A New Factoring Attack on Multi-Prime RSA with Small Prime Difference

Mengce Zheng and Honggang Hu

Key Laboratory of Electromagnetic Space Information, Chinese Academy of Sciences,
School of Information Science and Technology, University of Science and Technology of China, Hefei, China
mczheng@mail.ustc.edu.cn; hghu2005@ustc.edu.cn

Abstract

In this paper, we study the security of multi-prime RSA whose modulus is $N = p_1 p_2 \cdots p_r$ for $r \geq 3$ with small prime difference of size N^γ. In ACISP 2013, Zhang and Takagi showed a Fermat-like factoring attack, which can directly factor N for $\gamma < \frac{1}{r^2}$. We improve this bound to theoretically achieve $\gamma < \frac{2}{r(r+2)}$ by a new factoring attack. Furthermore, we also analyse specific MPRSA with imbalanced prime factors. Experimental results are provided to show the efficiency of our attack.

Keywords: Factoring attack, multi-prime RSA, small prime difference, cryptanalysis, lattice.

1 Introduction

1.1 Background

RSA [24] is a famous public key cryptosystem that has been widely used in various settings. However, the original RSA is not fit for constrained environments. Since people need faster and more efficient RSA encryption/decryption processes, several fast variants have been proposed and surveyed [4]. In this paper, we focus on multi-prime RSA (MPRSA) and some related attacks (mainly factoring attack with small prime difference) on MPRSA.

The MPRSA variant is based on modifying the RSA modulus to $N = p_1 p_2 \cdots p_r$ for $r \geq 3$. It is first patented by Compaq [6], using a modulus of the form $N = p_1 p_2 p_3$. We describe its key generation, encryption, and decryption algorithms and also the performance of MPRSA.

Algorithm 1 (Key Generation)

INPUT: *A security parameter n and a size parameter r.*

1: Generate r distinct primes $p_1, p_2, \ldots, p_r$ of n/r bit-size and set the modulus $N = \prod_{i=1}^{r} p_i$.

2: Pick a random number that is relatively coprime to $\varphi(N) = \prod_{i=1}^{r}(p_i - 1)$ as the public key e (e.g. $e = 65537$) and compute the private key $d = e^{-1} \bmod \varphi(N)$.

OUTPUT: *An RSA public/private key pair.*

Algorithm 2 (Encryption)

INPUT: *A message M and an RSA public key (N, e).*

1: Transform the string M into an integer $M \in \mathbb{Z}_N$ and compute the ciphertext as $C = M^e \bmod N$.

OUTPUT: *The corresponding ciphertext C.*

Algorithm 3 (Decryption with CRT)

INPUT: *A ciphertext C and an RSA private key d.*

1: Let $d_i \equiv d \bmod p_i - 1$ amd compute $M_i = C^{d_i} \bmod p_i$ for each i, $1 \leq i \leq r$.

2: Combine M_i's by the Chinese Remainder Theorem (CRT) to obtain the corresponding plaintext $M = C^d \bmod N$.

OUTPUT: *The corresponding message M.*

Throughout previous analysis, there are the following assumptions on primes of MPRSA modulus $N = p_1 p_2 \cdots p_r$ for $r \geq 3$.

1. $p_1 < p_2 < \cdots < p_r$,
2. $\frac{1}{2} N^{\frac{1}{r}} < p_1 < N^{\frac{1}{r}} < p_r < 2N^{\frac{1}{r}}$.

This second assumption indicates that all the primes are balanced. That is, all primes $p_1, p_2, \ldots, p_r$ are roughly of the same bit-size. The small prime difference Δ of $p_1, p_2, \ldots, p_r$ is defined as $\Delta = \max_{i \neq j} |p_i - p_j| = p_r - p_1 = N^\gamma$ where $0 < \gamma < \frac{1}{r}$.

The advantage of MPRSA is its efficiency when using Chinese Remainder Theorem (CRT) in its decryption process. From [4], we know that the speedup of MPRSA with $N = p_1 p_2 \cdots p_r$ for $r \geq 3$ over the standard RSA is approximately $\frac{r^2}{4}$. Moreover, several attacks (such as small private exponent attack, partial key exposure attack etc.) are less effective as r increases. But r should not be unrestrictedly large because of the Elliptic Curve Method (ECM) [21]. Since factoring an MPRSA modulus using ECM (i.e., 256-bit prime factors are considered within the factoring bound of ECM) is much easier with increasing r, one should choose $r = 3, 4$ or 5 for most MPRSA settings. Generally speaking, MPRSA might be a practical alternative for reducing the decryption costs.

1.2 Related Work

Suppose that N is an MPRSA modulus with r balanced primes $p_1, p_2, \ldots, p_r$. Let $e \approx N$ be a valid public exponent and $d = N^\delta$ be its corresponding private exponent. Many researchers have investigated the security of MPRSA for small private exponent [5, 13, 16, 14, 15, 26] and small prime difference [1, 27, 28, 26]. Below we review some previous results and point out the existing drawbacks. Since lattice method is always better than continued fraction approach, we just provide the results using lattice method.

Ciet *et al.*'s Atttack [5]

Given the public key (N, e), then d can be recovered in time polynomial in $\log N$ if

$$\delta < \frac{4}{3} - \frac{1}{3r} - \frac{2}{3r}\sqrt{4r^2 - 5r + 1}.$$

This attack is reduced to the Small Inverse Problem (SIP), namely $k(A + s) + 1 \equiv 0 \bmod e$, which is proposed by Boneh and Durfee [3]. In general, $|k|$ and $|s|$ are bounded by e^δ and $N^{1-\frac{1}{r}}$, respectively. Ciet *et al.* applied the original Boneh-Durfee lattice construction in [3] and obtained above result.

