)$ by $\boldsymbol{g_k}$, and we have

$$\varLambda = \left\{ \sum_{k=0}^{m} l_k \boldsymbol{g_k} \;\middle|\; l_0,\ l_1,\ \cdots,\ l_m \in \mathbb{Z} \right\}.$$

One can check that $\boldsymbol{g_0}, \boldsymbol{g_1}, \cdots, \boldsymbol{g_m}$ are linearly independent, thus $\varLambda$ is indeed a lattice, whose dimension is $s := m + 1$. And $\boldsymbol{g_0}, \boldsymbol{g_1}, \cdots, \boldsymbol{g_m}$ form a basis matrix $\mathcal{B}$ of the lattice $\varLambda$. For example, when $m = 2$, we have

$$g_0(x, z) = A^2x^2, \quad g_1(x, z) = -ABx^2 + Axz, \quad g_2(x, z) = B^2x^2 - 2Bxz + z^2.$$

Then one can obtain the basis vectors $\boldsymbol{g_0}, \boldsymbol{g_1}, \boldsymbol{g_2}$, which form the basis matrix

$$\mathcal{B} = \begin{pmatrix} A^2X^2 & 0 & 0 \\ -ABX^2 & AXZ & 0 \\ B^2X^2 & -2BXZ & Z^2 \end{pmatrix}.$$

From $f(x_0, z_0) \equiv 0 \pmod{A}$, we know $g_k(x_0, z_0) \equiv 0 \pmod{A^m}$, and thus $g(x_0, z_0) \equiv 0 \pmod{A^m}$ holds for any $g(x, z) \in \varLambda^*$. According to Lemma 2, if $\|g(Xx, Zz)\| < A^m/\sqrt{s}$ holds, we have $g(x_0, z_0) = 0$ holds over the integers.

Suppose such a polynomial $g(x, z) \in \varLambda^*$ with $g(x_0, z_0) = 0$ is obtained. Since $g(x, z)$ is homogeneous, let $\zeta := x/z$ and one gets $h(\zeta) := g(x, z)/z^m$ with $h(x_0/z_0) = 0$. Then x_0/z_0 can be easily found by extracting the rational roots of $h(\zeta)$ with classical methods (the discussions about extracting the small rational roots can be found on page 413 of Joux's book [17]). Let $u_0 := \gcd(x_0, z_0)$, $x_0' := x_0/u_0$, $z_0' := z_0/u_0$, and we have $u_0/y_0 = A/(Bx_0' - z_0')$. From the value of $x_0'/z_0' = x_0/z_0$, we know the values of x_0', z_0' since $\gcd(x_0', z_0') = 1$. Thus $Bx_0' - z_0'$ is known, and so is the value of $u_0/y_0 = A/(Bx_0' - z_0')$. Since $u_0 \mid x_0$ and $\gcd(x_0, y_0) = 1$, we know $\gcd(u_0, y_0) = 1$. Hence the values of u_0, y_0 are known, together with the known x_0', z_0', we obtain (x_0, y_0, z_0).

As a conclusion, in order to find (x_0, y_0, z_0), we only need to find a polynomial $g(x, z)$ in $\varLambda^*$ with the condition $\|g(Xx, Zz)\| < A^m/\sqrt{s}$. This is equivalent to finding a vector $\boldsymbol{g}$ in $\varLambda$ with the condition $\|\boldsymbol{g}\| < A^m/\sqrt{s}$. According to Lemma 1 (take $i = 1$), by running LLL algorithm one can find a vector $\boldsymbol{g}$ in $\varLambda$ with $\|\boldsymbol{g}\| \leqslant 2^{(s-1)/4} \det(\varLambda)^{1/s}$. From the above, to find (x_0, y_0, z_0), the following condition is sufficient:

$$2^{(s-1)/4} \det(\varLambda)^{1/s} < A^m/\sqrt{s}.$$

It is equivalent to $2^{s(s-1)/4} s^{s/2} \det(\varLambda) < (A^m)^s$. Note that researchers often ignore terms that do not depend on the large integer A. Thus, we obtain

$$\det(\varLambda) < (A^m)^s.$$

The basis matrix $\mathcal{B}$ of the lattice $\varLambda$ is a lower triangular square matrix, thus we can compute

$$\det(\varLambda) = \det \mathcal{B} = \textstyle\prod_{k=0}^{m} X^{m-k} Z^k A^{m-k} = (AXZ)^{m(m+1)/2}.$$

Substitute the value of $\det(\varLambda)$ and $s = m + 1$ in $\det(\varLambda) < (A^m)^s$, and we have

$$(AXZ)^{m(m+1)/2} < A^{m(m+1)} \quad \Leftrightarrow \quad XZ < A \quad \Leftrightarrow \quad \gamma + \alpha_1 < \alpha_2,$$

which completes the proof of Result 1.

4 First Improvement for Solving $Bx - Ay = z$

In this section, we present the first improvement for solving the equation $Bx - Ay = z$. The main result is stated as Theorem 1 together with Theorem 2 in Section 4.1. Then based on the first improvement, Section 4.2 shows some applications to cryptanalysis of RSA, such as the generalized IFP with shared MSBs or LSBs, attacks with known MSBs or LSBs of the prime factor, and so on. Similar to the previous results, the motivation of our applications in Section 4.2 comes from oracle based complexity of factorization problems.

4.1 Our Main Result

Our first improvement pays attention to $\gcd(x_0, z_0, A),\ \gcd(y_0, z_0, B)$. And it can improve Result 1 when either of these two great common divisors is large enough. We present it as follows.

Theorem 1. *Suppose there exists unknown* $(x_0, y_0, z_0) \in (\mathbb{Z}^+)^3$ *and unknown* $u_0, v_0 \in \mathbb{Z}^+$ *satisfying*

$$\begin{gathered} Bx_0 - Ay_0 = z_0, \quad |x_0| < X,\ |y_0| < Y,\ |z_0| < Z, \\ u_0 = \gcd(x_0, z_0, A),\ v_0 = \gcd(y_0, z_0, B), \quad |u_0| > U,\ |v_0| > V, \end{gathered}$$

where $\gcd(A, B) = 1,\ \gcd(x_0, y_0) = 1$. *Here* A, B, X, Y, Z, U, V *are known large positive integers, and for some known large positive integer* M *we have*

$$X = M^{\alpha_1},\ A = M^{\alpha_2},\ U = M^{\alpha},\ Y = M^{\beta_1},\ B = M^{\beta_2},\ V = M^{\beta},\ Z = M^{\gamma}.$$

X, Y, Z, U, V *are usually selected to satisfy* $x_0 \approx X,\ y_0 \approx Y,\ z_0 \approx Z,\ u_0 \approx U,\ v_0 \approx V$*, thus we also assume* $\beta_2 + \alpha_1 \approx \alpha_2 + \beta_1$ *holds such that* $z_0 \ll Bx_0, Ay_0$*. Then one can find all such solutions* $(x, y, z) = (x_0, y_0, z_0)$ *of the equation* $Bx - Ay = z$ *in polynomial time if*

$$\gamma + \alpha_1 - \alpha_2\ (\approx \gamma + \beta_1 - \beta_2) < \frac{\alpha^2}{\alpha_2} + \frac{\beta^2}{\beta_2}. \tag{2}$$

In Theorem 1, we emphasize that $u_0 = \gcd(x_0, z_0, A) = \gcd(x_0, z_0) = \gcd(x_0, A) = \gcd(z_0, A)$ and $v_0 = \gcd(y_0, z_0, B) = \gcd(y_0, z_0) = \gcd(y_0, B) = \gcd(z_0, B)$ according to $Bx_0 - Ay_0 = z_0$ and $\gcd(A, B) = 1,\ \gcd(x_0, y_0) = 1$. Besides, in consideration of the applications to cryptanalysis of RSA, we may also need to deal with the equation $Bx_0 - Ay_0 = Cz_0$. The corresponding result is described as follows.

Theorem 2. *For Theorem 1, suppose unknown* $(x_0, y_0, z_0) \in (\mathbb{Z}^+)^3$ *satisfies* $Bx_0 - Ay_0 = Cz_0$ *instead of* $Bx_0 - Ay_0 = z_0$*, where* C *is a known positive integer satisfying* $\gcd(C, A) = 1$ *or* $\gcd(C, B) = 1$*. Here we still assume* $\beta_2 + \alpha_1 \approx \alpha_2 + \beta_1$ *together with* $z_0 \ll Bx_0, Ay_0$*, while* $Cz_0 \ll Bx_0, Ay_0$ *is unnecessary. Then the condition to find* $(x, y, z) = (x_0, y_0, z_0)$ *in Theorem 1, namely, Inequality (2), remains unchanged.*

The Sketch of Proof Here we use Coppersmith's method to construct the desired lattice and thus give the sketch proof of Theorem 1. Theorem 2 can be proved in a similar manner. The detailed proofs of Theorem 1, 2 are given in Appendix C.

Set $x_0' := x_0/u_0,\ a_0' := A/u_0,\ y_0' := y_0/v_0,\ b_0' := B/v_0,\ z_0^* := z_0/(u_0v_0)$ and define $f(b', x', z^*) := -b'x' + z^* \in \mathbb{Z}[b', x', z^*]$. Let m be a positive integer, and set $\tau := \lceil \frac{\alpha_2-\alpha}{\alpha_2} m \rceil,\ i := \lceil \frac{\beta_2-\beta}{\beta_2} m \rceil$. Then for $k = 0, 1, \cdots, m$, we define the following polynomials in $\mathbb{Z}[b', x', z^*, v]$:

$$g_k(b', x', z^*, v) := E^{\min\{i,m-k\}} v^i (b'x')^{m-k} [f(b', x', z^*)]^k A^{\max\{\tau-k,0\}},$$

where E is the inverse of B modulo A^τ (such E must exit since $\gcd(A, B) = 1$). From $Bx_0 - Ay_0 = z_0$ one can check $f(b_0', x_0', z_0^*) \equiv 0 \pmod{a_0'}$. Then together with $A \equiv 0 \pmod{a_0'}$, we obtain

$$g_k(b_0', x_0', z_0^*, v_0) \equiv 0 \pmod{(a_0')^\tau}, \qquad k = 0, 1, \cdots, m. \tag{3}$$

Besides, for every polynomial $g_k(b', x', z^*, v)$, we replace each occurrence of the monomial $b'v$ by B and each occurrence of EB by 1, according to the relation $b_0'v_0 = B$ and the fact $EB \equiv 1 \pmod{(a_0')^\tau}$.

Next we define $\Lambda^* := \left\{ \sum_{k=0}^{m} l_k g_k(b', x', z^*, v) \;\middle|\; l_0,\ l_1,\ \cdots,\ l_m \in \mathbb{Z} \right\}$. Similar to Section 3.2, Λ^* corresponds to an lattice Λ, whose dimension is $s := m+1$. And according to Lemmas 1, 2, one can check that the condition $\det(\Lambda) < [(a_0')^\tau]^s$ is sufficient to obtain the desired (x_0, y_0, z_0). Finally, after some calculation with $m \to \infty$, we know that $\det(\Lambda) < [(a_0')^\tau]^s$ is equivalent to Inequality (2), which completes the proof of Theorem 1.

4.2 Applications to Cryptanalysis of RSA

Generalized IFP with Shared MSBs or LSBs Since finding small solutions of the equation $Bx - Ay = z$ can be derived from the IFP introduced in [25], one important application of our first improvement is the generalized IFP proposed in [28]. We present our result for the generalized IFP with shared MSBs as follows.

Proposition 1. *Suppose that there are two RSA moduli $N_1 = p_1q_1$ and $N_2 = p_2q_2$, where a_1p_1 and a_2p_2 share t most significant bits for some unknown positive integers a_1 and a_2. And for some large positive integer M (one can take $M = \max\{N_1, N_2\}$ for example), we have*

$$2^t = M^{t^*},\ p_1 = M^{\alpha_1^*},\ p_2 = M^{\alpha_2^*},\ q_1 = M^{\beta_1^*},\ q_2 = M^{\beta_2^*},\ a_1 = M^{\delta_1^*},\ a_2 = M^{\delta_2^*},$$

where $\alpha_1^* + \delta_1^* \approx \alpha_2^* + \delta_2^*$. For simplicity, we also assume $\gcd(a_1, a_2) = 1$ and $\gcd(a_i, p_j) = \gcd(a_i, q_j) = 1$ for $i, j = 1, 2$. Then N_1 and N_2 can be factored in polynomial time if

$$t^* > \frac{\alpha_1^*\beta_1^*}{\alpha_1^* + \beta_1^*} + \frac{\alpha_2^*\beta_2^*}{\alpha_2^* + \beta_2^*} + \delta_1^* + \delta_2^*. \tag{4}$$

Proof. Since a_1p_1 and a_2p_2 share t MSBs, we have $a_1p_1 - a_2p_2 = \widetilde{p}$, $|\widetilde{p}| < M^{\alpha_1^* + \delta_1^* - t^*}$. Without loss of generality, we can assume $\widetilde{p} > 0$ here, otherwise we can take $\widetilde{p} = a_2p_2 - a_1p_1$. It is easy to obtain

$$N_1 \cdot a_1q_2 - N_2 \cdot a_2q_1 = \widetilde{p}q_1q_2.$$

Take $B = N_1,\ x_0 = a_1q_2,\ A = N_2,\ y_0 = a_2q_1,\ z_0 = \widetilde{p}q_1q_2$ in Theorem 1. Then we have $\gcd(A,B) = 1,\ \gcd(x_0,y_0) = 1,\ u_0 = \gcd(x_0,z_0,A) = q_2,\ v_0 = \gcd(y_0,z_0,B) = q_1$ and $\alpha_1 \approx \delta_1^* + \beta_2^*,\ \alpha_2 = \alpha_2^* + \beta_2^*,\ \alpha \approx \beta_2^*,\ \beta_1 \approx \delta_2^* + \beta_1^*,\ \beta_2 = \alpha_1^* + \beta_1^*,\ \beta \approx \beta_1^*,\ \gamma \approx \alpha_1^* + \delta_1^* - t^* + \beta_1^* + \beta_2^*$. Next from Inequality (2), one gets $(\alpha_1^* + \delta_1^* - t^* + \beta_1^* + \beta_2^*) + (\delta_1^* + \beta_2^*) - (\alpha_2^* + \beta_2^*) < \frac{(\beta_2^*)^2}{\alpha_2^* + \beta_2^*} + \frac{(\beta_1^*)^2}{\alpha_1^* + \beta_1^*}$, which finally reduces to Inequality (4) according to $\alpha_1^* + \delta_1^* \approx \alpha_2^* + \delta_2^*$. Thus if Inequality (4) holds, one can obtain the values of $x_0 = a_1q_2,\ y_0 = a_2q_1,\ z_0 = \widetilde{p}q_1q_2$ efficiently. Then $q_1 = v_0 = \gcd(y_0, z_0, N_1),\ q_2 = u_0 = \gcd(x_0, z_0, N_2)$ are computed and $N_1,\ N_2$ are easily factored. And thus Proposition 1 follows.

