RSA暗号解読の実験

2次ふるい法(QS)による因数分解

東京工芸大学 工学部 コンピュータ応用学科
数値シミュレーション研究室
後 保範 (Ushiro Yasunori)

目　次

1. 2次ふるい法(QS)

QSの基本的な考え方(合成数nの因数分解)

$$X_1^2 - n = P_1^2 \cdot P_2^0 \cdot P_3^0 \cdot P_4^1$$
$$X_2^2 - n = P_1^0 \cdot P_2^1 \cdot P_3^3 \cdot P_4^1$$
$$X_3^2 - n = P_1^1 \cdot P_2^1 \cdot P_3^1 \cdot P_4^2$$
$$X_4^2 - n = P_1^1 \cdot P_2^0 \cdot P_3^2 \cdot P_4^0$$
$$\Longrightarrow (X_1X_2X_3X_4)^2 = (P_1^2P_2^1P_3^3P_4^2)^2 \mod n$$
$$\Longrightarrow X^2 = P^2 \mod n \Longrightarrow n = (X+P)\cdot(X-P)$$

1.1 2次ふるい法の例(1/4)

n=1042387を因数分解する

(1) Pはnの平方剰余より8個の素数を選ぶ(基底)

{P} = {2, 3, 11, 17, 19, 23, 43, 47}

平方剰余: 素数Pに$s^2 \equiv 0 \pmod P$の整数sが存在

(2) $t^2 > n$ となる整数tより順に下記を行う

$1021^2 - 1042387 = 54 = 2 \cdot 3^3$

$1022^2 - 1042387 = 2097 = 3^2 \cdot 233$

$1023^2 - 1042387 = 4142 = 2 \cdot 19 \cdot 109$

1.1 2次ふるい法の例(2/4)

{P}の積に分解されたもの(QS法によるふるい)

$1021^2 - n = 54 = 2^1 \cdot 3^3$

$1027^2 - n = 12342 = 2^1 \cdot 3^1 \cdot 11^2 \cdot 17^1$

$1030^2 - n = 18513 = 3^2 \cdot 11^2 \cdot 17^1$

$1061^2 - n = 83334 = 2^1 \cdot 3^1 \cdot 17^1 \cdot 19^1 \cdot 43^1$

$1112^2 - n = 194157 = 3^5 \cdot 17^1 \cdot 47^1$

$1129^2 - n = 232254 = 2^1 \cdot 3^3 \cdot 11^1 \cdot 17^1 \cdot 23^1$

$1148^2 - n = 275517 = 3^2 \cdot 11^3 \cdot 23^1$

$1175^2 - n = 338238 = 2^1 \cdot 3^2 \cdot 19^1 \cdot 23^1 \cdot 43^1$

$1217^2 - n = 438702 = 2^1 \cdot 3^1 \cdot 11^1 \cdot 17^2 \cdot 23^1$

$1390^2 - n = 889713 = 3^2 \cdot 11^2 \cdot 19^1 \cdot 43^1$

$1520^2 - n = 1268013 = 3^1 \cdot 17^1 \cdot 23^2 \cdot 47^1$

1.1 2次ふるい法の例(3/4)

ベキが2を法として従属な行は1,2,3で

$(1021 \cdot 1027 \cdot 1030)^2 \equiv (2 \cdot 3^3 \cdot 11^2 \cdot 17^2)^2 \pmod{n}$

n = 1042387から

$1021 \cdot 1027 \cdot 1030 \equiv 1080024010 \equiv 111078 \pmod{n}$

$2 \cdot 3^3 \cdot 11^2 \cdot 17^2 \equiv 111078 \pmod{n}$

従って

$111078^2 \equiv 111078^2 \pmod{n}$

これは自明のため解ではない

1.1 2次ふるい法の例(4/4)

ベキが2を法として従属な行は5,11で

$(1112 \cdot 1520)^2 \equiv (3^3 \cdot 17 \cdot 23 \cdot 47)^2 \pmod{n}$

n=1042387から

左辺 = $(647853)^2 \equiv (496179)^2$ = 右辺 (mod n)

従って

GCD(647853-496179,n)

=GCD(15674,1042387)=1487

よって

$1042387 = 701 \cdot 1487$

2. ふるいの実施方法(基底内)

(1) 基底の選定

(2) 除算によるふるい(遅い)