Hinek *et al.*'s Attack [13, 16, 15]

Given the public key (N, e), then d can be recovered in time polynomial in $\log N$ if

$$\delta < \frac{6}{5r} - \frac{4}{5} + \frac{2}{5r}\sqrt{4r^2 - 7r + 4},$$
$$\delta < 1 - \sqrt{1 - \frac{1}{r}}.$$

Hinek *et al.* [16] applied an extension of the Boneh-Durfee lattice proposed by Blömer and May [2] and provided the first improved bound. Later, the second improved bound is obtained by stronger lattice construction stated in [3].

Zhang and Takagi's Attack [27, 28]

Let $\Delta = p_r - p_1 = N^\gamma$, $0 < \gamma < \frac{1}{r}$ be small prime difference of the prime factors of N. Given the public key (N, e), then d can be probabilistically found in time polynomial in $\log N$, if γ and δ satisfy

$$\delta < 1 - \sqrt{1 + \gamma - \frac{2}{r}}.$$

Zhang and Takagi presented it in [27] by bounding $|k| \leq e^\delta$ and $|s| \leq N^{1+\gamma-\frac{2}{r}}$. Later, they [28] improved the result by applying a tighter bound for $|s|$.

$$\delta < 1 - \sqrt{1 + 2\gamma - \frac{3}{r}} \quad \text{for} \quad \gamma \geq \frac{3}{2r} - \frac{1+\delta}{4},$$
$$\delta < \frac{3}{r} - \frac{1}{4} - 2\gamma \quad \text{for} \quad \gamma < \frac{3}{2r} - \frac{1+\delta}{4}.$$

We note that for the second case, there exists a better factoring attack for quite small γ. The advantage is that factoring attack does not require any restriction on δ.

Takayasu and Kunihiro's Attack [26]

Given the public key (N, e), then d can be probabilistically found in time polynomial in $\log N$, if γ and δ satisfy

$$\delta < 1 - \sqrt{1 + 2\gamma - \frac{3}{r}} \quad \text{for} \quad \frac{3}{2}\left(\frac{1}{r} - \frac{1}{4}\right) \leq \gamma < \frac{1}{r},$$

$$\delta < 1 - \frac{2}{3}\left(\sqrt{(7 + 8\gamma - \frac{12}{r})(1 + 2\gamma - \frac{3}{r})} - 1 - 2\gamma + \frac{3}{r}\right) \quad \text{for} \quad 0 < \gamma < \frac{3}{2}\left(\frac{1}{r} - \frac{1}{4}\right),$$

Takayasu and Kunihiro summarized previous lattice-based methods to provided an improved lattice construction for solving SIP. It covers broader family of lattice construction and previous results. Due to this improvement, they also presented the insecure situation of MPRSA with small prime difference.

We note that only for $r = 3$, the second condition $\frac{3}{2}(\frac{1}{r} - \frac{1}{4})$ makes sense. While $r \geq 4$, the latter vanishes and it degenerates to $\delta < 1 - \sqrt{1 + 2\gamma - \frac{3}{r}}$ for $\gamma < \frac{1}{r}$. Hence, there exists factoring attack for quite small γ without any restriction on δ as well.

Remark 1 Boneh and Durfee [3] have noted that solving SIP is heuristic because the polynomials derived from lattice reduction algorithm are not guaranteed to be algebraically independent. If it is not the case, the private exponent d or the factorization of N cannot be recovered. Thus, we provide an assumption for algebraically independent polynomials.

Assumption 1 (Algebraic Independence) *The polynomials derived from lattice reduction algorithm (e.g. LLL algorithm) in our lattice-based method are algebraically independent.*

1.3 Our Contributions

In this paper, we study the factoring attack on MPRSA with small prime difference. Small prime difference was first introduced by de Weger [12] to improve the bound for solving SIP. In ACISP 2013, Zhang and Takagi [27] presented a Fermat-like factoring attack on MPRSA, which can directly lead to the factorization of N for $\gamma < \frac{1}{r^2}$. However, their work just utilizes partial advantage of the balanced primes. In our method, we show an improvement on factoring attack from $\gamma < \frac{1}{r^2}$ to $\gamma < \frac{2}{r(r+2)}$. Contrast to previous, we use the knowledge of all balanced primes to solve an r-variate integer polynomial.

Furthermore, we consider specific MPRSA with imbalanced primes. Since the prime factors of the modulus are imbalanced, small private exponent attacks do not work any more and factoring attack becomes more important. In this paper, we consider the imbalanced MPRSA with the following assumptions on primes of the modulus $N = p_1 p_2 \cdots p_r$ for $r \geq 3$.

1. $p_1 < p_2 < \cdots < p_r$,
2. $\frac{1}{2} N^{\frac{1-\beta}{r-1}} < p_1 < N^{\frac{1-\beta}{r-1}} < p_{r-1} < 2N^{\frac{1-\beta}{r-1}}$,
3. $p_r = N^\beta$ for $\frac{1}{r} < \beta < 1$.