In Proposition 1, the condition $\gcd(a_1, a_2) = 1$ is natural. Otherwise, one can suppose $a := \gcd(a_1, a_2) = \gcd(a_1, a_2, \widetilde{p}) = M^{\theta^*}$, and get $(a_1/a)p_1 - (a_2/a)p_2 = \widetilde{p}/a$, which reduces to $N_1 \cdot (a_1/a)q_2 - N_2 \cdot (a_2/a)q_1 = (\widetilde{p}/a)q_1q_2$. And the final corresponding result is changed to $t^* > \frac{\alpha_1^*\beta_1^*}{\alpha_1^* + \beta_1^*} + \frac{\alpha_2^*\beta_2^*}{\alpha_2^* + \beta_2^*} + (\delta_1^* - \theta^*) + (\delta_2^* - \theta^*)$. As for the condition $\gcd(a_i, p_j) = \gcd(a_i, q_j) = 1\ (i, j = 1, 2)$, it is used to make $\gcd(x_0, y_0) = 1,\ \gcd(x_0, z_0, A) = q_2,\ \gcd(y_0, z_0, B) = q_1$ hold in the proof of Proposition 1. In fact the condition $\gcd(a_1, p_2) = \gcd(a_1, q_1) = \gcd(a_2, p_1) = \gcd(a_2, q_2) = 1$ is sufficient. Besides, we should assume $\gcd(A, B) = \gcd(N_2, N_1) = 1$, otherwise N_1, N_2 have already been factored.

As is known to all, there must exist unknown $a_1^*, a_2^* \in \mathbb{Z}^+$ such that $a_1^*p_1 - a_2^*p_2 = 1$ (or $a_2^*p_2 - a_1^*p_1 = 1$) due to $\gcd(p_1, p_2) = 1$. And $a_1^*p_1 - a_2^*p_2 = 1$ means that $a_1^*p_1$ and $a_2^*p_2$ share nearly all of their bits beginning from the most significant bit. Therefore, our attack of Proposition 1 may apply to the case when unknown a_1^*, a_2^* are small enough for $a_1^*p_1 - a_2^*p_2 = 1$.

Proposition 1 is obtained according to Theorem 1. Similarly, from Theorem 2 we have the following result for the generalized IFP with shared LSBs.

Proposition 2. *For Proposition 1, suppose a_1p_1 and a_2p_2 share t least significant bits, instead of t most significant bits. Then the condition to factor N_1 and N_2 in polynomial time, namely, Inequality (4), remains unchanged.*

Proof. Since a_1p_1 and a_2p_2 share t LSBs, we have $a_1p_1 - a_2p_2 = 2^t\widetilde{p}$, $|\widetilde{p}| < M^{\alpha_1^* + \delta_1^* - t^*}$. Then one obtains $N_1 \cdot a_1q_2 - N_2 \cdot a_2q_1 = 2^t \cdot \widetilde{p}q_1q_2$. Similarly, take $B = N_1,\ x_0 = a_1q_2,\ A = N_2,\ y_0 = a_2q_1,\ C = 2^t,\ z_0 = \widetilde{p}q_1q_2$ in Theorem 2, and finally we know Inequality (4) is sufficient to factor N_1 and N_2 efficiently. And thus Proposition 2 follows.

Table 1 summarizes related works in [12, 20, 25, 28] and our contribution. Among them, [25] considers the case of shared LSBs, and [12] is for the case of shared MSBs, while other results apply to the both two cases. From Table 1, we know the IFP is exactly the circumstance of $\alpha_1^* \approx \alpha_2^*,\ \beta_1^* \approx \beta_2^*,\ \delta_1^* = \delta_2^* = 0$ in

Table 1. Results for IFP and generalized IFP

IFP with shared MSBs or LSBs	Generalized IFP with shared MSBs or LSBs
(For $\alpha_1^* \approx \alpha_2^*,\ \beta_1^* \approx \beta_2^*,\ \delta_1^* = \delta_2^* = 0$)	(For any $\alpha_1^*, \alpha_2^*, \beta_1^*, \beta_2^*, \delta_1^*, \delta_2^*$)
Results in [25] and [12]: $t^* > 2\beta_1^*$	Results in [28]: $t^* > \beta_1^* + \beta_2^* + \delta_1^* + \delta_2^*$
Results in [20]: $t^* > \frac{2\alpha_1^*\beta_1^*}{\alpha_1^*+\beta_1^*}$	Our results: $t^* > \frac{\alpha_1^*\beta_1^*}{\alpha_1^*+\beta_1^*} + \frac{\alpha_2^*\beta_2^*}{\alpha_2^*+\beta_2^*} + \delta_1^* + \delta_2^*$

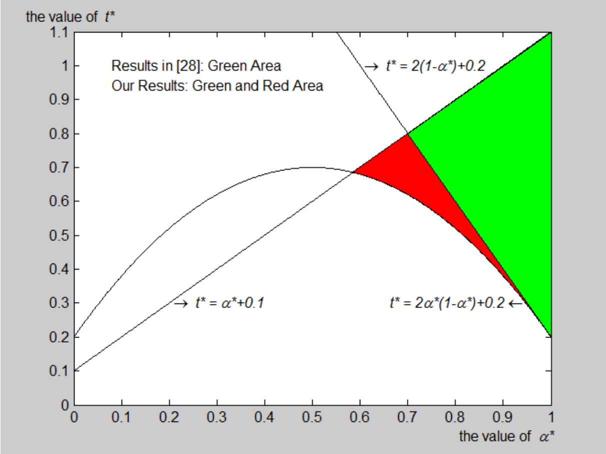

Fig. 1. Comparison between our results and those in [28] for generalized IFP with shared MSBs or LSBs when $\alpha_1^* \approx \alpha_2^* \approx \alpha^*,\ \beta_1^* \approx \beta_2^* \approx 1-\alpha^*,\ \delta_1^* \approx \delta_2^* \approx 0.1$

the generalized IFP. And just as [20] improves the results of [12, 25], our results are also better than those in [28].

Let $N_1 \approx N_2 \approx M$ and $\alpha_1^* \approx \alpha_2^* \approx \alpha^*,\ \beta_1^* \approx \beta_2^* \approx 1-\alpha^*,\ \delta_1^* \approx \delta_2^* \approx 0.1$ in Proposition 1 and Proposition 2. Then Inequality (4) turns out to be $t^* > 2\alpha^*(1-\alpha^*)+0.2$, while results given in [28] imply $t^* > 2(1-\alpha^*)+0.2$. Besides, there is another condition $t^* < \alpha_1^* + \delta_1^* \approx \alpha^* + 0.1$ since a_1p_1 and a_2p_2 share t bits. For this example, Figure 1 illustrates the comparison between our results and those in [28]. And our new improvement is denoted by the Red Area in Figure 1.

Given MSBs or LSBs of the Prime Factor An n-bit RSA modulus $N = pq$ with balanced p, q can be factored, if $0.25n$ MSBs or LSBs of p are exposed. This important result was proposed in [6, 7, 16] and is state-of-the-art up to now. Our new attacks, stated as the following two propositions, will improve the results in [6, 7, 16] if some greatest common divisor is large enough. We also consider the general case where p, q are not necessarily balanced.

Proposition 3. *For RSA modulus $N = pq$ with $p = N^{\alpha^*}$, suppose we have known t most significant bits of p with $2^t = N^{t^*}$. Namely, p_m is exposed for $p = p_m \cdot W + p_l$ with $W \approx N^{\alpha^*-t^*}$. Besides, we set $\gcd(p_l, kq+1) = N^{\theta^*}$ for* *some known small positive integer k (one can take $k = 1$ for example). Then N can be factored in polynomial time if*

$$t^* > \alpha^*(1 - \alpha^*) - (\theta^*)^2. \tag{5}$$

Proof. From $p = p_m \cdot W + p_l$, we obtain $N = Wp_mq + p_lq$, which is equivalent to

$$N \cdot (kq + 1) - (kN + Wp_m) \cdot q = p_lq,$$

with known N, p_m, W, k. Take $B = N,\ x_0 = kq + 1,\ A = kN + Wp_m,\ y_0 = q,\ z_0 = p_lq$ in Theorem 1, and one can get $\gcd(A, B) = 1,\ \gcd(x_0, y_0) = 1,\ u_0 = \gcd(x_0, z_0, A) = \gcd(p_l, kq + 1),\ v_0 = \gcd(y_0, z_0, B) = q$. Since the positive integer k is small enough, we obtain $\alpha_1 \approx 1 - \alpha^*,\ \alpha_2 \approx 1,\ \alpha \approx \theta^*,\ \beta_1 \approx 1 - \alpha^*,\ \beta_2 = 1,\ \beta \approx 1 - \alpha^*,\ \gamma \approx 1 - t^*$. Next from Inequality (2), one gets $(1-t^*)+(1-\alpha^*)-1 < (\theta^*)^2+(1-\alpha^*)^2$, which finally reduces to Inequality (5). Then N is factored after we obtain the values of x_0, y_0, z_0. And thus Proposition 3 follows.

Proposition 4. *For RSA modulus $N = pq$ with $p = N^{\alpha^*}$, suppose we have known t least significant bits of p with $2^t = N^{t^*}$. Namely, p_l is exposed for $p = p_m \cdot 2^t + p_l$. Besides, we set $\gcd(p_m, kq + 1) = N^{\theta^*}$ for some known small positive integer k (one can take $k = 1$ for example). Then N can be factored in polynomial time if*

$$t^* > \alpha^*(1 - \alpha^*) - (\theta^*)^2. \tag{6}$$

Proof. From $p = p_m \cdot 2^t + p_l$, we obtain $N = 2^tp_mq + p_lq$, which is equivalent to $N \cdot (kq + 1) - (kN + p_l) \cdot q = 2^tp_mq$ with known N, p_l, t, k. Similarly, take $B = N,\ x_0 = kq + 1,\ A = kN + p_l,\ y_0 = q,\ C = 2^t,\ z_0 = p_mq$ in Theorem 2, and finally we know Inequality (6) is sufficient to factor N efficiently. And thus Proposition 4 follows.

For the case of $\alpha^* = 0.5$, we obtain $t^* > 0.25-(\theta^*)^2$ from both Inequality (5) in Proposition 3 and Inequality (6) in Proposition 4. As mentioned before, the best known result is $t^* > 0.25$ for exposed MSBs of p (due to [6, 7, 16]) or exposed LSBs of p (due to [7, 16]). Thus we make improvement if $\theta^* > 0$.

Actually, we first generalize the result $t^* > 0.25$ in [6, 7, 16] to $t^* > \alpha^*(1-\alpha^*)$ for $0 < \alpha^* < 1$, then present an improvement $t^* > \alpha^*(1 - \alpha^*) - (\theta^*)^2$ if $\theta^* > 0$. Besides, there is another condition $t^* < \alpha^*$ since t bits of p are exposed. In Figure 2, we use the Green Area to denote our generalized result $t^* > \alpha^*(1-\alpha^*)$, and the Red Area to illustrate the possible improvement $t^* > \alpha^*(1 - \alpha^*) - (\min\{\alpha^* - t^*, 1 - \alpha^*\})^2$ for the extreme case of $\theta^* = \min\{\alpha^* - t^*, 1 - \alpha^*\}$. One can check that $t^* > \alpha^*(1 - \alpha^*) - (\min\{\alpha^* - t^*, 1 - \alpha^*\})^2$ is equivalent to $t^* > \alpha^*(1-\alpha^*)-(\alpha^*-t^*)^2,\ 0 < \alpha^* \leqslant 0.5$ or $t^* > \alpha^*(1-\alpha^*)-(1-\alpha^*)^2,\ 0.5 < \alpha^* < 1$.