(3) 高速なふるい

  (a) ふるいテーブルの作成

  (b) ふるいテーブルによるふるい

  (c) ふるい結果による行列の作成

2.1 基底の選定

(1) L以下の素数を選定

(2) 下記が成立する素数pを基底に選定

$x^2 - n \equiv 0 \pmod{p}$

となる整数xが存在する。

(3) (2)の条件は平方剰余で下記と同等

$n^{(p-1)/2} \pmod{p} \equiv 1$ ： 平方剰余

$\equiv -1$ ：平方非剰余

→ 平方剰余ならpは基底となる

2.1 基底の選定の例

(1) n=930091 (分解対象数)

(2) 43以下の素数を対象とする

素数pに対しs=$n^{(p-1)/2}$ (mod p)は下記

(a) s=1となる素数

{ 2, 3, 5, 7,17, 43 }

(b) s=-1となる素数

{ 11, 13, 19, 23, 29, 31, 37, 43}

→ 基底は-1を加え { -1, 2, 3, 5, 7, 17, 43 }

2.2 除算によるふるい(遅い)

(1) n=930091、sqrt(n)=965

(2) 基底= { -1, 2, 3, 5, 7, 17, 43 }

(3) x=sqrt(n)とし、下記の様に分解

$x^2 - n$ = 1134 = $2 \cdot 3^4 \cdot 7$ → 採用

$(x+1)^2 - n$ = 3065 = $5 \cdot 613$ → 不採用

$(x+2)^2 - n$ = 4998 = $2 \cdot 3 \cdot 7^2 \cdot 17$ → 採用

….

$(x-1)^2 - n$ = -795 = $-1 \cdot 3 \cdot 5 \cdot 53$ → 不採用

$(x-2)^2 - n$ = -2722 = $-1 \cdot 2 \cdot 1361$ → 不採用

2.3 高速なふるい

(1) 原理

$f(x) = x^2 - n \equiv 0 \pmod p$なら

整数kに対し$f(x+k \cdot p) \equiv 0 \pmod p$を利用

(2) 適用方法

$f(s) = s^2 - n \equiv 0 \pmod p$を求める

区間内のs+kpに対して、f(s+kp)がpで割れる評価値(最初1として、pを除算)を計算

f(x)とf(x)の評価値が同じデータを採用

2.4 ふるいテーブルの作成

(1) $f(s) = (M+s)^2 - n \equiv 0 \pmod{p^k}$を求める

(2) 求めたテーブル, M=sqrt(n)=965

p	p^k	s
2	2	0
3	3	0, 2
3	9	0, 5
3	27	0, 14
5	5	1, 4
5	25	6, 14
7	7	0, 2
17	17	2, 6
43	43	23, 25

2.5 ふるいの具体例(1/2)

x	x^2-n	評価値	2	3	3^2	3^3	5	5^2…	
865	1134	1134	2	3	3	3	1	1	→ 採用
866	3065	5	1	1	1	1	5	1	
867	4998	4998	2	3	1	1	1	1	→ 採用
868	6933	3	1	3	1	1	1	1	
869	8870	10	2	1	1	1	5	1	
870	10809	9	1	3	3	1	1	1	
871	12750	12750	2	3	1	1	5	5	→ 採用
872	14693	7	1	1	1	1	1	1	
873	16638	6	2	3	1	1	1	1	
874	18585	315	1	3	3	1	5	1	

2.5 ふるいの具体例(2/2)

x	x^2-n	評価値	2	3	3^2	3^3	5	5^2…	
875	20534	2	2	1	1	1	1	1	
876	22485	15	1	3	3	1	5	1	
877	24438	6	2	3	1	1	1	1	
878	26393	1	1	1	1	1	1	1	
879	28350	28350	2	3	3	3	5	5	→ 採用
880	30309	3	1	3	1	1	1	1	
881	32270	70	2	1	1	1	5	1	
882	34233	3	1	3	1	1	1	1	
883	36198	18	2	3	3	1	1	1	
884	38165	85	1	1	1	1	5	1	

2.6 ふるい結果による行列の作成

(1) 採用したデータを集めて作成(+,-方向に70)
(2) ふるい結果の行列

x	x^2-n	-1	2	3	5	7	17	43
965	1134	0	1	4	0	1	0	0
967	4998	0	1	1	0	2	1	0
971	12750	0	1	1	3	0	1	0
979	28350	0	1	4	2	1	0	0
988	46053	0	0	2	0	1	1	1
902	-116487	1	0	2	0	1	0	2
904	-112875	1	0	1	3	1	0	1
916	-91035	1	0	2	1	1	2	0
947	-33282	1	1	2	0	0	0	2