This third assumption indicates that all primes are imbalanced while the second indicates that the smaller $r-1$ primes are still balanced. We show that given the public key (N, e) of an imbalanced MPRSA with $p_r = N^\beta$, then N can be probabilistically factored in time polynomial in $\log N$, if γ satisfies

$$\gamma < \frac{2(1-\beta)(r-2+\beta)}{(r-1)(r^2+2\beta-3)}.$$

1.4 Organizations

In Sect. 2, we introduce lattice-based methods to solve modular and integer equations. In Sect. 3, we present our improved factoring attack on balanced MPRSA with small prime difference. In Sect. 4, we further describe factoring attack on MPRSA with imbalanced primes. In Sect. 5, we analyse the performance of our attacks by experiments and comparisons.

2 Preliminaries

In this section, we briefly introduce the LLL algorithm [20] and Coppersmith's technique [8, 7, 9] (also referred as Howgrave-Graham's lemma [17] and Coron's reformulation [10, 11]). One can refer to [22, 23] for more details.

The LLL algorithm proposed by Lenstra, Lenstra and Lovász [20] is practically used for finding approximately small lattice vectors due to its efficient polynomial-time running results. We provide the following substratal lemma for our method.

Lemma 1 (LLL [20]) *Let $\mathcal{L}$ be a lattice spanned by a basis $(\vec{b}_1, \vec{b}_2, \ldots, \vec{b}_m)$. The LLL algorithm outputs a reduced basis $(\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_m)$ of $\mathcal{L}$ in polynomial time, that satisfies*

$$\|\vec{v}_1\|, \|\vec{v}_2\|, \ldots, \|\vec{v}_i\| \leq 2^{\frac{m(m-1)}{4(m+1-i)}} \det(\mathcal{L})^{\frac{1}{m+1-i}},$$

for reduced basis vectors $\vec{v}_i$, $1 \leq i \leq m$.

The main idea of Coppersmith's technique is to transform finding small roots of a modular equation (or an integer equation) to extracting roots over the integers. To do so, one can collect polynomials sharing a common root modulo R^m for some well-chosen integer R and m. Then one can apply the polynomials' coefficient vectors to construct a lattice basis. Using lattice reduction algorithms (i.e., the

LLL algorithm [20]), one can obtain a collection of equations over the integers with sufficiently small norm. Thus, one finally solve the desired root.

The following lemma presented by Howgrave-Graham [17] gives a criterion for judging whether the desired root of a modular equation is also a root over $\mathbb{Z}$. To a given polynomial $g(x_1, \ldots, x_n) = \sum a_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$, its norm is defined by $\|g(x_1, \ldots, x_n)\|^2 := \sum |a_{i_1,\ldots,i_n}|^2$.

Lemma 2 (Howgrave-Graham [17]) *Let $g(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ be an integer polynomial that is a sum of at most m monomials. Suppose that*

1. $g(x_1^{(0)}, \ldots, x_n^{(0)}) \equiv 0 \pmod{R}$, *where* $|x_1^{(0)}| < X_1, \ldots, |x_n^{(0)}| < X_n$,
2. $\|g(x_1 X_1, \ldots, x_n X_n)\| < \frac{R}{\sqrt{m}}$.

Then we have $g(x_1^{(0)}, \ldots, x_n^{(0)}) = 0$ over the integers.

Below we introduce a useful lemma for solving a linear equation modulo an unknown divisor of the modulus N by combining Lemma 1 and 2. This lemma directly leads to previous factoring attack.

Lemma 3 (Coppersmith [9]/May [23]) *Let N be an integer of unknown factorization, which has an unknown divisor $p \geq N^\tau$, $0 < \tau \leq 1$. Let $f_p(x)$ be a univariate monic polynomial of degree n and $\epsilon > 0$. Then for a sufficiently large N, we can find all solutions $x^{(0)}$ in time $O(\epsilon^{-7} n^5 \log^9 N)$ for the equation*

$$f_p(x) \equiv 0 \bmod p \quad \textit{for} \quad |x^{(0)}| \leq N^{\frac{\tau^2}{n} - \epsilon}.$$

In our method, we want to solve an r-variate integer polynomial by the following lemma [18]. First, let $X_i = N^{\eta_i}$ for positive integer N and real positive number η_i for $i = 1, 2, \ldots, r$. Then, we also define $\|f(\cdot)\|_\infty$ as the largest coefficient (in absolute value form) of all the monomials in polynomial $f(\cdot)$.

Lemma 4 (Jochemsz [18]) *Given the polynomial*

$$f(x_1, x_2, \ldots, x_r) = \prod_{i=1}^{r} (x_i + p) - N \in \mathbb{Z}[x_1, x_2, \ldots, x_r].$$

For any $\epsilon > 0$, the root $(x_1^{(0)}, x_2^{(0)}, \ldots, x_r^{(0)})$ satisfying $|x_i^{(0)}| < X_i$ for such bounds X_i ($i = 1, 2, \ldots, r$) can be discovered if N is sufficiently large and

$$X_1 X_2 \cdots X_r < W^{\frac{2}{r+1} - \epsilon},$$

where $W = \|f(x_1 X_1, x_2 X_2, \ldots, x_r X_r)\|_\infty$. The running time is polynomial in $\log N$ and $1/\epsilon$.

One can see [18] for detailed analysis. The first special case was showed by Coppersmith in [9] for finding the bound $XY < W^{\frac{2}{3}}$ of the polynomial $f(x, y) = (p_0 + x)(q_0 + y) - N$ to factor the modulus N with known bits of prime factors p, q. Later, Jochemsz [19, 18] provided a generalized strategy for finding roots of such polynomials.