In fact, in Figure 2 the area of actual improvement is much smaller than the Red Area since θ^* is very small at random. However, our attacks are also useful for the area of malicious generation of RSA moduli, i.e. the construction of backdoor RSA moduli. Here we take Proposition 4 with $\alpha^* = 0.5$ for example. After q is generated, one can choose a large enough factor of $q+1$ as p_m to make

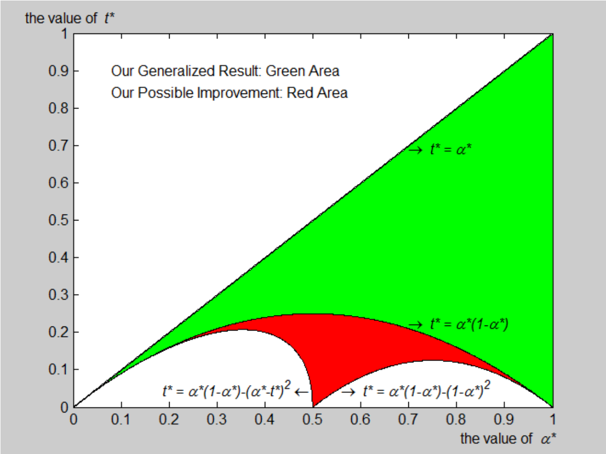

Fig. 2. Our generalized result and possible improvement for the attack on RSA with known MSBs or LSBs of the prime factor

$\gcd(p_m, q+1) = p_m$ hold, and it implies $\theta^* \approx 0.5 - t^*$. Then $p = p_m \cdot 2^t + p_l$ is generated, where p_l is randomly selected until p is a prime integer. Let θ^* approximate 0.5, and one can make $t^* \approx 0.5 - \theta^*$ small enough such that the bits of p_l can be easily obtained by side channel attacks together with exhaustive search. Notice that Inequality (6) in Proposition 4 turns out to be $t^* \approx 0.5 - \theta^* > 0.25 - (\theta^*)^2$, which always holds if $\theta^* \neq 0.5$. As a conclusion, this circumstance satisfies the condition to implement our attack and then factor $N = pq$.

Other Cryptanalysis of RSA We present some other cryptanalysis as the applications of our improvement to solve $Bx - Ay = z$. These results, together with the proofs, are similar to Proposition 3, which is obtained based on our Theorem 1.

First let us consider the partially approximate common divisor problem (PACDP), which was proposed by Howgrave-Graham [16] in 2001. And this problem plays an important role in the area of cryptanalysis of RSA.

Suppose there exist known $A, B \in \mathbb{Z}^+$ and unknown $d_0 \in \mathbb{Z}^+, a_0 \in \mathbb{Z}$, satisfying $\gcd((A + a_0), B) = d_0$, $\gcd(A, B) = 1$ and $A = M^{\alpha^*}$, $B = M^{\beta^*}$, $d_0 > M^{\delta^*}$, $|a_0| < M^{\gamma^*} \ll M^{\alpha^*}$ for some known $M \in \mathbb{Z}^+$. Howgrave-Graham [16] claimed that one can obtain such a_0, d_0 in polynomial time under the condition $\gamma^* < (\delta^*)^2/\beta^*$. Based on our Theorem 1, we can obtain a new condition $\gamma^* < (\delta^*)^2/\beta^* + (\theta^*)^2/\alpha^*$, where $\theta^* := \log_M[\gcd(A, |a_0|)]$. Thus our result is better if $\theta^* > 0$.

Next we focus on the result of factoring the RSA modulus by solving the equation $ex + y \equiv 0 \pmod{p}$, which was first presented by Nitaj [29] in 2012 and then improved by Lu et al. [22] in 2015.

For RSA modulus $N = pq$ with $p, q \approx N^{0.5}$ and public exponent $e = N^{\alpha^*}$, suppose there exist $x_0, y_0 \in \mathbb{Z}$ with $\gcd(x_0, y_0) = 1$ such that $ex_0 + y_0 \equiv 0 \pmod{p}$, $ex_0+y_0 \not\equiv 0 \pmod{N}$ and $|x_0| < N^{\gamma_1^*}$, $|y_0| < N^{\gamma_2^*}$. Nitaj [29] claimed that N can be factored in polynomial time if $\gamma_1^* + \gamma_2^* < (\sqrt{2} - 1)/2 \approx 0.207$ holds. Later Lu et al. [22] gave a better result $\gamma_1^* + \gamma_2^* < 0.25$. Based on our Theorem 1, we can obtain a new condition $\gamma_1^* + \gamma_2^* < 0.25 + (\theta^*)^2/\alpha^*$, where $\theta^* := \log_N[\gcd(e, |y_0|)]$. In the same way, we can make improvement for the case of $\theta^* > 0$. And if one supposes $p = N^{\delta^*}$ instead of $p, q \approx N^{0.5}$, our new condition can be generalized to $\gamma_1^* + \gamma_2^* < (\delta^*)^2 + (\theta^*)^2/\alpha^*$.

Besides, Nitaj also presented a similar attack on CRT-RSA in [29], which is again improved by Lu et al. in [22]. And we note that from our Theorem 1, one can also get a better result if some greatest common divisor is large enough.

5 Second Improvement for Solving $Bx - Ay = z$

In this section, we present the second improvement for solving the equation $Bx - Ay = z$. The main result is stated as Theorem 3 in Section 5.1. Then as an application of the second improvement, Section 5.2 shows an attack on RSA when some LSBs of the private exponent together with some MSBs of the prime factor are exposed. Subsequently, another attack aimed to factor an RSA modulus with a small difference of its prime factors is also proposed in Section 5.2.

5.1 Our Main Result

Our second improvement relies on another condition. Namely, it assumes that we have known the integer C satisfying $z_0 \equiv C \pmod{x_0}$, $z_0 \simeq z_0 - C$. Here $z_0 \simeq z_0 - C$ means that z_0 and $z_0 - C$ have the same bit size. Different from Theorem 1, this improvement does not need the condition $\gcd(A, B) = 1$, $\gcd(x_0, y_0) = 1$ in the proof. We present the second improvement as Theorem 3 below, together with an assumption on which Theorem 3 relies.

Assumption 1 *The resultant computations for the polynomials obtained by Coppersmith's method yield non-zero polynomials.*

Theorem 3. *Suppose there exists unknown $(x_0, y_0, z_0) \in (\mathbb{Z}^+)^3$ satisfying*

$$Bx_0 - Ay_0 = z_0, \quad |x_0| < X,\ |y_0| < Y,\ |z_0| < Z,$$
$$z_0 \equiv C \pmod{x_0}, \quad z_0 \simeq z_0 - C.$$

Here the integer C is known, and A, B, X, Y, Z are known large positive integers, and for some known large positive integer M we have

$$X = M^{\alpha_1},\ A = M^{\alpha_2},\ Y = M^{\beta_1},\ B = M^{\beta_2},\ Z = M^{\gamma}.$$

X, Y, Z are usually selected to satisfy $x_0 \approx X,\ y_0 \approx Y,\ z_0 \approx Z$, thus we also assume $\beta_2 + \alpha_1 \approx \alpha_2 + \beta_1$ holds such that $z_0 \ll Bx_0, Ay_0$. Then under Assumption 1, one can find all such solutions $(x, y, z) = (x_0, y_0, z_0)$ of the equation $Bx - Ay = z$ in polynomial time if $\alpha_1 < \alpha_2,\ 4\alpha_1 + \alpha_2 \leqslant 4\gamma$ and

$$\gamma + \alpha_1 - \alpha_2 \ (\approx \gamma + \beta_1 - \beta_2) < \frac{\alpha_1^2}{\alpha_2}. \tag{7}$$

Since Assumption 1 is heuristic, we need to perform experiments to examine it, which is done in Section 6. And the successful experimental results in Section 6 justify the validity of Theorem 3 and its applications to cryptanalysis of RSA.

The Sketch of Proof Based on Coppersmith's method, here we present the construction of the desired lattice and thus give the sketch proof of Theorem 3. And one can refer to Appendix D for the detailed proof.

From $z_0 \equiv C \pmod{x_0}$ we know that there exists an integer w_0 satisfying $x_0 w_0 = z_0 - C$. Define $f(x, z) := -Bx + z \in \mathbb{Z}[x, z]$, and let m, τ be two positive integers. Then we define the following polynomials in $\mathbb{Z}[x, z, w]$:

$$\begin{aligned} g_{t,j}(x, z) &:= x^{t-j}[f(x, z)]^j A^{m-j}, \quad j = 0, 1, \cdots, t, \quad t = 0, 1, \cdots, m, \\ h_{i,j}(x, z, w) &:= w^i [f(x, z)]^j A^{m-j}, \quad j = \theta_i, \theta_i + 1, \cdots, m, \quad i = 1, 2, \cdots, \tau, \end{aligned}$$

where $\theta_i := \lceil \eta i \rceil$ for an undetermined parameter η. From $Bx_0 - Ay_0 = z_0$ one gets $f(x_0, z_0) \equiv 0 \pmod{A}$. Thus we obtain

$$\begin{aligned} g_{t,j}(x_0, z_0) &\equiv 0 \pmod{A^m}, \quad j = 0, 1, \cdots, t, \quad t = 0, 1, \cdots, m, \\ h_{i,j}(x_0, z_0, w_0) &\equiv 0 \pmod{A^m}, \quad j = \theta_i, \theta_i + 1, \cdots, m, \quad i = 1, 2, \cdots, \tau. \end{aligned}$$

Besides, for every polynomial $h_{i,j}(x, z, w)$, we replace each occurrence of the monomial xw by $z - C$ according to the relation $x_0 w_0 = z_0 - C$.

Next we define Λ^* as the set of all integral linear combinations of these $g_{t,j}(x, z)$ and $h_{i,j}(x, z, w)$. Similar to Section 3.2, Λ^* corresponds to an lattice Λ, and we denote its dimension by s. According to Lemmas 1, 2 and under Assumption 1, one can check that the condition $\det(\Lambda) < (A^m)^{s-1}$ is sufficient to obtain the desired (x_0, y_0, z_0). Then take $\eta = \frac{m}{\tau} = \frac{\sqrt{\gamma - \alpha_1}}{\sqrt{\alpha_2} - \sqrt{\gamma - \alpha_1}}$ and $m \to \infty$, and after some calculation we know that $\det(\Lambda) < (A^m)^{s-1}$ is equivalent to Inequality (7).

For the proof of Theorem 3, we also have to make the basis matrix of Λ a square matrix and a lower triangular matrix. And for this purpose the condition $\theta_{i+1} \geqslant \theta_i + 1$ is sufficient. Finally in order to make $\theta_{i+1} \geqslant \theta_i + 1$ hold, we also need another two conditions $\alpha_1 < \alpha_2,\ 4\alpha_1 + \alpha_2 \leqslant 4\gamma$ besides Inequality (7).

5.2 Applications to Cryptanalysis of RSA

According to Theorem 3, we propose an attack on RSA when some LSBs of the private exponent together with some MSBs of the prime factor are exposed.

Proposition 5. *Given the RSA modulus $N = pq$ with $p, q \approx N^{0.5}$, the public exponent $e \approx N$, and the private exponent $d = N^{\beta^*}$. Suppose we have known t_1 most significant bits of p with $2^{t_1} = N^{t_1^*}$, and t_2 least significant bits of d with $2^{t_2} = N^{t_2^*}$. Namely, p_m is exposed for $p = p_m \cdot W + p_l$ with $W \approx N^{0.5-t_1^*}$, and d_l is exposed for $d = d_m \cdot 2^{t_2} + d_l$. Then under Assumption 1, one can factor N in polynomial time if $d_l < N^{\beta^*-t_1^*-0.5}$, $4t_1^* + t_2^* \leqslant 1$ and*

$$\beta^* < 1 + t_2^* - \sqrt{(1+t_2^*)(0.5-t_1^*)}. \tag{8}$$

Proof. Since p_m is exposed for $p = p_m \cdot W + p_l$, one can get the value of q_m for $q = q_m \cdot W + q_l$. According to $e \cdot d \equiv 1 \pmod{\varphi(N)}$, we know there exists $k \in \mathbb{Z}^+$ with $k \approx N^{\beta^*}$ such that $1 = ed - k\varphi(N) = e \cdot (d_m \cdot 2^{t_2} + d_l) - k \cdot [N - (p_m \cdot W + p_l) - (q_m W + q_l) + 1]$, which is equivalent to

$$(N - Wp_m - Wq_m + 1) \cdot k - 2^{t_2} e \cdot d_m = (p_l + q_l)k + ed_l - 1,$$

where only $(p_l + q_l), d_m, k$ are unknown. Take $B = N - Wp_m - Wq_m + 1,\ x_0 = k,\ A = 2^{t_2}e,\ y_0 = d_m,\ z_0 = (p_l + q_l)k + ed_l - 1$ in Theorem 3, and one can get $C = ed_l - 1$ satisfies $z_0 \equiv C \pmod{x_0}$ and $z_0 \simeq z_0 - C$ due to $d_l < N^{\beta^*-t_1^*-0.5}$. Besides, we have $\alpha_1 \approx \beta^*,\ \alpha_2 \approx 1 + t_2^*,\ \beta_1 \approx \beta^* - t_2^*,\ \beta_2 \approx 1,\ \gamma \approx \beta^* + 0.5 - t_1^*$, where $\alpha_1 < \alpha_2$ already holds, and $4\alpha_1 + \alpha_2 \leqslant 4\gamma \Leftrightarrow 4t_1^* + t_2^* \leqslant 1$. Then from Inequality (7), we obtain $(\beta^* + 0.5 - t_1^*) + \beta^* - (1 + t_2^*) < \frac{(\beta^*)^2}{1+t_2^*}$, which finally reduces to Inequality (8). Then N is factored after we obtain the values of x_0, y_0, z_0. And thus Proposition 5 follows.