3. ふるいの実施方法(基底外)

(1) 考え方

v=(x^2 – n)/評価値が1にならなくても、一定の値より小さいと、それを2次ふるいとして仮採用

仮採用したふるいデータに同じvがでれば採用

(2) 実施方法

(a) 一定値=最大の基底×h(入力)とする

(b) v=1となるデータは無条件採用(1次)

(c) vが一定値以下なら仮採用

(d) 仮採用で同じvが2件以上ならvを基底に、当該仮採用データを本採用(2次)

3.1 ふるいの実施例(基底外)

x	a/b	$a=x^2-n$	評価値(b)	
965	1	1134	1134	→ 1次採用
966	613	3065	5	→ 2次候補
967	1	4998	4998	→ 1次採用
968	2311	6933	3	
969	887	8870	10	(43は14番目の素数)
970	1201	10809	9	
971	1	12750	12750	→ 1次採用
972	2099	14693	7	(14×10番の素数)
973	2773	16638	6	a/b < 787
974	59	18585	315	→ 2次候補

4. 0-1行列から従属行の計算

(1) ふるい結果の行列から0-1行列へ変換

基底のベキの部分の(mod 2)をとり、0-1行列作成 → 行列Aとする(データ数 × 基底数)

(2) 行列A(0-1行列)の消去

軸交換付きガウス消去法で、消去する

軸交換及び対角行列番号をベクトルに残す

→ 高速化にはビット表示や疎行列処理する

(3) 消去した行列から従属行の取り出し

消去後の行列は複数の従属行を示す

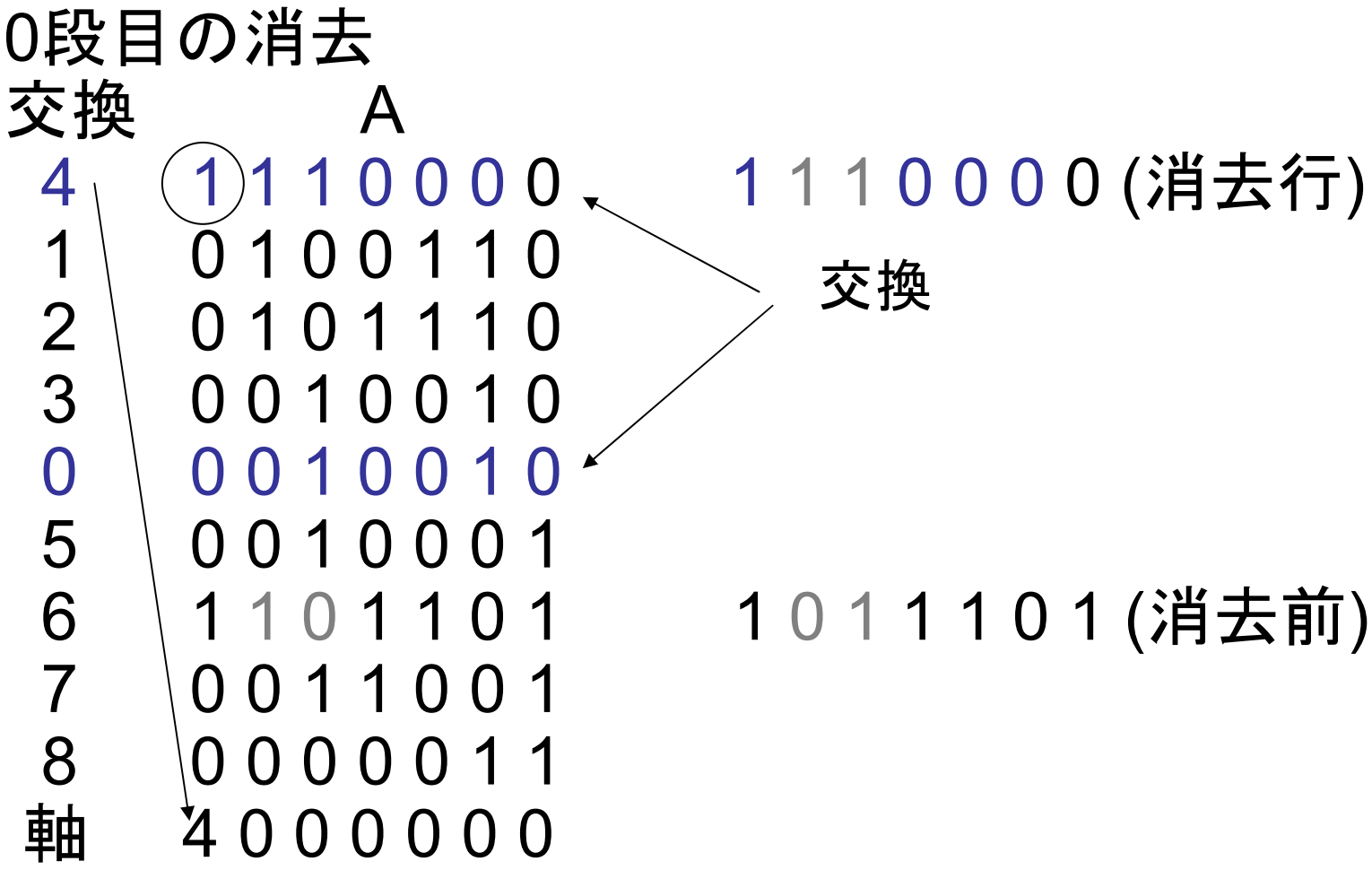
4.1 0-1行列への変換(1/6)