3 Improved Factoring Attack with Balanced Primes

For an MPRSA instance with a balanced modulus $N = p_1 p_2 \cdots p_r$ for $r \geq 3$, where $p_1 < p_2 < \cdots < p_r$. Define the small prime difference of $p_1, p_2, \ldots, p_r$ by $\Delta = p_r - p_1 = N^\gamma, 0 < \gamma < \frac{1}{r}$. We know that there exists $|p_i - p| < p_r - p_1 = N^\gamma$ for $p = \lfloor N^{\frac{1}{r}} \rfloor$. Thus, let $x_i = p_i - p$ for $i = 1, 2, \ldots, r$ and define a monic linear modular polynomial $f_{p_i}(x) = x + p \bmod p_i$, which has a root x_i modulo p_i. According to Lemma 3, we can efficiently find every x_i by solving r many linear modular equations $f_{p_i}(x) \equiv 0 \bmod p_i$ if γ is small enough. We show the following proposition for this factoring attack.

Proposition 1 (Zhang and Takagi [27]) *Let $N = p_1 \cdots p_r$ be an MPRSA balanced modulus, where $p_1 < p_2 < \cdots < p_r$, $p_r - p_1 = N^\gamma$, $0 < \gamma < \frac{1}{r}$. If $\gamma < \frac{1}{r^2}$, then N can be factored in time polynomial in* $\log N$.

The proof is straightforward when applying Lemma 3 to each $f_{p_i}(x_i) = x_i + p \equiv 0 \bmod p_i$ for $\tau = \frac{1}{r}$. Opposite to previous method of making use of $f_{p_i}(x_i)$ separately, we gather them together to solve an r-variate integer polynomial. More concretely, we present the following factoring attack.

Attack 1 *Let $N = p_1 \cdots p_r$ be an MPRSA balanced modulus, where $p_1 < p_2 < \cdots < p_r$, $p_r - p_1 = N^\gamma$, $0 < \gamma < \frac{1}{r}$. If $\gamma < \frac{2}{r(r+2)}$, then N can be factored in time polynomial in* $\log N$.

Notice that $f_{p_i}(x_i) = x_i + p = p_i$. We have

$$f(x_1, x_2, \ldots, x_r) = \prod_{i=1}^{r}(x_i + p) - N = \prod_{i=1}^{r} f_{p_i}(x_i) - N = \prod_{i=1}^{r} p_i - N = 0.$$

Before we apply Lemma 4 to above polynomial, we must figure out η_i for $i = 1, \ldots, r$ and W. It is clear that $\eta_i = \gamma$ since $|x_i| = |p_i - p| < p_r - p_1 = N^\gamma$. However, it may be a little complicated for W. We roughly have $W = \max\{N - p^r, p^{r-1}N^\gamma\}$ by the definition. Since all primes have a small difference N^γ, N and p^r differ from each other in $N^{\gamma r}$ least significant bits. Hence, we have

$$W = \max\{N^{\gamma r}, N^{\frac{r-1}{r} + \gamma}\} = N^{\frac{r-1}{r} + \gamma}.$$

It can be easily inferred because

$$\gamma < \frac{1}{r} \Leftrightarrow \gamma(r-1) < \frac{r-1}{r} \Leftrightarrow \gamma r < \frac{r-1}{r} + \gamma.$$

From Lemma 4, the condition reduces to (we omit the tiny term ϵ therein)

$$\gamma r < \frac{2}{r+1}\left(\frac{r-1}{r}+\gamma\right) \Leftrightarrow \gamma\left(\frac{r(r+1)}{2}-1\right) < \frac{r-1}{r} \Leftrightarrow \gamma < \frac{2}{r(r+2)}.$$

Thus, the final condition is

$$\gamma < \frac{2}{r(r+2)}.$$

Figure. 1 shows that our bound is more optimized. Experimental results will be given in Sect. 5.

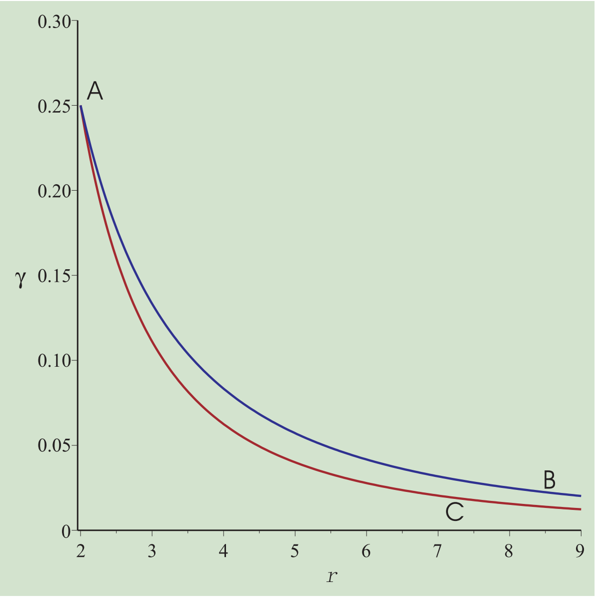

Figure 1: Curve AB and AC represent $\gamma = \frac{2}{r(r+2)}$ and $\gamma = \frac{1}{r^2}$, respectively

Next we provide concrete lattice construction for solving this polynomial $f(x_1, x_2, \ldots, x_r)$. Define two sets S and S_R for a positive integers s.

$$S = \bigcup \left\{x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} | x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \text{ is a monomial of } f^{s-1}\right\},$$

$$S_R = \bigcup \left\{x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} | x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \text{ is a monomial of } f^{s}\right\}.$$