In 2008, Sarkar and Maitra [36] obtained the result $t_2^* > g(\beta^*, t_1^*) := \frac{1}{3}(0.5 - t_1^*) + \frac{2}{3}\sqrt{(0.5-t_1^*)^2 + 3\beta^*(0.5-t_1^*)} + \beta^* - 1$, with the notations β^*, t_1^*, t_2^* defined in Proposition 5. On the other hand, Inequality (8) is equivalent to $t_2^* > h(\beta^*, t_1^*) := \frac{1}{2}(0.5 - t_1^*) + \frac{1}{2}\sqrt{(0.5-t_1^*)^2 + 4\beta^*(0.5-t_1^*)} + \beta^* - 1$. One can check that $g(\beta^*, t_1^*) > h(\beta^*, t_1^*)$ always holds for $\beta^* > 0$. Thus the result of our Proposition 5 is better than that of Sarkar and Maitra [36] under the extra condition $d_l < N^{\beta^*-t_1^*-0.5}$, $4t_1^* + t_2^* \leqslant 1$.

The comparison between the two results of Proposition 5 and [36] for $\beta^* = 0.700, 0.650, 0.600$ is illustrated by Table 2. For example, suppose that $\log_2 N \approx 1000$ and $\beta^* = 0.600$, the result of Sarkar and Maitra [36] shows that if 20

Table 2. Comparison between the result of our Proposition 5 (i.e. $t_2^* > h(\beta^*, t_1^*)$) and that in [36] (i.e. $t_2^* > g(\beta^*, t_1^*)$) for $\beta^* = 0.700, 0.650, 0.600$

t_1^*	0.000	0.010	0.020	0.030	0.040	0.050	0.060	0.070	0.080
$g(0.700, t_1^*)$	0.627	0.614	0.602	0.589	0.577	0.564	0.551	0.539	0.526
$h(0.700, t_1^*)$	0.592	0.580	0.567	0.555	0.542	0.530	0.517	0.504	0.491
$g(0.650, t_1^*)$	0.555	0.542	0.530	0.518	0.505	0.493	0.480	0.468	0.455
$h(0.650, t_1^*)$	0.522	0.510	0.498	0.486	0.473	0.461	0.448	0.436	0.423
$g(0.600, t_1^*)$	0.482	0.470	0.457	0.445	0.433	0.421	0.409	0.396	0.384
$h(0.600, t_1^*)$	0.452	0.440	0.428	0.416	0.403	0.391	0.379	0.367	0.354

Table 3. Attacks on RSA related to Proposition 5

	$t_1^* = 0$	$t_1^* \geqslant 0$
$t_2^* = 0$	Result in [3, 14]: $\beta^* < 1 - \sqrt{0.5} \approx 0.292$ (No extra conditions)	Result in [30]: $\beta^* < 1 - \sqrt{0.5 - t_1^*}$ ($t_1^* \leqslant 0.25$)
$t_2^* \geqslant 0$	Result in [39]: $\beta^* < 1 + t_2^* - \sqrt{0.5(1 + t_2^*)}$ ($d_l < N^{\beta^* - 0.5}$)	Our result: $\beta^* < 1 + t_2^* - \sqrt{(1 + t_2^*)(0.5 - t_1^*)}$ ($d_l < N^{\beta^* - t_1^* - 0.5}$, $4t_1^* + t_2^* \leqslant 1$)

bits (the MSBs) of p are obtained (i.e. $t_1^* = 0.020$) by exhaustive search (or side channel attacks), one needs more than 457 known bits (the LSBs) of d to factor the RSA modulus $N = pq$. While according to our Proposition 5, more than 428 known bits (the LSBs) of d are sufficient for the case of $d_l < N^{\beta^* - t_1^* - 0.5}$, $4t_1^* + t_2^* \leqslant 1$.

Note that usually one has $d_l \approx N^{t_2^*}$ at random, thus the condition $d_l < N^{\beta^* - t_1^* - 0.5}$ implies that we assume some MSBs of the exposed d_l are all 0 bits. When it comes to implementation of RSA, one may choose a private exponent d consisting of many 0 bits in order to improve the performance by lowering the Hamming weight of d and thus reducing the total number of multiplications. And the condition $d_l < N^{\beta^* - t_1^* - 0.5}$ may hold for this case.

Besides, let us consider several special cases for Proposition 5. (I) For the case of $t_1^* = 0$, the condition $4t_1^* + t_2^* \leqslant 1$ naturally holds. Thus to factor N we only need $d_l < N^{\beta^* - 0.5}$ and $\beta^* < 1 + t_2^* - \sqrt{0.5(1 + t_2^*)}$. This is exactly the result in [39]. (II) For the case of $t_2^* = 0$, we know there are no exposed bits in d. Therefore, we should take $d_l = 0$ and the condition $d_l < N^{\beta^* - t_1^* - 0.5}$ also naturally holds. In order to factor N, now we only need $t_1^* \leqslant 0.25$ and $\beta^* < 1 - \sqrt{0.5 - t_1^*}$, which is presented in [30]. Here we note that if $t_1^* > 0.25$, one can also successfully factor N according to Proposition 3 for $p, q \approx N^{0.5}$. (III) As for the case of $t_1^* = t_2^* = 0$, from above analysis we know both the condition $d_l < N^{\beta^* - t_1^* - 0.5}$ and the condition $4t_1^* + t_2^* \leqslant 1$ already hold. And one can obtain the best small private exponent attack result $\beta^* < 1 - \sqrt{0.5} \approx 0.292$ shown in [3] and [14]. All the above cases have been summarized in Table 3.

Finally, we present another attack based on Theorem 3. It considers an RSA modulus with a small difference of its prime factors.

Proposition 6. *Given the RSA modulus $N = pq$ with $p, q \approx N^{0.5}$, the public exponent $e \approx N$, and the private exponent $d = N^{\beta^*}$. Suppose p, q share t most significant bits with $2^t = N^{t^*}$, namely, we have $|p - q| < N^{0.5 - t^*}$. Then under Assumption 1, one can factor N in polynomial time if*

$$t^* \leqslant 0.125, \quad \beta^* < 1 - \sqrt{0.5 - 2t^*}. \tag{9}$$

Proof. From $|p - q| < N^{0.5 - t^*}$ and $(p - q)^2 = (p + q)^2 - 4N = (p + q + 2N^{0.5})(p + q - 2N^{0.5})$, one obtains $0 < p + q - 2N^{0.5} = (p - q)^2/(p + q + 2N^{0.5}) < (p -$

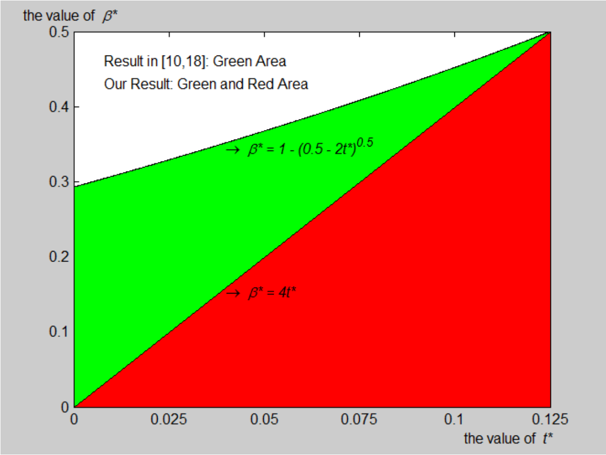

Fig. 3. Comparison between the result of our Proposition 6 and that in [10, 18] for the attack on RSA with a small difference of prime factors

$q)^2/4N^{0.5} < (1/4)\cdot N^{0.5-2t^*}$. Take $D := \lceil 2N^{0.5}\rceil$, and omit the constant factor $1/4$. Then roughly we have $|p+q-D| < N^{0.5-2t^*}$. From $e\cdot d \equiv 1 \pmod{\varphi(N)}$, we know there exists $k \in \mathbb{Z}^+$ with $k \approx N^{\beta^*}$ such that $1 = ed - k\varphi(N) = e\cdot d - k\cdot[(N-D+1)-(p+q-D)]$. It is equivalent to $Bx_0 - Ay_0 = z_0$, where $B := N-D+1 \approx N$, $x_0 := k \approx N^{\beta^*}$, $A := e \approx N$, $y_0 := d = N^{\beta^*}$, $z_0 := (p+q-D)k - 1 < N^{\beta^*+0.5-2t^*}$ with $z_0 \equiv -1 \pmod{x_0}$. According to the conditions $\alpha_1 < \alpha_2$, $4\alpha_1 + \alpha_2 \leqslant 4\gamma$, $\gamma + \alpha_1 - \alpha_2 < \alpha_1^2/\alpha_2$ in Theorem 3, we can obtain Inequality (9) and complete the proof of Proposition 6 as before.

In 2002, by generalizing the best small private exponent attack result $d < N^{0.292}$ [3], Weger [10] obtained the condition $4t^* \leqslant \beta^*$, $\beta^* < 1 - \sqrt{0.5 - 2t^*}$ for factoring an RSA modulus with a small difference of its prime factors. Another proof of Weger's result can also be found in [18]. Since $4t^* \leqslant \beta^*$, $\beta^* < 1 - \sqrt{0.5-2t^*} \Rightarrow 4t^* < 1 - \sqrt{0.5-2t^*}$, $\beta^* < 1 - \sqrt{0.5-2t^*} \Rightarrow t^* \leqslant 0.125$, $\beta^* < 1 - \sqrt{0.5-2t^*}$, the condition of our Proposition 6 is weaker than that in [10, 18], which implies that our result is better. It is also illustrated by Figure 3, where the Red Area denotes our new improvement.

6 Experiments

In order to examine the justification of our approach based on Coppersmith's method, we have implemented several experiments in SAGE 5.0 over Linux Fedora 16 on a laptop with 2.80GHz Intel Core2 CPU and 4GB RAM. All the experiments are successful, and thus they verify the correctness of our results. Some experimental examples are given in Tables 4, 5 and 6 below.

6.1 Experimental Examples for Our First Improvement

Table 4 shows experiments for our results of generalized IFP with shared MSBs or LSBs. The notations $M, N_1, N_2, t^*, \alpha_1^*, \alpha_2^*, \beta_1^*, \beta_2^*, \delta_1^*, \delta_2^*$ are already defined in Proposition 1 or 2. And in Section 4.1 (or Appendix C) we have defined the notations m, τ, i, which are used for the construction of the lattice Λ and the proof of Theorem 1 or 2. Besides, $\dim(\Lambda)$ denotes the dimension of Λ, and "Time(LLL)" denotes the time used for LLL algorithm for each experimental example. As for "Bit size", we just means $\log_2\{[\det(\Lambda)]^{1/\dim(\Lambda)}\}$. Let $\mathcal{B}$ denote the basis matrix of Λ and we know $\mathcal{B}$ is a lower triangular matrix from Appendix C. Note that $\det(\Lambda) = \sqrt{\det(\mathcal{B}\mathcal{B}^T)} = |\det \mathcal{B}| = \det \mathcal{B}$. Thus $[\det(\Lambda)]^{1/\dim(\Lambda)}$ is the geometric mean of the diagonal entries of the basis matrix $\mathcal{B}$, and one can roughly regard $\log_2\{[\det(\Lambda)]^{1/\dim(\Lambda)}\}$ as the bit size of the entries of $\mathcal{B}$. According to Lemma 1, the time for the LLL algorithm is related to both $\dim(\Lambda)$ and the bit size of the entries of $\mathcal{B}$.

The experimental examples of Table 5 are about our results when some MSBs or LSBs of the prime factor are given. The definition of the notations $N, k, t^*, \alpha^*, \theta^*$ can be found in Proposition 3 or 4. And again Section 4.1 (or Appendix C) defines the notations m, τ, i for the construction of the lattice Λ. The experimental examples for other cryptanalysis of RSA in Section 4.2 are similar to those for Proposition 3.

Table 4. Some experimental examples for Proposition 1 (the case of MSBs) and Proposition 2 (the case of LSBs) with $\beta_1^* \approx 1 - \alpha_1^*$, $\beta_2^* \approx 1 - \alpha_2^*$ and $\log_2 M \approx \log_2 N_1 \approx \log_2 N_2 \approx 1500$

Case	t^*	α_1^*	α_2^*	δ_1^*	δ_2^*	m	τ	i	$\dim(\Lambda)$	Bit size	Time(LLL)
MSBs	0.606	0.679	0.684	0.066	0.061	15	10	10	16	1.008×10^4	6.596 seconds
MSBs	0.533	0.666	0.799	0.133	0.000	15	11	9	16	1.198×10^4	7.649 seconds
MSBs	0.687	0.733	0.500	0.000	0.233	23	16	11	24	1.725×10^4	116.3 seconds
LSBs	0.529	0.580	0.579	0.000	0.000	23	13	13	24	1.093×10^4	27.73 seconds
LSBs	0.433	0.752	0.805	0.053	0.000	19	15	14	20	1.782×10^4	35.12 seconds
LSBs	0.756	0.803	0.479	0.000	0.324	19	15	9	20	1.802×10^4	88.97 seconds

Table 5. Some experimental examples for Proposition 3 (the case of MSBs) and Proposition 4 (the case of LSBs) with $k = 1$ and $\log_2 N \approx 1000$

Case	t^*	α^*	θ^*	m	τ	i	$\dim(\Lambda)$	Bit size	Time(LLL)
MSBs	0.240	0.500	0.165	27	13	22	28	6.474×10^3	23.14 seconds
MSBs	0.203	0.603	0.247	27	16	20	28	9.531×10^3	48.51 seconds
MSBs	0.260	0.400	0.000	42	16	42	43	6.101×10^3	102.1 seconds
LSBs	0.156	0.501	0.332	42	21	28	43	1.043×10^4	756.2 seconds
LSBs	0.255	0.473	0.071	34	16	31	35	7.582×10^3	65.23 seconds
LSBs	0.220	0.720	0.000	34	24	34	35	1.709×10^4	117.4 seconds

Table 6. Some experimental examples for Proposition 5 (under the extra condition $d_l < N^{\beta^* - t_1^* - 0.5}$, $4t_1^* + t_2^* \leqslant 1$) with $\log_2 N \approx 2000$

β^*	t_1^*	t_2^*	m	τ	$\theta_1, \theta_2, \cdots, \theta_\tau$	$\dim(\varLambda)$	Bit size	Time(LLL)
0.650	0.100	0.500	5	4	$2, 3, 4, 5$	31	1.480×10^4	40.83 seconds
0.656	0.141	0.435	5	4	$2, 3, 4, 5$	31	1.426×10^4	36.50 seconds
0.580	0.049	0.480	7	5	$2, 3, 4, 5, 7$	55	2.024×10^4	1282 seconds
0.543	0.012	0.440	7	5	$2, 3, 5, 6, 7$	53	1.989×10^4	680.9 seconds
0.784	0.072	0.692	6	5	$2, 3, 4, 5, 6$	43	2.022×10^4	238.9 seconds
0.708	0.118	0.525	6	5	$2, 3, 4, 5, 6$	43	1.820×10^4	225.9 seconds

6.2 Experimental Examples for Our Second Improvement

As for our results obtained by the second improvement for solving the equation $Bx - Ay = z$, they all rely on Assumption 1. Here we point out that in all of our experiments for Propositions 5, 6, Assumption 1 always holds and RSA modulus N can always be successfully factored. Table 6 gives some experimental examples for Proposition 5. Similar to Tables 4, 5, the notations $N, d_l, \beta^*, t_1^*, t_2^*$ are defined in Proposition 5, while in Section 5.1 (or Appendix D) one can find the definition of the notations $m, \tau, \theta_1, \theta_2, \cdots, \theta_\tau$, which are used to construct $\varLambda$ and prove Theorem 3. The experimental examples for Proposition 6 are similar to those for Proposition 5.