0-1行列への変換結果

$$
A = \begin{matrix}
0&0&1&0&0&1&0\\
0&1&0&0&1&1&0\\
0&1&0&1&1&1&0\\
0&0&1&0&0&1&0\\
1&1&1&0&0&0&0\\
0&0&1&0&0&0&1\\
1&0&1&1&1&0&1\\
0&0&1&1&0&0&1\\
0&0&0&0&0&1&1
\end{matrix}
$$

7×6の行列
→ 従属行は1個以上ある

4.2 行列の消去(1/4)


0段目の消去
交換
A
4 1 1 1 0 0 0 0
1 0 1 0 0 1 1 0
2 0 1 0 1 1 1 0
3 0 0 1 0 0 1 0
0 0 0 1 0 0 1 0
5 0 0 1 0 0 0 1
6 1 1 0 1 1 0 1
7 0 0 1 1 0 0 1
8 0 0 0 0 0 1 1
軸 4 0 0 0 0 0 0 0
1 1 1 0 0 0 0 0 (消去行)
交換
1 0 1 1 1 0 1 (消去前)

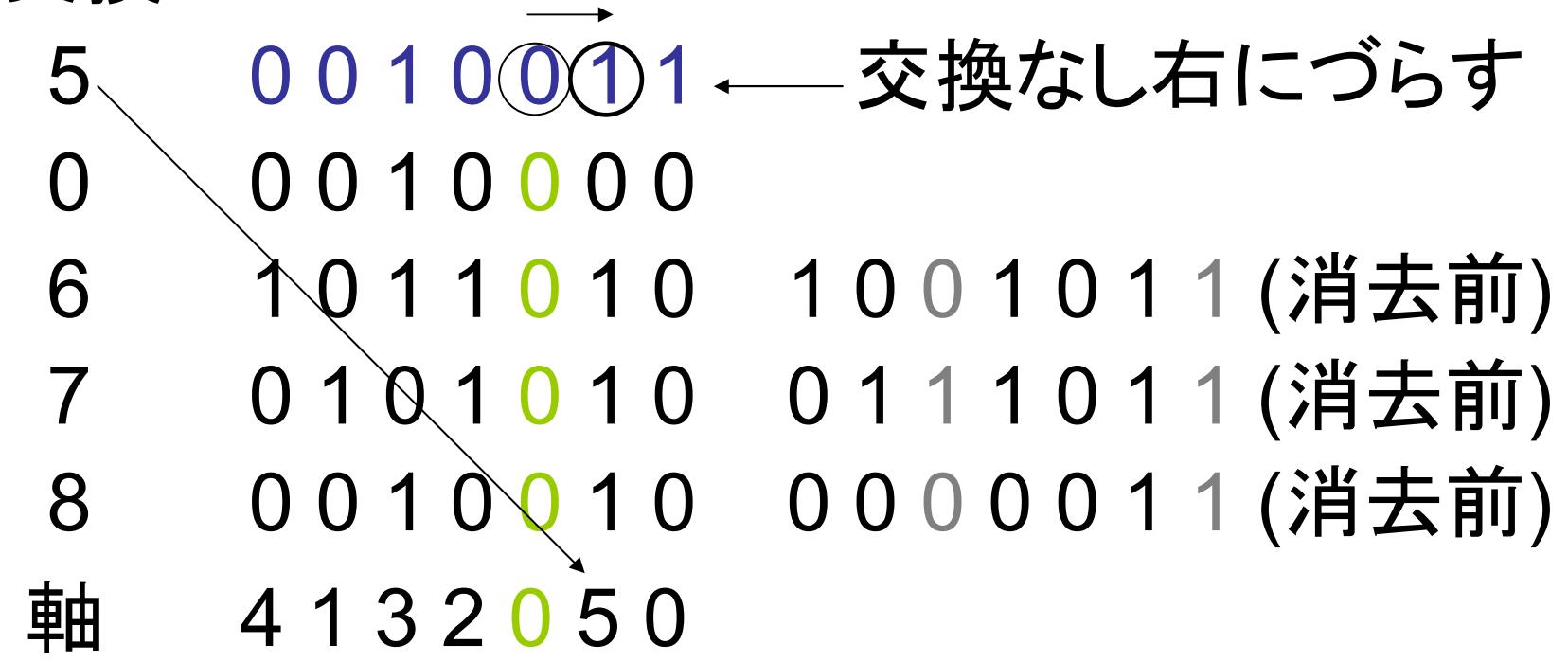
4.2 行列の消去(2/4)