By calculating the expansion of f^{s-1} (or f^s), we know the relation of every element in S (or S_R) to its exponent i_j for $j = 1, 2, \ldots, r$.

$$x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \in S \leftrightarrow i_j = 0, \ldots, s-1, \text{ for } j = 1, 2, \ldots, r,$$

$$x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \in S_R \leftrightarrow i_j = 0, \ldots, s, \text{ for } j = 1, 2, \ldots, r.$$

For $R = W\prod_{i=1}^{r} X_i^{s-1} = N^{\frac{r-1}{r}+\gamma+\gamma r(s-1)}$, we then define $f' = (p^r - N)^{-1} f \bmod R$ and shift polynomials below,

$$g : x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} f' \cdot \frac{R}{W\prod_{j=1}^{r} X_j^{i_j}}, \text{ for } x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \in S,$$

$$g' : x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \cdot R, \text{ for } x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \in S_R \backslash S.$$

Notice that above shift polynomials g and g' modulo R are equal to zero. Afterwards, we use the LLL algorithm to search several integer linear combinations of g and g', whose norm is ensured to be sufficiently small. (This has been mentioned in Sect. 2.) The lattice $\mathcal{L}$ is constructed by the coefficient vectors of g and g' by substituting $x_i X_i$ for each x_i. It is always represented by a square basis matrix whose rows are corresponding vectors.

Before showing an example of such a basis matrix, we first define the monomial order $\prec$ in our method as $x_1^{i_1} x_2^{i_2} \cdots x_r^{i_r} \prec x_1^{j_1} x_2^{j_2} \cdots x_r^{j_r}$ if $i_1 + i_2 + \cdots + i_r < j_1 + j_2 + \cdots + j_r$ or $\sum_{k=1}^{r} i_k = \sum_{k=1}^{r} j_k$, $\sum_{k=1}^{t} i_k > \sum_{k=1}^{t} j_k$ for $t = 1, 2, \ldots, r-1$. Then a simple example is showed in Table 1, where non-zero off-diagonal entries are marked by $-$.

Table 1: A simple example with $s = 1$ and $r = 3$

	1	x_1	x_2	x_3	x_1x_2	x_1x_3	x_2x_3	$x_1x_2x_3$
$g_{0,0,0}$	1	$-$	$-$	$-$	$-$	$-$	$-$	$-$
$g'_{1,0,0}$		RX_1						
$g'_{0,1,0}$			RX_2					
$g'_{0,0,1}$				RX_3				
$g'_{1,1,0}$					RX_1X_2			
$g'_{1,0,1}$						RX_1X_3		
$g'_{0,1,1}$							RX_2X_3	
$g'_{1,1,1}$								$RX_1X_2X_3$

When $\gamma < \frac{2}{r(r+2)}$ and a suitable s is chosen, we can obtain $r-1$ many polynomials $f_1, f_2, \ldots, f_{r-1}$ apart from f. Moreover, they share a common root $(p_1 - p, p_2 - p, \ldots, p_r - p)$ over the integers. We can solve p_i for $1 \le i \le r$ under Assumption 1, which directly lead to the factorization of N.

The running time depends on reducing the basis matrix and extracting the common root. Both of them can be done in polynomial time. The LLL algorithm can output the desired polynomials in time polynomial in $\log N$. The Gröbner basis computation for finding the common root is often polynomial time computable in practice. Furthermore, we assume that the running time of the Gröbner basis computation is negligible compared to the LLL algorithm. Additionally, one could have more polynomials than required amount after the LLL algorithm. Hence, we usually use the Gröbner basis computation rather than resultant computation.

Remark 2 Though we obtain a broader bound on small prime difference, there exist some disadvantages like success rate and lattice dimension. In Sect. 5, we will implement experiments and succeed to extract our desired roots in practice. From our lattice construction, the dimension can be calculated as $(s+1)^r$, which is exponential in r. Fortunately, r is usually set $3, 4$ and 5. When s is also fixed small (i.e., 1,2), our method is still efficient.

4 Factoring Attack with Imbalanced Primes

In this section, we further analyze factoring attack on MPRSA with imbalanced primes. The reason why we need MPRSA with imbalanced primes is that one cannot efficiently perform previous small private exponent attacks on imbalanced MPRSA. Below we recall our assumptions on imbalanced MPRSA stated in Sect. 1.

We consider the imbalanced MPRSA with the following assumptions on the primes of the modulus $N = p_1 p_2 \cdots p_r$ for $r \geq 3$.

1. $p_1 < p_2 < \cdots < p_r$,
2. $\frac{1}{2}N^{\frac{1-\beta}{r-1}} < p_1 < N^{\frac{1-\beta}{r-1}} < p_{r-1} < 2N^{\frac{1-\beta}{r-1}}$,
3. $p_r = N^{\beta}$ for $\frac{1}{r} < \beta < 1$.

The small prime difference Δ of $p_1, p_2, \ldots, p_{r-1}$ is defined as $\Delta = p_{r-1} - p_1 = N^{\gamma}$ where $0 < \gamma < \frac{1-\beta}{r-1}$. Another small prime difference $\bar{\Delta}$ of p_r and N^{β} is defined as $\bar{\Delta} = |p_r - N^{\beta}| = N^{\bar{\gamma}}$ where $0 < \bar{\gamma} < \beta$. In fact, $\bar{\gamma}$ can be determined with high probability.