7 Conclusion

In this paper, we revisit some cryptanalysis of RSA, which are summarized into one framework, namely, finding small solutions $(x, y, z) = (x_0, y_0, z_0)$ of the equation $Bx - Ay = z$. For the general case of solving this equation, we show that Wiener's method, May-Ritzenhofen's method and Coppersmith's method are equivalent, and they give the same result after omitting some negligible terms. For some special cases, we present two improvements for solving $Bx - Ay = z$ based on Coppersmith's method. And according to these two improvements, we obtain some new applications to cryptanalysis of RSA, such as new results about the generalized IFP, attacks with known bits of the prime factor, an attack on RSA when some bits of the private exponent together with the prime factor are exposed, and so on. The justification of our approach is also examined through experiments. Moreover, we believe that our two improvements to solve $Bx - Ay = z$ may find other new applications to cryptanalysis of RSA.

References

1. Aono Y. A new lattice construction for partial key exposure attack for RSA. Public Key Cryptography-PKC 2009. Springer Berlin Heidelberg, 2009: 34-53.
2. Blömer J, May A. New partial key exposure attacks on RSA. Advances in Cryptology-CRYPTO 2003. Springer Berlin Heidelberg, 2003: 27-43.

3. Boneh D, Durfee G. Cryptanalysis of RSA with private key d less than $N^{0.292}$. Advances in Cryptology-EUROCRYPT 1999. Springer Berlin Heidelberg, 1999: 1-11.
4. Boneh D, Durfee G, Frankel Y. An attack on RSA given a small fraction of the private key bits. Advances in Cryptology-ASIACRYPT 1998. Springer Berlin Heidelberg, 1998: 25-34.
5. Coppersmith D. Finding a small root of a univariate modular equation. Advances in Cryptology-EUROCRYPT 1996. Springer Berlin Heidelberg, 1996: 155-165.
6. Coppersmith D. Finding a small root of a bivariate integer equation; factoring with high bits known. Advances in Cryptology-EUROCRYPT 1996. Springer Berlin Heidelberg, 1996: 178-189.
7. Coppersmith D. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. Journal of Cryptology, 1997, 10(4): 233-260.
8. Coron J S. Finding small roots of bivariate integer polynomial equations revisited. Advances in Cryptology-EUROCRYPT 2004. Springer Berlin Heidelberg, 2004: 492-505.
9. Coron J S, May A. Deterministic polynomial-time equivalence of computing the RSA secret key and factoring. Journal of Cryptology, 2007, 20(1): 39-50.
10. De Weger B. Cryptanalysis of RSA with small prime difference. Applicable Algebra in Engineering, Communication and Computing, 2002, 13(1): 17-28.
11. Ernst M, Jochemsz E, May A, et al. Partial key exposure attacks on RSA up to full size exponents. Advances in Cryptology-EUROCRYPT 2005. Springer Berlin Heidelberg, 2005: 371-386.
12. Faugère J C, Marinier R, Renault G. Implicit factoring with shared most significant and middle bits. Public Key Cryptography-PKC 2010. Springer Berlin Heidelberg, 2010: 70-87.
13. Hardy G H, Wright E M. An introduction to the theory of numbers. Oxford University Press, 1979.
14. Herrmann M, May A. Maximizing small root bounds by linearization and applications to small secret exponent RSA. Public Key Cryptography-PKC 2010. Springer Berlin Heidelberg, 2010: 53-69.
15. Howgrave-Graham N. Finding small roots of univariate modular equations revisited. Crytography and Coding. Springer Berlin Heidelberg, 1997: 131-142.
16. Howgrave-Graham N. Approximate integer common divisors. Cryptography and Lattices. Springer Berlin Heidelberg, 2001: 51-66.
17. Joux A. Algorithmic cryptanalysis. CRC Press, 2009.
18. Kumar S, Narasimham C. Cryptanalysis of RSA with small prime difference using unravelled linearization. International Journal of Computer Applications, 2013, 61(3).
19. Lenstra A K, Lenstra H W, Lovász L. Factoring polynomials with rational coefficients. Mathematische Annalen, 1982, 261(4): 515-534.
20. Lu Y, Peng L, Zhang R, et al. Towards optimal bounds for implicit factorization problem. International Conference on Selected Areas in Cryptography-SAC 2015. Springer International Publishing, 2015: 462-476.
21. Lu Y, Zhang R, Lin D. Improved bounds for the implicit factorization problem. Advances in Mathematics of Communications, 2013, 7(3): 243-251.
22. Lu Y, Zhang R, Peng L, et al. Solving linear equations modulo unknown divisors: revisited. Advances in Cryptology-ASIACRYPT 2015. Springer Berlin Heidelberg, 2015: 189-213.
23. May A. New RSA vulnerabilities using lattice reduction methods. Dissertation for Ph.D. Degree, University of Paderborn, 2003.

24. May A. Computing the RSA secret key is deterministic polynomial time equivalent to factoring. Advances in Cryptology-CRYPTO 2004. Springer Berlin Heidelberg, 2004: 213-219.
25. May A, Ritzenhofen M. Implicit factoring: on polynomial time factoring given only an implicit hint. Public Key Cryptography-PKC 2009. Springer Berlin Heidelberg, 2009: 1-14.
26. Meyer C D. Matrix analysis and applied linear algebra. Cambridge University Press, Cambridge, 2000.
27. Minkowski H. Geometrie der Zahlen. Teubner-Verlag, 1896.
28. Nitaj A, Ariffin M R K. Implicit factorization of unbalanced RSA moduli. Journal of Applied Mathematics and Computing, 2015, 48(1-2): 349-363.
29. Nitaj A. A new attack on RSA and CRT-RSA. Progress in Cryptology-AFRICACRYPT 2012. Springer International Publishing, 2012: 221-233.
30. Peng L, Hu L, Huang Z, et al. Partial prime factor exposure attacks on RSA and its Takagi's variant. International Conference on Information Security Practice and Experience-ISPEC 2015. Springer International Publishing, 2015: 96-108.
31. Peng L, Hu L, Lu Y, et al. Implicit factorization of RSA moduli revisited (short paper). International Workshop on Security-IWSEC 2015. Springer International Publishing, 2015: 67-76.
32. Peng L, Hu L, Xu J, et al. Further improvement of factoring RSA moduli with implicit hint. Progress in Cryptology-AFRICACRYPT 2014. Springer International Publishing, 2014: 165-177.
33. Rivest R L, Shamir A, Adleman L. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 1978, 21(2): 120-126.
34. Sarkar S. Partial key exposure: generalized framework to attack RSA. Progress in Cryptology-INDOCRYPT 2011. Springer Berlin Heidelberg, 2011: 76-92.
35. Sarkar S, Gupta S S, Maitra S. Partial key exposure attack on RSA - improvements for limited lattice dimensions. Progress in Cryptology-INDOCRYPT 2010. Springer Berlin Heidelberg, 2010: 2-16.
36. Sarkar S, Maitra S. Improved partial key exposure attacks on RSA by guessing a few bits of one of the prime factors. International Conference on Information Security and Cryptology. Springer, Berlin, Heidelberg, 2008: 37-51.
37. Sarkar S, Maitra S. Approximate integer common divisor problem relates to implicit factorization. IEEE Transactions on Information Theory, 2011, 57(6): 4002-4013.
38. Takayasu A, Kunihiro N. Partial key exposure attacks on RSA: achieving the boneh-durfee bound. International Workshop on Selected Areas in Cryptography-SAC 2014. Springer International Publishing 2014: 345-362.
39. Wang S, Qu L, Li C, et al. Generalized framework to attack RSA with special exposed bits of the private key. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 2017, 100(10): 2113-2122.
40. Wiener M J. Cryptanalysis of short RSA secret exponents. IEEE Transactions on Information theory, 1990, 36(3): 553-558.

Appendix A: Wiener's Method to Prove Result 1

Wiener's method is based on approximations using continued fractions. Thus at first we need to briefly introduce continued fraction and a related lemma, and one can see [13] for details.

Let $\eta = \eta_0$ be a positive rational number. For $i = 0, 1, 2 \cdots$, define $a_i = \lfloor \eta_i \rfloor$ and $\eta_{i+1} = 1/(\eta_i - a_i)$ unless η_n is an integer for some $n \geqslant 0$ (there must exists such an n since η is a rational number). Then η can be expressed as a continued fraction, namely,

$$\eta =< a_0, a_1, \cdots, a_{n-1}, a_n >:= a_0 + 1/(a_1 + 1/(\cdots + 1/(a_{n-1} + 1/a_n) \cdots)).$$

For $i \geqslant 0$, $A_i/B_i :=< a_0, a_1, \cdots, a_i >$ are called the convergents of $\eta = A_n/B_n$, where $\gcd(A_n, B_n) = 1$. And we note that the total number of convergents is polynomial in $\log(B_n)$. The related lemma [13, Theorem 184] is stated as follows:

Lemma 3. ***(Legendre)*** *Let η be a positive rational number. Suppose*

$$\left| \eta - \frac{x_0}{y_0} \right| < \frac{1}{2y_0^2}$$

and $\gcd(x_0, y_0) = 1$*. Then $\frac{x_0}{y_0}$ is one of the convergents of η.*

One can generalize the original method of Wiener's attack in [40] to get the proof of Result 1. For simplicity, here we use Lemma 3 to directly prove it. And the method is almost the same as Wiener's original method.

From $Bx_0 - Ay_0 = z_0$ we have $\left| \frac{A}{B} - \frac{x_0}{y_0} \right| = \frac{x_0}{y_0} - \frac{A}{B} = \frac{z_0}{By_0}$. According to Lemma 3, if $\frac{z_0}{By_0} < \frac{1}{2y_0^2}$ holds, $\frac{x_0}{y_0}$ is one of the convergents of $\frac{A}{B}$. The total number of convergents is polynomial in $\log B$. Thus we try every convergent of $\frac{A}{B}$ as the value of $\frac{x_0}{y_0}$ and compute z_0. Then from them we will obtain small solutions (x_0, y_0, z_0) satisfying $|x_0| < X$, $|y_0| < Y$, $|z_0| < Z$. As a conclusion, the only condition to find (x_0, y_0, z_0) is

$$\tfrac{z_0}{By_0} < \tfrac{1}{2y_0^2} \;\Leftrightarrow\; 2y_0 z_0 < B \;\Leftrightarrow\; \log_M 2 + \log_M y_0 + \log_M z_0 < \log_M B.$$

Omitting $\log_M 2$, we know the condition $\gamma + \beta_1 - \beta_2 < 0$ is sufficient, which completes the proof of Result 1.

If one considers Wiener's original method, the final condition obtained is $\frac{3}{2}y_0 z_0 < B$, instead of $2y_0 z_0 < B$. At last, we note that Wiener's result $d < N^{0.25}$ for small private exponent attack, can be directly obtained from Result 1.

Appendix B: May-Ritzenhofen's Method to Prove Result 1

Similar to Appendix A, at first we need to introduce some preliminaries used for May-Ritzenhofen's method. It includes the following three lemmas about lattice.

Recall that in Section 3.1 we have already introduced some basic knowledge about lattice, including the basis of a lattice, the determinant of a lattice and the norm of a vector. Hadamard's inequality [26] relates the norm of the basis vectors to the determinant for a lattice.

Lemma 4. ***(Hadamard)*** *Let* $\mathcal{B} = \begin{pmatrix} \boldsymbol{b_1} \\ \boldsymbol{b_2} \\ \vdots \\ \boldsymbol{b_s} \end{pmatrix} \in \mathbb{Z}^{s \times s}$ *be an arbitrary non-singular matrix. Then*

$$\det(\mathcal{B}) \leqslant \prod_{i=1}^{n} \|\boldsymbol{b_i}\|.$$

The successive minima $\lambda_i(\mathit{\Lambda})$ of the lattice $\mathit{\Lambda}$ are defined as the minimal radius of a ball containing i linearly independent lattice vectors of $\mathit{\Lambda}$. The shortest non-zero vector $\boldsymbol{v}$ of a lattice $\mathit{\Lambda}$ must have the norm $\|\boldsymbol{v}\| = \lambda_1(\mathit{\Lambda})$, and it also satisfies the Minkowski bound [27].