1段目の消去(0行は対象外なので除く)

交換	A	
1	0 1 0 0 1 1 0	← 交換なし
2	0 1 0 1 0 0 0	0 1 0 1 1 1 0 (消去前)
3	0 0 1 0 0 1 0	
0	0 0 1 0 0 1 0	
5	0 0 1 0 0 0 1	
6	1 1 0 1 0 1 1	1 1 0 1 1 0 1 (消去前)
7	0 0 1 1 0 0 1	
8	0 0 0 0 0 1 1	
軸	4 1 0 0 0 0 0	

4.2 行列の消去(3/4)

4段の消去(0～3行は対象外なので除く)



4.2 行列の消去(4/4)

最終段の消去(0～4行は対象外なので除く)

交換　　A

0　0 0 1 0 0 0 0

6　1 0 1 1 0 1 0

7　0 1 0 1 0 1 0

8　0 0 1 0 0 1 0

軸　4 1 3 2 0 5 0

→ 4行残っているので
従属行は4組

従属行1 → 0　3

4.3 従属行

最終段より

交換	A		従属行
0	0 0 1 0 0 0 0	→	0 3
6	1 0 1 1 0 1 0	→	6 4 3 2 5
7	0 1 0 1 0 1 0	→	7 1 2 5
8	0 0 1 0 0 1 0	→	8 3 5

軸　4 1 3 2 0 5 0

5. QS法による因数分解

(1) 0,3行の基底のベキの集計結果

基底	-1	2	3	5	7	17	43
ベキ数	0	2	8	2	2	0	0

(2) 0,3行の関係式

左辺(x) = $965 \times 979 = 14644 \pmod{930091}$

右辺(ベキ) = $2 \times 3^4 \times 5 \times 7 = 5670$

(3) 因数分解

b = 左辺 – 右辺 = 14644 – 5670 = 8974

GCD(8974, 930091)=641 → n=641 × 1451

6. 提出するレポートの纏め方

(1) ワードを使用し、10ページ以上(表紙、目次、図、表を含む)を作成
(2) 表、図はQS法のプログラムの実行結果からエクセルで作成し、ワードに貼り付け
(3) E-learning(Moodle) の「調査・製作ワークショップ」中の「RSA暗号解読の実験レポート」に作成したワードを提出
(4) 提出期限は講義後10日(11/28)
(5) 採点は提出したワードだけで行う

6.1 レポートの表紙

調査・製作ワークショップ
「暗号解読の数値実験結果報告」

作成日　:　2008年11月xx日
担当教員:　後 保範
提出者:　　コンピュータ応用学科3年
　　学籍番号　氏名

6.2 レポートの目次

目次

1. はじめに
2. QS法の概要
3. 因数分解対象データと使用計算機
4. QS法の数値実験結果
5. 分解桁数と基底数(素数の数)の関係
6. おわりに
7. 参考文献及び資料

6.3 因数分解対象データ

(1) 分解対象数(クラスで変えている)

対象数のビット数が(a)又は(b)の5ケースで行う

(a) 90, 100, 110, 120, 130ビット

(b) 100, 110, 120, 130, 140ビット

(2) 使用基底数(指定は素数の数)