For this imbalanced MPRSA, one can find $r-1$ balanced primes by Lemma 3 provided

$$\gamma < \left(\frac{1-\beta}{r-1}\right)^2.$$

This knowledge is enough to factor the modulus N. Moreover, we can perform similar factoring attack on imbalanced MPRSA, which is analogous to Attakc 1 in Sect. 3.

Attack 2 *Let $N = p_1 \cdots p_r$ be an MPRSA imbalanced modulus, where $p_1 < p_2 < \cdots < p_r$, $p_{r-1} - p_1 = N^{\gamma}$, $0 < \gamma < \frac{1-\beta}{r-1}$ and $p_r = N^{\beta}$, $\frac{1}{r} < \beta < 1$. If $\gamma < \frac{2(1-\beta)(r-2+\beta)}{(r-1)(r^2+2\beta-3)}$, then N can be factored in time polynomial in* $\log N$.

Similarly, we have

$$\bar{f}(x_1, x_2, \dots, x_r) = (x_r + \bar{p}) \prod_{i=1}^{r-1} (x_i + p) - N = \prod_{i=1}^{r} p_i - N = 0,$$

where $p = \lfloor N^{\frac{1-\beta}{r-1}} \rfloor, \bar{p} = \lfloor \frac{N}{p^{r-1}} \rfloor, x_r = p_r - \bar{p}$ and $x_i = p_i - p$ for $i = 1, 2, \dots, r-1$.

We also should figure out η_i for $i = 1, \dots, r$ and W stated in Lemma 4. Since we know that $\eta_i = \gamma$ for $1 \leq i \leq r-1$, we have $\eta_r = \bar{\gamma} = \frac{\beta\gamma(r-1)}{1-\beta}$. We roughly have $W = \max\{N - \bar{p}p^{r-1}, p^{r-1}N^{\bar{\gamma}}, \bar{p}p^{r-2}N^{\gamma}\}$ for W. From our assumptions, N and $\bar{p}p^{r-1}$ differ from each other in $N^{\frac{\gamma(r-1)}{1-\beta}}$ least significant bits. Hence, we have

$$W = \max\{N^{\frac{\gamma(r-1)}{1-\beta}}, N^{1-\beta+\frac{\beta\gamma(r-1)}{1-\beta}}, N^{\beta+\gamma+\frac{(r-2)(1-\beta)}{r-1}}\} = N^{\beta+\gamma+\frac{(r-2)(1-\beta)}{r-1}}.$$

It can be inferred by

$$\gamma < \frac{1-\beta}{r-1} \Leftrightarrow \gamma(r-1) < 1-\beta \Leftrightarrow \frac{\gamma(r-1)}{1-\beta} < 1-\beta + \frac{\beta\gamma(r-1)}{1-\beta},$$

and

$$\begin{aligned} & 1-\beta + \frac{\beta\gamma(r-1)}{1-\beta} < \beta + \gamma + \frac{(r-2)(1-\beta)}{r-1} \\ \Leftrightarrow & \frac{1-\beta}{r-1} + \frac{\beta\gamma(r-1)}{1-\beta} < \beta + \gamma \\ \Leftrightarrow & \frac{\gamma(\beta r - 1)}{1-\beta} < \frac{\beta r - 1}{r-1} \quad (\frac{1}{r} < \beta \Leftrightarrow \beta r - 1 > 0) \\ \Leftrightarrow & \gamma < \frac{1-\beta}{r-1}. \end{aligned}$$

Thus, the condition reduces to (we omit the tiny term ϵ therein)

$$\gamma(r-1) + \frac{\beta\gamma(r-1)}{1-\beta} < \frac{2}{r+1}\left(\beta + \gamma + \frac{(r-2)(1-\beta)}{r-1}\right) \Leftrightarrow \gamma < \frac{2(1-\beta)(r-2+\beta)}{(r-1)(r^2+2\beta-3)}.$$

Eventually, the condition is

$$\gamma < \frac{2(1-\beta)(r-2+\beta)}{(r-1)(r^2+2\beta-3)}.$$

Our bounds for various r and $\beta = 0.4, 0.5$ are showed in Figure. 2.

Remark 3 When β is exactly equal to $\frac{1}{r}$, this bound $\gamma < \frac{2(1-\beta)(r-2+\beta)}{(r-1)(r^2+2\beta-3)}$ reduces to $\gamma < \frac{2}{r(r+2)}$. It means that factoring attack with balanced primes can be viewed as a special case of that with imbalanced