Lemma 5. ***(Minkowski)*** *Let s be the dimension (and the rank) of the lattice $\mathit{\Lambda}$. Then $\mathit{\Lambda}$ contains a non-zero vector $\boldsymbol{v}$ satisfying*

$$\|\boldsymbol{v}\| = \lambda_1(\mathit{\Lambda}) \leqslant \sqrt{s}[\det(\mathit{\Lambda})]^{1/s}.$$

For a two-dimensional lattice $\mathit{\Lambda}$, the basis vectors $\boldsymbol{v_1}, \boldsymbol{v_2}$ with norms $\|\boldsymbol{v_1}\| = \lambda_1(\mathit{\Lambda}), \|\boldsymbol{v_2}\| = \lambda_2(\mathit{\Lambda})$ can be efficiently computable by means of Gaussian reduction. Namely, we have the following lemma [26].

Lemma 6. ***(Gauss)*** *Given a basis (square) matrix $\mathcal{B}$ of a two-dimensional lattice $\mathit{\Lambda}$, the Gauss-reduced lattice basis vectors $\boldsymbol{v_1}, \boldsymbol{v_2}$, where*

$$\|\boldsymbol{v_1}\| = \lambda_1(\mathit{\Lambda}), \quad \|\boldsymbol{v_2}\| = \lambda_2(\mathit{\Lambda}),$$

can be determined in time $O(\log^2(\max\{\|\boldsymbol{v_1}\|, \|\boldsymbol{v_2}\|\}))$.

Now we can generalize May and Ritzenhofen's attack in [25] to get another proof of Result 1. It includes two cases, namely, $\gamma \geqslant \alpha_1$ and $\gamma < \alpha_1$.

(I) Suppose $\gamma \geqslant \alpha_1$, and select a positive integer C satisfying $C \approx M^{\gamma-\alpha_1}$. We construct a two-dimensional lattice $\mathit{\Lambda}$ whose basis (square) matrix $\mathcal{B}$ is

$$\begin{pmatrix} C & B \\ 0 & A \end{pmatrix}.$$

From $Bx_0 - Ay_0 = z_0$ we know that $\boldsymbol{u} := (Cx_0, z_0) \in \mathit{\Lambda}$. Then according to Lemma 6, we can obtain the basis vectors $\boldsymbol{v_1}, \boldsymbol{v_2}$ with norms $\|\boldsymbol{v_1}\| = \lambda_1(\mathit{\Lambda}), \|\boldsymbol{v_2}\| = \lambda_2(\mathit{\Lambda})$. And there exist $a_1, a_2 \in \mathbb{Z}$, such that $\boldsymbol{u} = a_1\boldsymbol{v_1} + a_2\boldsymbol{v_2}$.

If $\|\boldsymbol{u}\| < \lambda_2(\mathit{\Lambda})$ holds, from the definition of $\lambda_2(\mathit{\Lambda})$ one can obtain $a_2 = 0$ and $\boldsymbol{u} = a_1\boldsymbol{v_1} \Leftrightarrow \boldsymbol{v_1} = \boldsymbol{u}/a_1 = (C \cdot x_0/a_1, z_0/a_1)$. From $B \cdot x_0/a_1 - A \cdot y_0/a_1 = z_0/a_1$ and the fact that $\boldsymbol{v_1}$ is generated from the basis matrix $\mathcal{B}$, we have $\boldsymbol{v_1} = (C \cdot x_0/a_1, z_0/a_1) = x_0/a_1 \cdot (C, B) - y_0/a_1 \cdot (0, A)$. Thus $x_0/a_1, -y_0/a_1 \in \mathbb{Z}$, and we have $a_1|x_0,\ a_1|y_0$. Since $\gcd(x_0, y_0) = 1$, finally one gets $a_1 = \pm 1$ and $\boldsymbol{v_1} = \pm\boldsymbol{u} = \pm(Cx_0, z_0)$. The vector $\boldsymbol{v_1}$ is already obtained, thus we can easily get the values of x_0, z_0 and then the value of y_0.

As a conclusion, the only condition to find (x_0, y_0, z_0) is $\|\boldsymbol{u}\| < \lambda_2(\mathit{\Lambda})$. Lemma 5 tells $\lambda_1(\mathit{\Lambda}) \leqslant \sqrt{2}[\det(\mathit{\Lambda})]^{1/2}$, while Lemma 4 implies $\det(\mathit{\Lambda}) = \det(\mathcal{B}) \leqslant$

$\|\boldsymbol{v_1}\|\|\boldsymbol{v_2}\| = \lambda_1(\Lambda)\lambda_2(\Lambda)$. Thus we obtain $\lambda_2(\Lambda) \geqslant \frac{\det(\Lambda)}{\lambda_1(\Lambda)} \geqslant \frac{\det(\Lambda)}{\sqrt{2}[\det(\Lambda)]^{1/2}} = \frac{1}{\sqrt{2}}(AC)^{1/2} \approx M^{(\gamma-\alpha_1+\alpha_2)/2}$. Roughly we have $\|\boldsymbol{u}\| \leqslant \max\{\sqrt{2}Cx_0, \sqrt{2}z_0\} < \max\{\sqrt{2}CX, \sqrt{2}Z\} \approx M^{\gamma}$. Hence to make $\|\boldsymbol{u}\| < \lambda_2(\Lambda)$ hold, it is sufficient that

$$M^{\gamma} < M^{(\gamma-\alpha_1+\alpha_2)/2} \;\Leftrightarrow\; \gamma < (\gamma - \alpha_1 + \alpha_2)/2 \;\Leftrightarrow\; \gamma + \alpha_1 - \alpha_2 < 0.$$

(II) For the case of $\gamma < \alpha_1$, we will select a positive integer D satisfying $D \approx M^{\alpha_1-\gamma}$, and the basis (square) matrix $\mathcal{B}$ is changed to

$$\begin{pmatrix} 1 & D \cdot B \\ 0 & D \cdot A \end{pmatrix}.$$

The rest of the proof is almost the same as the former case and we omit it.

From the above, we know Result 1 follows. Finally, we note that May and Ritzenhofen's result for the IFP, can be directly obtained from Result 1.

Appendix C: Detailed Proofs of Theorems 1, 2

Firstly, based on Coppersmith's method we present the detailed proof of Theorem 1. We may again give the same definitions for some notations which are already introduced in the sketch of proof in Section 4.1. That allows us to show an integrated proof, which makes it unnecessary to review the sketch of proof again. Secondly, we mainly introduce the polynomials for the lattice construction to prove a lemma which is similar to Theorem 1. Finally, the proof of Theorem 2 is given directly as a consequence of this lemma.

(I) Set $x_0' := x_0/u_0,\ a_0' := A/u_0,\ y_0' := y_0/v_0,\ b_0' := B/v_0,\ z_0^* := z_0/(u_0v_0)$ and $B' := \lceil B/V \rceil,\ X' := \lceil X/U \rceil,\ Z^* := \lceil Z/(UV) \rceil$. Then we have $|b_0'| < B',\ |x_0'| < X',\ |z_0^*| < Z^*$.

Define $f(b', x', z^*) := -b'x' + z^* \in \mathbb{Z}[b', x', z^*]$. From

$$Bx_0 - Ay_0 = z_0 \;\Leftrightarrow\; b_0'v_0x_0'u_0 - a_0'u_0y_0'v_0 = z_0^*u_0v_0 \;\Leftrightarrow\; b_0'x_0' - a_0'y_0' = z_0^*,$$

we obtain

$$f(b_0', x_0', z_0^*) \equiv 0 \pmod{a_0'}.$$

Besides the above three variables b', x', z^*, we also introduce another variable v for v_0.

Let the positive integers m, τ and the non-negative integer i be undetermined parameters. Then for $k = 0, 1, \cdots, m$, we define

$$g_k(b', x', z^*, v) := E^{\min\{i, m-k\}} v^i (b'x')^{m-k} [f(b', x', z^*)]^k A^{\max\{\tau-k, 0\}},$$

where E is the inverse of B modulo A^{τ}, namely, $EB \equiv 1 \pmod{A^{\tau}}$. Such E must exit since $\gcd(A, B) = 1$. Recall that $f(b_0', x_0', z_0^*) \equiv 0 \pmod{a_0'}$ and $A \equiv 0 \pmod{a_0'}$, thus we obtain

$$g_k(b_0', x_0', z_0^*, v_0) \equiv 0 \pmod{(a_0')^{\tau}}, \qquad k = 0, 1, \cdots, m. \tag{10}$$

For every polynomial $g_k(b',x',z^*,v)$, we replace each occurrence of the monomial $b'v$ by B according to the relation $b'_0v_0 = B$. Then we replace each occurrence of EB by 1 in every $g_k(b',x',z^*,v)$, which does not contradict the correctness of Equation (10) since one can check $EB \equiv 1 \pmod{(a'_0)^\tau}$.

Next we need to define the monomial order "$\prec$". For convenience, we use $v^i(b')^{m-k}(x')^{m-k}(z^*)^k$ to denote $(b')^{(m-k)-i}(x')^{m-k}(z^*)^k$ if $i \leqslant m-k$ and denote $v^{i-(m-k)}(x')^{m-k}(z^*)^k$ if $i > m-k$. Then, "$\prec$" is defined such that $v^i(b')^{m-k_1}(x')^{m-k_1}(z^*)^{k_1} \prec v^i(b')^{m-k_2}(x')^{m-k_2}(z^*)^{k_2}$ if and only if $k_1 < k_2$. Let $\varLambda^*$ be the set of all integral linear combinations of $g_k(b',x',z^*,v)$ $(k = 0,1,2,\cdots,m)$. Similar to Section 3.2, there is a one-to-one correspondence between a polynomial $g(b',x',z^*,v)$ in $\varLambda^*$ and a vector $\boldsymbol{g}$ in a subset $\varLambda$ of $\mathbb{Z}^{m+1}$, where the components of $\boldsymbol{g}$ are the coefficients of $g(B'b',X'x',Z^*z^*,Vv)$ in the order of "$\prec$". Here $\varLambda$ is exactly the lattice we want to construct, and the corresponding vectors of $g_k(b',x',z^*,v)$ $(k = 0,1,2,\cdots,m)$ form the basis matrix $\mathcal{B}$ of our lattice $\varLambda$.

A simple example of $\mathcal{B}$ for $m = 5, \tau = 4, i = 3$ is shown in Table 7, where other non-zero off-diagonal entries are denoted by "$*$". Here we use the polynomial $g_3(b',x',z^*,v)$ in Table 7 to illustrate the above two replacements and the one-to-one correspondence between a polynomial and a row vector. According to the definition, we know

$$\begin{aligned}
g_3(b',x',z^*,v) &= E^{\min\{3,5-3\}}v^3(b'x')^{5-3}[f(b',x',z^*)]^3A^{\max\{4-3,0\}}\\
&= E^2v^3(b')^2(x')^2(-b'x'+z^*)^3A\\
&= (Eb'v)^2v(x')^2[-(b'x')^3+3(b'x')^2z^*-3b'x'(z^*)^2+(z^*)^3]A\\
&= (Eb'v)^2[-(b'v)(b')^2(x')^5+3(b'v)b'(x')^4z^*-3(b'v)(x')^3(z^*)^2\\
&\quad +v(x')^2(z^*)^3]A.
\end{aligned}$$

Replace each occurrence of the monomial $b'v$ by B, and each occurrence of EB by 1. Then we obtain

$$\begin{aligned}
g_3(b',x',z^*,v) &= 1^2\cdot[-B(b')^2(x')^5+3Bb'(x')^4z^*-3B(x')^3(z^*)^2+v(x')^2(z^*)^3]A\\
&= -AB(b')^2(x')^5+3ABb'(x')^4z^*-3AB(x')^3(z^*)^2+Av(x')^2(z^*)^3.
\end{aligned}$$

Finally, according to the definition of the monomial order "$\prec$", the corresponding coefficient vector of $g_3(B'b',X'x',Z^*z^*,Vv)$ is

$$(-AB(B')^2(X')^5,\ 3ABB'(X')^4Z^*,\ -3AB(X')^3(Z^*)^2,\ AV(X')^2(Z^*)^3,\ 0,\ 0).$$

Table 7. The basis matrix $\mathcal{B}$ when $m = 5, \tau = 4, i = 3$

	$(b')^2(x')^5$	$b'(x')^4z^*$	$(x')^3(z^*)^2$	$v(x')^2(z^*)^3$	$v^2x'(z^*)^4$	$v^3(z^*)^5$
g_0	$A^4(B')^2(X')^5$					
g_1	$*$	$A^3B'(X')^4Z^*$				
g_2	$*$	$*$	$A^2(X')^3(Z^*)^2$			
g_3	$*$	$*$	$*$	$AV(X')^2(Z^*)^3$		
g_4	$*$	$*$	$*$	$*$	$V^2X'(Z^*)^4$	
g_5	$*$	$*$	$*$	$*$	$*$	$V^3(Z^*)^5$

Let s denote the dimension of Λ, and we have $s = m+1$. From Equation (10) and the definition of Λ^*, one obtains $g(b_0', x_0', z_0^*, v_0) \equiv 0 \pmod{(a_0')^\tau}$ holds for any $g(b', x', z^*, v) \in \Lambda^*$. According to Lemma 2, if $\|g(B'b', X'x', Z^*z^*, Vv)\| < (a_0')^\tau/\sqrt{s}$ holds, we have $g(b_0', x_0', z_0^*, v_0) = 0$ holds over the integers.