(a) 計算時間が最も短くなる部分を中心に

(b) 各ビットに対し10～20個の異なる数で測定

(3) 注意事項

異常な計算時間の場合は再測定が必要

6.4 QS法の数値実験例(1/2)

表.1 130ビットの基底数(素数)による分解結果(QS)

素数 NP	時間 Time(s)	データ数 1次	データ数 合計	ふるい数 (M)	分解結果 OK	分解結果 NG
1500	12.6	318	1130	109.6	6	14
2000	9.4	433	1496	67.4	15	17
2500	8.7	592	1833	48.3	20	22
3000	9.3	749	2168	39.2	34	27
3500	10.4	924	2492	33.1	34	40
4000	12.1	1110	2792	29.6	55	40
合計					164	160

分解対象数:739818217219532983579711540812099863131

6.4 QS法の数値実験例(2/2)

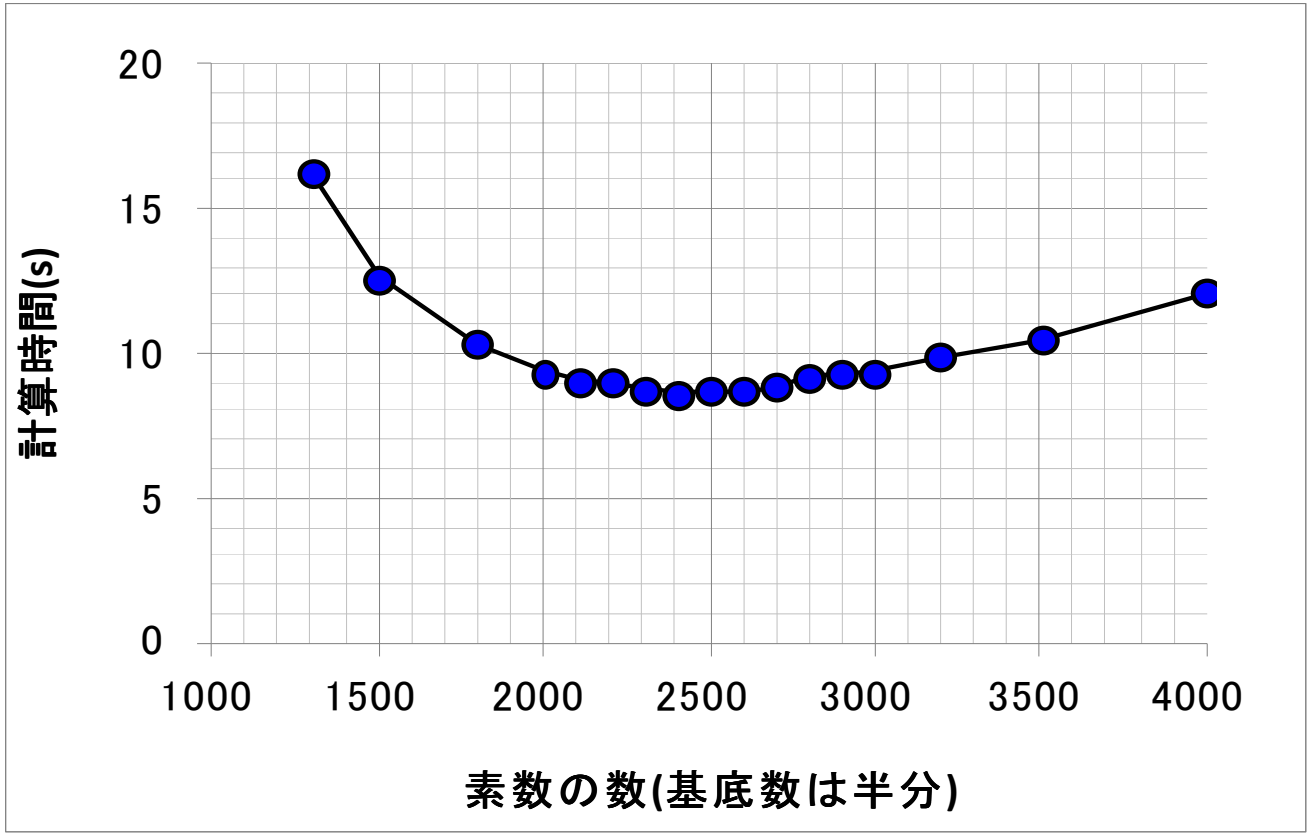


図1. QS法による因数分解時間（130ビット)