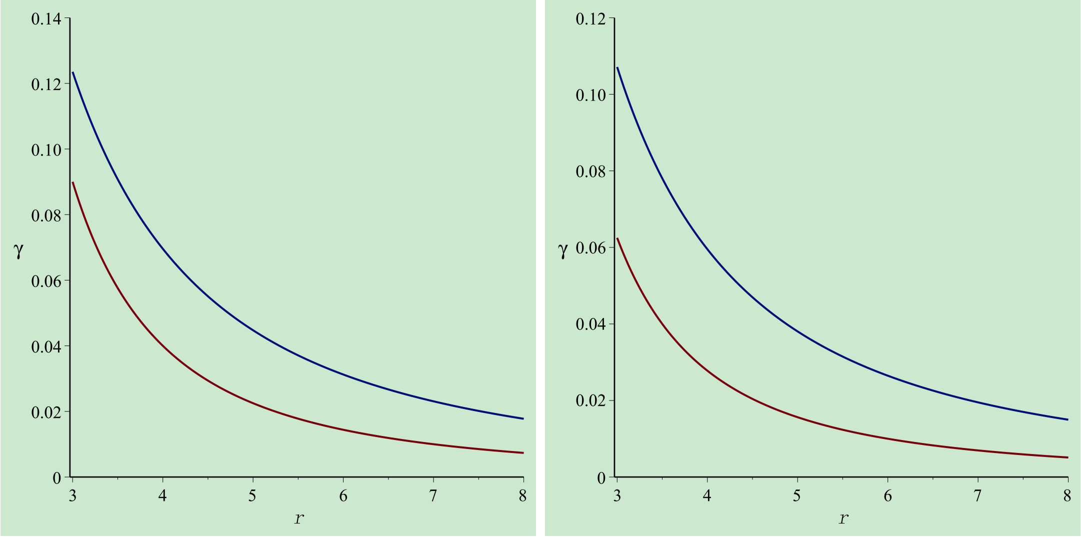

Figure 2: Upper and lower curves represent $\gamma = \frac{2(1-\beta)(r-2+\beta)}{(r-1)(r^2+2\beta-3)}$ and $\gamma = \left(\frac{1-\beta}{r-1}\right)^2$, respectively

primes. For imbalanced MPRSA, we must control the bit-size of the smaller primes. In other words, the primes should be chosen outside the factoring bound of ECM.

5 Experimental Results

In this section, we state some experimental results to show the practical performance of our method. These experiments were carried out under Ubuntu 14.04 running on a computer with Intel(R) Core$^{\text{TM}}$ i5 CPU 2.40 GHz, 2 GB RAM. We used the LLL implementation available in the NTL library [25] to reduce a basis matrix. The numbers used in each experiment were chosen uniformly at random.

During the experiments, we collected three polynomials satisfying our requirements for $s = 1$. In other words, after running the LLL algorithm, we obtained enough sufficiently short vectors. Hence, we could extract the common root by the Gröbner basis computation and finally factored the modulus N.

We give experimental results on two distinct attacks according to Sect. 3 and Sect. 4, respectively. More details are stated below. The γ_t-column provides the asymptotic bound on γ for given β and r (the imbalanced case indicates that $\beta = \frac{1}{r}$) when lattice dimension goes to infinity. The γ_e-column provides the observed experimental bound on γ for chosen β, r, s and $\log_2 N$ in our lattice setting. The γ_p-column provides the previous experimental bound on γ. We denote the bit-size of N, the LLL algorithm running time and our lattice dimension by $\log_2 N$, T and D, respectively.

Attack 1

For our factoring attack on balanced MPRSA with $r = 3$, we choose $s = 1$ and $s = 2$. It implies that we should reduce 8-dimensional and 27-dimensional lattices by the LLL algorithm. The results about the comparison of previous bound [27] and our asymptotic and experimental bounds are showed in Table 2.

Table 2: The comparison of previous bound and our asymptotic, experimental bounds on γ for our factoring attack on balanced MPRSA with $r = 3, s = 1, 2$

$\log_2 N$	$r = 3$	$s = 1$	$D = 8$	$\gamma_p \log_2 N$	$s = 2$	$D = 27$
	$\gamma_t \log_2 N$	$\gamma_e \log_2 N$	T (second)		$\gamma_e \log_2 N$	T (second)
900	120	99	1.056	97	99	478.436
1200	160	132	2.524	129	132	973.197
1500	200	165	4.016	162	165	2052.38
1800	240	199	7.196	194	199	3316.72
2100	280	232	10.544	227	232	4725.13
2400	320	265	15.248	259	265	6737.62
2700	360	298	21.464	291	299	9697.17
3000	400	332	28.638	324	332	14804.44

We first comment on the results for $D = 8$. For N of all bit-sizes, we collect three polynomials sharing a common root over the integers in the experiments. Then we take them into the Gröbner basis computation and finally obtain the right values of $p_1 - p, p_2 - p$, and $p_3 - p$, which lead to the factorization of $N = p_1 p_2 p_3$. As the modulus increases in this case, we observe that the upper bound γ_e has a tiny rise from 0.11 to 0.1106. When we turn to the Gröbner basis computation, the common roots can be successfully extracted in less than one second.

For $D = 27$, we collect more polynomials satisfying our condition in each experiment. Unfortunately, although there are more than three polynomials (actually eight polynomials), we cannot find the common root for $\gamma_e > 0.111$. From Table 2, we observe that the upper bound γ_e almost remains unchanging for the same settings except a higher lattice dimension. Thus, our attack is confirmed under Assumption 1 for $\gamma < 0.111$, which almost reaches previous theoretical bound $\frac{1}{9}$.

For $r > 3$, we do not carry out experiments since the corresponding lattice dimension is huge ($D = (s + 1)^r$). There always exist three algebraically independent polynomials in our experiment (including $r = 4$) for $s = 1$. These polynomials are sufficient to factor N for $N = p_1 p_2 p_3$ while we need more polynomials for $r > 3$. In addition, above experimental results already show our improvement.

Attack 2

We choose $s = 1$ for our factoring attack on imbalanced MPRSA with $r = 3$. Then we should reduce an 8-dimensional lattice by the LLL algorithm. The results about our asymptotic and experimental bounds are showed in Table 3.