Suppose such a polynomial $g(b', x', z^*, v) \in \Lambda^*$ with $g(b_0', x_0', z_0^*, v_0) = 0$ is obtained. Then we set $\widetilde{g}(b', x', z^*) := (b')^i g(b', x', z^*, B/b')$, where i is the parameter given before. One can check that every monomial (neglect the corresponding coefficient) of $\widetilde{g}(b', x', z^*)$ must have the form of $(b'x')^{m-j}(z^*)^j$. Let $\omega := (b'x')/z^*$ and we obtain $h(\omega) := \widetilde{g}(b', x', z^*)/(z^*)^m$ with $h((b_0'x_0')/z_0^*) = 0$. Then $(b_0'x_0')/z_0^*$ can be found by extracting the rational roots of $h(\omega)$ with classical methods. The values of $b_0'x_0', z_0^*$ are obtained since one can check $\gcd(b_0'x_0', z_0^*) = 1$. Thus $b_0'x_0' - z_0^*$ is known, and so is the value of $u_0/y_0' = A/(b_0'x_0' - z_0^*)$. Then we get the values of u_0, y_0' due to $\gcd(u_0, y_0') = 1$. Similarly, one can also obtain the values of v_0, x_0' according to $v_0/x_0' = B/(b_0'x_0')$ and $\gcd(v_0, x_0') = 1$. Hence the values of $z_0^*, u_0, y_0', v_0, x_0'$ are known, and finally we successfully obtain $(x_0, y_0, z_0) = (x_0'u_0, y_0'v_0, z_0^*u_0v_0)$.

As a conclusion, in order to find (x_0, y_0, z_0), we only need to find a polynomial $g(b', x', z^*, v)$ in Λ^* with the condition $\|g(B'b', X'x', Z^*z^*, Vv)\| < (a_0')^\tau/\sqrt{s}$. This is equivalent to finding a vector $\boldsymbol{g}$ in Λ with the condition $\|\boldsymbol{g}\| < (a_0')^\tau/\sqrt{s}$. According to Lemma 1 (take $i = 1$), by running LLL algorithm one can find a vector $\boldsymbol{g}$ in Λ with $\|\boldsymbol{g}\| \leqslant 2^{(s-1)/4}\det(\Lambda)^{1/s}$. From the above, to find (x_0, y_0, z_0), the following condition is sufficient:

$$2^{(s-1)/4}\det(\Lambda)^{1/s} < (a_0')^\tau/\sqrt{s}.$$

It is equivalent to $2^{s(s-1)/4}s^{s/2}\det(\Lambda) < [(a_0')^\tau]^s$. Note that researchers often ignore terms $2^{s(s-1)/4}s^{s/2}$. Thus, we obtain

$$\det(\Lambda) < [(a_0')^\tau]^s.$$

Define $\xi := \frac{\tau}{m}$, $\sigma := \frac{i}{m}$, and we only consider the case of $0 < \xi \leqslant 1, 0 \leqslant \sigma \leqslant 1$. Now one gets $\tau = \xi m$, $i = \sigma m$, which is used in the following calculation of $\det(\Lambda)$. As seen in Table 7, 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 $\det(\Lambda) = A^{s_A}(B')^{s_{B'}}V^{s_V}(X')^{s_{X'}}(Z^*)^{s_{Z^*}}$, and then we have

$$\begin{aligned}
s_A &= \textstyle\sum_{k=0}^{\tau}(\tau - k) = \frac{1}{2}\tau(\tau+1) = \frac{1}{2}\xi^2m^2 + o(m^2),\\
s_{B'} &= \textstyle\sum_{k=0}^{m-i}[(m-k) - i] = \frac{1}{2}(m-i)(m-i+1) = \frac{1}{2}(1-\sigma)^2m^2 + o(m^2),\\
s_V &= \textstyle\sum_{k=m-i}^{m}[i - (m-k)] = \frac{1}{2}i(i+1) = \frac{1}{2}\sigma^2m^2 + o(m^2),\\
s_{X'} &= \textstyle\sum_{k=0}^{m}(m-k) = \frac{1}{2}m(m+1) = \frac{1}{2}m^2 + o(m^2),\\
s_{Z^*} &= \textstyle\sum_{k=0}^{m}k = \frac{1}{2}m(m+1) = \frac{1}{2}m^2 + o(m^2).
\end{aligned}$$

Substitute the value of $\det(\Lambda)$ and $s = m+1$ in $\det(\Lambda) < [(a_0')^\tau]^s$, and we have

$$A^{\frac{1}{2}\xi^2+\frac{o(m^2)}{m^2}}(B')^{\frac{1}{2}(1-\sigma)^2+\frac{o(m^2)}{m^2}}V^{\frac{1}{2}\sigma^2+\frac{o(m^2)}{m^2}}(X')^{\frac{1}{2}+\frac{o(m^2)}{m^2}}(Z^*)^{\frac{1}{2}+\frac{o(m^2)}{m^2}} < (a_0')^{\xi+\frac{o(m^2)}{m^2}}.$$

Together with $A = M^{\alpha_2},\ B' = \lceil B/V \rceil \approx M^{\beta_2-\beta},\ V = M^{\beta},\ X' = \lceil X/U \rceil \approx M^{\alpha_1-\alpha},\ Z^* = \lceil Z/(UV) \rceil \approx M^{\gamma-\alpha-\beta},\ a_0' = A/u_0 \approx M^{\alpha_2-\alpha}$, it is obtained that

$$\alpha_2 \cdot \frac{1}{2}\xi^2 + (\beta_2-\beta)\cdot\frac{1}{2}(1-\sigma)^2 + \beta\cdot\frac{1}{2}\sigma^2 + (\alpha_1-\alpha)\cdot\frac{1}{2} + (\gamma-\alpha-\beta)\cdot\frac{1}{2} < (\alpha_2-\alpha)\cdot\xi + \frac{o(m^2)}{m^2}.$$

Take $m \to \infty$ and omit the term $\frac{o(m^2)}{m^2}$, then we have

$$[\frac{1}{2}\alpha_2\xi^2 - (\alpha_2-\alpha)\xi] + [\frac{1}{2}\beta_2\sigma^2 - (\beta_2-\beta)\sigma] + \frac{1}{2}(\gamma+\alpha_1+\beta_2) - (\alpha+\beta) < 0. \quad (11)$$

In order to minimize the left-hand side of Inequality (11), the optimized values of ξ and σ are given by $\xi = (\alpha_2-\alpha)/\alpha_2,\ \sigma = (\beta_2-\beta)/\beta_2$. After substituting $\xi = (\alpha_2-\alpha)/\alpha_2,\ \sigma = (\beta_2-\beta)/\beta_2$ in Inequality (11), we acquire

$$-\frac{1}{2}\cdot\frac{(\alpha_2-\alpha)^2}{\alpha_2} - \frac{1}{2}\cdot\frac{(\beta_2-\beta)^2}{\beta_2} + \frac{1}{2}(\gamma+\alpha_1+\beta_2) - (\alpha+\beta) < 0,$$

which finally ends up with Inequality (2). And thus we complete the proof of Theorem 1.

(II) In order to present the proof of Theorem 2, we need to prove the following lemma at first.

Lemma 7. *For Theorem 1, suppose $A = A_1A_2,\ B = B_1B_2$, where A_1, A_2, B_1, B_2 are known positive integers with $A_1 = M^{\alpha_{21}}, A_2 = M^{\alpha_{22}}, B_1 = M^{\beta_{21}}, B_2 = M^{\beta_{22}}$. And redefine $u_0 = \gcd(x_0, z_0, A_2) \approx U = M^{\alpha},\ v_0 = \gcd(y_0, z_0, B_2) \approx V = M^{\beta}$. Then the condition to find $(x, y, z) = (x_0, y_0, z_0)$ in Theorem 1, namely, Inequality (2), is changed to*

$$\gamma + \alpha_1 - \alpha_2\ (\approx \gamma + \beta_1 - \beta_2) < \frac{\alpha^2}{\alpha_{22}} + \frac{\beta^2}{\beta_{22}}. \quad (12)$$

Let us reset $x_0' := x_0/u_0,\ a_0' := A_2/u_0,\ y_0' := y_0/v_0,\ b_0' := B_2/v_0,\ z_0^* := z_0/(u_0v_0)$ and $f(b', x', z^*) := -B_1b'x' + z^* \in \mathbb{Z}[b', x', z^*]$. From

$$Bx_0 - Ay_0 = z_0 \Leftrightarrow B_1b_0'v_0x_0'u_0 - A_1a_0'u_0y_0'v_0 = z_0^*u_0v_0 \Leftrightarrow B_1b_0'x_0' - A_1a_0'y_0' = z_0^*,$$

one obtains

$$f(b_0', x_0', z_0^*) \equiv 0 \pmod{A_1a_0'}.$$

Again we also introduce another variable v for v_0. For $k = 0, 1, \cdots, m$, we redefine

$$g_k(b', x', z^*, v) := E^{\min\{i, m-k\}}v^i(b'x')^{m-k}[f(b', x', z^*)]^kA_1^{m-k}A_2^{\max\{\tau-k,0\}},$$

where m, τ, i are still three undetermined parameters, and E is the inverse of B_2 modulo $A_1^mA_2^\tau$. Then we obtain

$$g_k(b_0', x_0', z_0^*, v_0) \equiv 0 \pmod{A_1^m(a_0')^\tau}, \qquad k = 0, 1, \cdots, m.$$

Just as before, for every polynomial $g_k(b', x', z^*, v)$, we replace each occurrence of the monomial $b'v$ by B_2, and replace each occurrence of EB_2 by 1. Now the coefficient vectors of these $g_k(b'_0, x'_0, z^*_0, v_0)$ form a basis matrix $\mathcal{B}$ of the lattice Λ which we want to construct. Similar to the proof of Theorem 1, to find (x_0, y_0, z_0), we only need the condition $\det(\Lambda) < [A_1^m (a'_0)^\tau)]^s$, where s is the dimension of Λ. After substituting the calculation of $\det(\Lambda)$ and $s = m+1$ in $\det(\Lambda) < [A_1^m (a'_0)^\tau)]^s$, and taking the optimized values of τ/m and i/m as $m \to \infty$, finally we can get Inequality (12) and thus Lemma 7 follows.

(III) At last, we complete the proof of Theorem 2 as follows. Without loss of generality, one can assume $\gcd(C, A) = 1$. Thus there exist two integers D_1, D_2, such that $D_1 C = 1 + D_2 A$ and $0 < D_1 < A$, $0 < D_2 < C$. Then from $Bx_0 - Ay_0 = Cz_0$, we obtain $D_1 Bx_0 - D_1 Ay_0 = D_1 Cz_0 = (1 + D_2 A)z_0 = z_0 + D_2 Az_0$, which reduces to

$$B^* x_0 - Ay^*_0 = z_0, \quad B^* := D_1 B, \ y^*_0 := D_1 y_0 + D_2 z_0.$$

Set $d := \gcd(x_0, y^*_0)$. Then we have $d \mid z_0$, which implies $d \mid \gcd(x_0, z_0) = \gcd(x_0, z_0, A)$ and $d \mid (y^*_0 - D_2 z_0) = D_1 y_0$. Together with $\gcd(x_0, y_0) = 1$ and $\gcd(A, D_1) = 1$, we obtain $\gcd(x_0, y^*_0) = d = 1$. One can also get $\gcd(A, B^*) = 1$ due to $\gcd(A, D_1) = \gcd(A, B) = 1$. Similarly, we have $\gcd(y^*_0, z_0, B) = \gcd(z_0, B) = \gcd(y_0, z_0, B) = v_0$ and $\gcd(x_0, z_0, A) = u_0$. Finally, let us set $B^* = B^*_1 B^*_2$, $B^*_1 = D_1$, $B^*_2 = B$ and $A = A_1 A_2$, $A_1 = 1$, $A_2 = A$ for $B^* x_0 - Ay^*_0 = z_0$, and Theorem 2 follows according to Lemma 7.

Appendix D: Detailed Proof of Theorem 3

Firstly, let us suppose the conditions $\alpha_1 < \gamma < \alpha_2$ and $\alpha_1 + \alpha_2 \leqslant 2\gamma$ hold. Then under Assumption 1, we use Coppersmith's method to prove that Inequality (7) is sufficient to find the desired (x_0, y_0, z_0). Secondly, we prove that the condition $\alpha_1 + \alpha_2 \leqslant 2\gamma$ can be changed to the condition $4\alpha_1 + \alpha_2 \leqslant 4\gamma$. Finally, we note that $\alpha_1 < \gamma < \alpha_2$ is equivalent to $\alpha_1 < \alpha_2$ under the condition $4\alpha_1 + \alpha_2 \leqslant 4\gamma$ and Inequality (7). Similar to Appendix C, we may again give the same definitions for some notations introduced in the sketch of proof in Section 5.1.

(I) According to $z_0 \equiv C \pmod{x_0}$ and $z_0 \simeq z_0 - C$, we know there exists an integer w_0 such that $x_0 w_0 = z_0 - C$, and roughly we have $|w_0| < W := \lceil Z/X \rceil$ since $x_0 \approx X$. Here we introduce the variable w for w_0.

Define $f(x, z) := -Bx + z$. From $Bx_0 - Ay_0 = z_0$ we obtain

$$f(x_0, z_0) \equiv 0 \pmod{A}.$$

Let m, τ be positive integers, and then define

$$\begin{aligned} g_{t,j}(x, z) &:= x^{t-j} [f(x, z)]^j A^{m-j}, \quad j = 0, 1, \cdots, t, \quad t = 0, 1, \cdots, m, \\ h_{i,j}(x, z, w) &:= w^i [f(x, z)]^j A^{m-j}, \quad j = \theta_i, \theta_i + 1, \cdots, m, \quad i = 1, 2, \cdots, \tau, \end{aligned}$$

where $\theta_i := \lceil \eta i \rceil$, $\eta := \frac{\gamma - \alpha_1}{\alpha_2 - \gamma}$. The choice of θ_i is similar to [39], with the purpose of optimizing the lattice construction. It is obvious that

$$\begin{aligned} g_{t,j}(x_0, z_0) &\equiv 0 \pmod{A^m}, \quad j = 0, 1, \cdots, t, \quad t = 0, 1, \cdots, m, \\ h_{i,j}(x_0, z_0, w_0) &\equiv 0 \pmod{A^m}, \quad j = \theta_i, \theta_i + 1, \cdots, m, \quad i = 1, 2, \cdots, \tau. \end{aligned}$$

Besides, for every polynomial $h_{i,j}(x,z,w)$, we replace each occurrence of the monomial xw by $z-C$ according to the relation $x_0w_0 = z_0 - C$.

The monomial order "$\prec$" is defined such that (1) $x^{t_1-j_1}z^{j_1} \prec w^{i_2}z^{j_2}$ always holds; (2) $x^{t_1-j_1}z^{j_1} \prec x^{t_2-j_2}z^{j_2}$ if and only if $t_1 < t_2$ or $t_1 = t_2, j_1 < j_2$; (3) $w^{i_1}z^{j_1} \prec w^{i_2}z^{j_2}$ if and only if $i_1 < i_2$ or $i_1 = i_2, j_1 < j_2$. Similarly, we can define the polynomial order "$\prec^*$" for all the $g_{t,j}(x,z)$ and $h_{i,j}(x,z,w)$. Just as Section 3.2, the coefficient vectors of these $g_{t,j}(Xx,Zz)$ and $h_{i,j}(Xx,Zz,Ww)$ are determined according to "$\prec$", and these coefficient vectors form a basis matrix $\mathcal{B}$ of the lattice $\mathit{\Lambda}$ according to "$\prec^*$".

From the conditions $\alpha_1 < \gamma < \alpha_2$ and $\alpha_1 + \alpha_2 \leqslant 2\gamma$, one can obtain $\eta = \frac{\gamma-\alpha_1}{\alpha_2-\gamma} \geqslant 1$, which implies the condition $\theta_{i+1} \geqslant \theta_i + 1$. Here we note that it is the condition $\theta_{i+1} \geqslant \theta_i + 1$ that makes $\mathcal{B}$ a square matrix and a lower triangular matrix. One can refer to [39, Proposition 4.1] and its proof for details.

Let s denote the dimension of $\mathit{\Lambda}$. Similar to Section 3.2, combining Lemma 1 (take $i=2$) and Lemma 2, if

$$2^{s/4}\det(\mathit{\Lambda})^{1/(s-1)} < A^m/\sqrt{s}$$

holds, after running LLL algorithm one can obtain two polynomials $g_1(x,z,w)$, $g_2(x,z,w)$ satisfying $g_1(x_0,z_0,w_0) = 0$, $g_2(x_0,z_0,w_0) = 0$. According to the relation $z_0 = x_0w_0 + C$, we then set $\widetilde{g}_1(x,w) = g_1(x, xw+C, w)$, $\widetilde{g}_2(x,w) = g_2(x, xw+C, w)$. Next by computing resultants we can eliminate the variable w, namely, we obtain $h(x) = \mathrm{Res}_w[\widetilde{g}_1(x,w), \widetilde{g}_2(x,w)]$ satisfying $h(x_0) = 0$. If Assumption 1 holds, $h(x) \not\equiv 0$. Thus one can use any standard root-finding algorithm to recover $x_0 \in \mathbb{Z}^+$ from $h(x)$. Similarly, $w_0 \in \mathbb{Z}^+$ is also computed from $\widetilde{g}_1(x_0,w)$ or $\widetilde{g}_2(x_0,w)$. Then we get the values of $z_0 = x_0w_0 + C$ and $y_0 = (Bx_0 - z_0)/A$. As a conclusion, under Assumption 1, in order to find (x_0,y_0,z_0), we only need the condition $2^{s/4}\det(\mathit{\Lambda})^{1/(s-1)} < A^m/\sqrt{s}$, or roughly we only need the condition $\det(\mathit{\Lambda}) < (A^m)^{s-1}$ as most researchers do.

Define $\xi := \frac{\tau}{m}$, and we have $\tau = \xi m$. Together with $\theta_i = \lceil \eta i \rceil$, we can compute

$$\begin{aligned} s &= \textstyle\sum_{t=0}^{m}(t+1) + \sum_{i=1}^{\xi m}(m - \lceil \eta i \rceil + 1) \\ &= [\tfrac{1}{2}m^2 + o(m^2)] + [(\xi - \tfrac{1}{2}\eta\xi^2)m^2 + o(m^2)] \\ &= (-\tfrac{1}{2}\eta\xi^2 + \xi + \tfrac{1}{2})m^2 + o(m^2), \end{aligned}$$

$$\begin{aligned} \det(\mathit{\Lambda}) &= \textstyle\prod_{t=0}^{m}\prod_{j=0}^{t} X^{t-j}Z^jA^{m-j} \cdot \prod_{i=1}^{\xi m}\prod_{j=\lceil \eta i \rceil}^{m} W^iZ^jA^{m-j} \\ &= \textstyle\prod_{t=0}^{m}(XZA^{-1})^{\frac{1}{2}t(t+1)}A^{m(t+1)} \cdot \\ &\quad \textstyle\prod_{i=1}^{\xi m} W^{i(m-\lceil \eta i \rceil+1)}(ZA^{-1})^{\frac{1}{2}(m+\lceil \eta i \rceil)(m-\lceil \eta i \rceil+1)}A^{m(m-\lceil \eta i \rceil+1)} \\ &= (XZA^{-1})^{\frac{1}{6}m^3+o(m^3)}A^{\frac{1}{2}m^3+o(m^3)} \cdot \\ &\quad W^{(\frac{1}{2}\xi^2-\frac{1}{3}\eta\xi^3)m^3+o(m^3)}(ZA^{-1})^{(\frac{1}{2}\xi-\frac{1}{6}\eta^2\xi^3)m^3+o(m^3)}A^{(\xi-\frac{1}{2}\eta\xi^2)m^3+o(m^3)} \\ &= X^{\frac{1}{6}m^3+o(m^3)}Z^{(-\frac{1}{6}\eta^2\xi^3+\frac{1}{2}\xi+\frac{1}{6})m^3+o(m^3)}W^{(-\frac{1}{3}\eta\xi^3+\frac{1}{2}\xi^2)m^3+o(m^3)} \cdot \\ &\quad A^{(\frac{1}{6}\eta^2\xi^3-\frac{1}{2}\eta\xi^2+\frac{1}{2}\xi+\frac{1}{3})m^3+o(m^3)}. \end{aligned}$$

Substitute the calculation of $s, \det(\Lambda)$ in $\det(\Lambda) < (A^m)^{s-1}$, and we have

$$X^{\frac{1}{6}+\frac{o(m^3)}{m^3}} Z^{-\frac{1}{6}\eta^2\xi^3+\frac{1}{2}\xi+\frac{1}{6}+\frac{o(m^3)}{m^3}} W^{-\frac{1}{3}\eta\xi^3+\frac{1}{2}\xi^2+\frac{o(m^3)}{m^3}} A^{\frac{1}{6}\eta^2\xi^3-\frac{1}{2}\eta\xi^2+\frac{1}{2}\xi+\frac{1}{3}+\frac{o(m^3)}{m^3}}$$
$$< A^{-\frac{1}{2}\eta\xi^2+\xi+\frac{1}{2}+\frac{o(m^3)}{m^3}}.$$

Together with $X = M^{\alpha_1},\ Z = M^{\gamma},\ W = \lceil Z/X \rceil \approx M^{\gamma-\alpha_1},\ A = M^{\alpha_2}$, it is obtained that

$$\begin{aligned}&\alpha_1 \cdot \tfrac{1}{6} + \gamma \cdot (-\tfrac{1}{6}\eta^2\xi^3 + \tfrac{1}{2}\xi + \tfrac{1}{6}) + (\gamma - \alpha_1) \cdot (-\tfrac{1}{3}\eta\xi^3 + \tfrac{1}{2}\xi^2)\\ &< -\alpha_2 \cdot (\tfrac{1}{6}\eta^2\xi^3 - \tfrac{1}{2}\eta\xi^2 + \tfrac{1}{2}\xi + \tfrac{1}{3}) + \alpha_2 \cdot (-\tfrac{1}{2}\eta\xi^2 + \xi + \tfrac{1}{2}) + \tfrac{o(m^3)}{m^3}.\end{aligned}$$

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

$$[\frac{1}{6}(\alpha_2-\gamma)\eta^2-\frac{1}{3}(\gamma-\alpha_1)\eta]\xi^3+\frac{1}{2}(\gamma-\alpha_1)\xi^2-\frac{1}{2}(\alpha_2-\gamma)\xi+\frac{1}{6}(\gamma+\alpha_1-\alpha_2) < 0. \quad (13)$$

According to the definition of $h_{i,j}(x, z, w)$, we can obtain $\theta_\tau \leqslant m$. Recall that m is an integer, thus $\theta_\tau \leqslant m \Leftrightarrow \lceil \eta\tau \rceil \leqslant m \Leftrightarrow \eta\tau \leqslant m \Leftrightarrow \eta \cdot \xi m \leqslant m \Leftrightarrow \xi \leqslant \frac{1}{\eta}$. One can check that the left-hand side of Inequality (13) is a decreasing function of ξ for $\xi \in (0, \frac{1}{\eta}]$. In order to minimize the left-hand side of Inequality (13), we set $\xi = \frac{1}{\eta}$ in Inequality (13) and get

$$(\gamma - \alpha_1)\frac{1}{\eta^2} - 2(\alpha_2 - \gamma)\frac{1}{\eta} + (\gamma + \alpha_1 - \alpha_2) < 0,$$

which finally ends up with Inequality (7) after we substitute $\eta = \frac{\gamma-\alpha_1}{\alpha_2-\gamma}$. Thus we have proved that Inequality (7) is sufficient to find the desired (x_0, y_0, z_0) under the conditions $\alpha_1 < \gamma < \alpha_2$ and $\alpha_1 + \alpha_2 \leqslant 2\gamma$.

(II) Now let us change the condition $\alpha_1+\alpha_2 \leqslant 2\gamma$ to the condition $4\alpha_1+\alpha_2 \leqslant 4\gamma$. We redefine $\theta_i := \lceil \frac{1}{\xi} i \rceil = \lceil \frac{m}{\tau} i \rceil$ instead of $\theta_i := \lceil \eta i \rceil$. Then Inequality (13) is changed to

$$[\frac{1}{6}(\alpha_2-\gamma)\frac{1}{\xi^2}-\frac{1}{3}(\gamma-\alpha_1)\frac{1}{\xi}]\xi^3+\frac{1}{2}(\gamma-\alpha_1)\xi^2-\frac{1}{2}(\alpha_2-\gamma)\xi+\frac{1}{6}(\gamma+\alpha_1-\alpha_2) < 0,$$

which is equivalent to

$$(\gamma - \alpha_1)\xi^2 - 2(\alpha_2 - \gamma)\xi + (\gamma + \alpha_1 - \alpha_2) < 0. \quad (14)$$

This time we reset $\xi = \frac{\sqrt{\alpha_2}-\sqrt{\gamma-\alpha_1}}{\sqrt{\gamma-\alpha_1}}$ in Inequality (14), and finally under the condition $\alpha_1 < \gamma < \alpha_2$, one can also obtain Inequality (7) after some reduction.

We still need the condition $\theta_{i+1} \geqslant \theta_i + 1$ which makes $\mathcal{B}$ a square matrix and a lower triangular matrix. And the condition $\frac{1}{\xi} \geqslant 1$ is sufficient. Under the condition $\alpha_1 < \gamma < \alpha_2$, one can check that $\frac{1}{\xi} = \frac{\sqrt{\gamma-\alpha_1}}{\sqrt{\alpha_2}-\sqrt{\gamma-\alpha_1}} \geqslant 1$ is equivalent to $4\alpha_1 + \alpha_2 \leqslant 4\gamma$, which replaces the previous condition $\alpha_1 + \alpha_2 \leqslant 2\gamma$.

Originally, we set $\xi = \frac{1}{\eta} = \frac{\alpha_2-\gamma}{\gamma-\alpha_1}$. Note that Inequality (7) is equivalent to $\alpha_2 > \alpha_1 + \sqrt{\alpha_2(\gamma - \alpha_1)}$ under the condition $\alpha_1 < \gamma < \alpha_2$. Then one knows

$\frac{\alpha_2-\gamma}{\gamma-\alpha_1} > \frac{\alpha_1+\sqrt{\alpha_2(\gamma-\alpha_1)}-\gamma}{\gamma-\alpha_1} = \frac{\sqrt{\alpha_2}-\sqrt{\gamma-\alpha_1}}{\sqrt{\gamma-\alpha_1}}$. After we reset $\xi = \frac{\sqrt{\alpha_2}-\sqrt{\gamma-\alpha_1}}{\sqrt{\gamma-\alpha_1}}$, the condition $\frac{1}{\xi} \geqslant 1$ will hold more easily.

(III) At last, we note that $\alpha_1 < \gamma < \alpha_2$ is equivalent to $\alpha_1 < \alpha_2$ under the condition $4\alpha_1 + \alpha_2 \leqslant 4\gamma$ and Inequality (7). This is because one can obtain $\alpha_1 < \gamma$ from $4\alpha_1 + \alpha_2 \leqslant 4\gamma$, and obtain $\gamma < \alpha_2$ from $\alpha_1 < \alpha_2$ and Inequality (7).

From the above, we know Theorem 3 follows.