Table 3: Our asymptotic and experimental bounds on γ for factoring attack on imbalanced MPRSA with $r = 3, s = 1$

$\log_2 N$	$\beta \log_2 N$	$r = 3$	$s = 1$	$D = 8$
		$\gamma_t \log_2 N$	$\gamma_e \log_2 N$	$\gamma_p \log_2 N$
1200	420	157	129	122
1200	480	148	119	104
1200	540	138	93	88
2400	840	314	259	245
2400	960	296	239	209
2400	1080	277	187	176
4800	1680	628	519	491
4800	1920	592	479	419
4800	2160	554	375	352

The experimental results demonstrate the efficiency of our factoring attack on imbalanced MPRSA. As showed in Table 3, our experimental bounds cover previous theoretical bounds. But this factoring attack becomes less effective as β increases. The reason may be that the small difference between p_3 and $\bar{p}$ is no longer "small" for larger β (i.e., $\beta > 0.48$ for $r = 3$). As the modulus increases, the upper bound γ_e has a tiny rise as well. Moreover, the Gröbner basis computation quickly yields the common roots in less than one second.

To summarize, our factoring attack makes a theoretical improvement by taking full advantage of the small prime difference. We then verify it by later experiments. The conclusion is that small prime difference is also a vulnerable feature for MPRSA. Especially for much smaller prime difference (i.e., $\frac{1}{r^2} < \gamma < \frac{2}{r(r+2)}$), our factoring attack is better than small private exponent attack and previous factoring attack.

Acknowledgements

This work is partly supported by the National Natural Science Foundation of China (Grant Nos. 61271271, 61522210), 100 Talents Program of Chinese Academy of Sciences, and the Fundamental Research Funds for the Central Universities in China (Grant No. WK2101020005).

References

[1] H. M. Bahig, A. Bhery, and D. I. Nassr. Cryptanalysis of multi-prime RSA with small prime difference. In *Information and Communications Security*, pages 33–44. Springer, 2012.

[2] J. Blömer and A. May. Low secret exponent RSA revisited. In *Cryptography and Lattices*, pages 4–19. Springer, 2001.

[3] D. Boneh and G. Durfee. Cryptanalysis of RSA with private key d less than $N^{0.292}$. *Information Theory, IEEE Transactions on*, 46(4):1339–1349, 2000.

[4] D. Boneh and H. Shacham. Fast variants of RSA. *CryptoBytes*, 5(1):1–9, 2002.

[5] M. Ciet, F. Koeune, F. Laguillaumie, and J.-J. Quisquater. Short private exponent attacks on fast variants of RSA. Technical report, UCL Crypto Group Technical Report Series CG-2002/4, Université Catholique de Louvain, 2002.

[6] T. Collins, D. Hopkins, S. Langford, and M. Sabin. Public key cryptographic apparatus and method, 1997. US Patent#5,848,159.

[7] D. Coppersmith. Finding a small root of a bivariate integer equation; factoring with high bits known. In *Advances in cryptology–EUROCRYPT'96*, pages 178–189. Springer, 1996.

[8] D. Coppersmith. Finding a small root of a univariate modular equation. In *Advances in cryptology–EUROCRYPT'96*, pages 155–165. Springer, 1996.

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

[10] J.-S. Coron. Finding small roots of bivariate integer polynomial equations revisited. In *Advances in Cryptology–EUROCRYPT 2004*, pages 492–505. Springer, 2004.

[11] J.-S. Coron. Finding small roots of bivariate integer polynomial equations: A direct approach. In *Advances in Cryptology–CRYPTO 2007*, pages 379–394. Springer, 2007.

[12] B. De Weger. Cryptanalysis of RSA with small prime difference. *Applicable Algebra in Engineering, Communication and Computing*, 13(1):17–28, 2002.

[13] M. J. Hinek. Low public exponent partial key and low private exponent attacks on multi-prime RSA. Master's thesis, University of Waterloo, 2002.

[14] M. J. Hinek. Small private exponent partial key-exposure attacks on multi-prime RSA. Technical report, Citeseer, 2005.

[15] M. J. Hinek. On the security of multi-prime RSA. *Journal of Mathematical Cryptology*, 2(2):117–147, 2008.

[16] M. J. Hinek, M. K. Low, and E. Teske. On some attacks on multi-prime RSA. In *Selected areas in Cryptography*, pages 385–404. Springer, 2003.

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

[18] E. Jochemsz. *Cryptanalysis of RSA variants using small roots of polynomials*. PhD thesis, Technische Universiteit Eindhoven, 2007.

[19] E. Jochemsz and A. May. A strategy for finding roots of multivariate polynomials with new applications in attacking RSA variants. In *Advances in Cryptology–ASIACRYPT 2006*, pages 267–282. Springer, 2006.

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

[21] H. W. Lenstra Jr. Factoring integers with elliptic curves. *Annals of mathematics*, pages 649–673, 1987.

[22] A. May. *New RSA vulnerabilities using lattice reduction methods*. PhD thesis, University of Paderborn, 2003.

[23] A. May. Using LLL-reduction for solving RSA and factorization problems. In *The LLL algorithm*, pages 315–348. Springer, 2010.

[24] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[25] V. Shoup. http://shoup.net/ntl/.

[26] A. Takayasu and N. Kunihiro. General bounds for small inverse problems and its applications to multi-prime RSA. In *Information Security and Cryptology–ICISC 2014*, pages 3–17. Springer, 2014.

[27] H. Zhang and T. Takagi. Attacks on multi-prime RSA with small prime difference. In *Information Security and Privacy*, pages 41–56. Springer, 2013.

[28] H. Zhang and T. Takagi. Improved attacks on multi-prime RSA with small prime difference. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 97(7):1533–1541, 2014.