RSA暗号計算 No.4
0-1行列の従属行の計算

2007年5月

後　保範　(東京工芸大学)

目次

1. 0-1行列の従属行計算の目的

(1) 0-1行列の従属行計算の目的

多数の$a_1 \cdot a_2 \cdots a_k \equiv p_1 \cdot p_2 \cdots p_h$ (mod n)の関係を組み合わせて$\alpha^2 \equiv \beta^2$ (mod n)なる関係を選び出すため。

(2) 0-1行列を発生させる計算

(a) ふるい法関係

MPQS, GNFS, MBPS等のふるい法

楕円曲線法(ECM)ではこの計算は不要

2. ガウス消去法(原理的方法)

対象とするn×m次元の0-1行列をAとする

(1) n×n次元の単位行列をIとする。
n×(m+n)次元の行列A+Iを作成

(2) A+Iの行列に対して列交換ガウス消去法(mod 2の基で)を適用する。

(3) Aに対応する行の要素が全てゼロの行を取り出す。

(4) 上記に対応するIの部分で1のある列番号がそのまま従属行の番号となる。

2.1 ガウス消去前の行列

行番号	A	+ I
1	0 1 0 1 0 0 1 1	1 0 0 0 0 0 0 0
2	0 1 1 0 0 0 0 1	0 1 0 0 0 0 0 0
3	1 1 1 1 0 0 0 0	0 0 1 0 0 0 0 0
4	0 0 1 0 1 1 0 0	0 0 0 1 0 0 0 0
5	0 1 1 0 0 1 0 0	0 0 0 0 1 0 0 0
6	0 1 1 1 1 0 1 0	0 0 0 0 0 1 0 0
7	1 1 1 1 0 1 1 0	0 0 0 0 0 0 1 0
8	0 1 0 0 1 1 0 1	0 0 0 0 0 0 0 1

2.2 ガウス消去後の行列

行番号	A	+ I
1	1 1 1 1 0 0 0 0	0 0 1 0 0 0 0 0
2	0 1 1 0 0 0 0 1	0 1 0 0 0 0 0 0
3	0 0 1 1 0 0 1 0	1 1 0 0 0 0 0 0
4	0 0 0 1 1 1 1 0	1 1 0 1 0 0 0 0
5	0 0 0 0 0 1 0 1	0 1 0 0 1 0 0 0
6	0 0 0 0 0 0 1 1	0 1 1 0 1 0 1 0
7	0 0 0 0 0 0 0 0	1 1 0 1 1 1 0 0
8	0 0 0 0 0 0 0 0	0 1 0 1 0 0 0 1

2. 3 求める従属行

(1) 消去後の7行目から

7　0 0 0 0 0 0 0 0　1 1 0 1 1 1 0 0

→ 従属行: 1, 2, 4, 5, 6

(2) 消去後の8行目から

8　0 0 0 0 0 0 0 0　0 1 0 1 0 0 0 1

→ 従属行: 2, 4, 8

7

3. ガウス消去法の工夫

(1) 行列n×m次元行列Aで消去する
(2) 長さnとmのベクトルを用意する
(3) 軸交換と枢軸情報をベクトルに保持
(4) 枢軸をksとし消去は下記

$A_{ij} = A_{ij} - A_{is} \cdot A_{kj} \pmod 2, \quad i>k, \ j \neq s$

(5) 消去完了行以下に従属行の情報
(6) 両ベクトルで本来の従属行情報に変換

8

3.1 消去前の行列

交換	行列A							
1	0	1	0	1	0	0	1	1
2	0	1	1	0	0	0	0	1
3	1	1	1	1	0	0	0	0
4	0	0	1	0	1	1	0	0
5	0	1	1	0	0	1	0	0
6	0	1	1	1	1	0	1	0
7	1	1	1	1	0	1	1	0
8	0	1	0	0	1	1	0	1

枢軸	0	0	0	0	0	0	0	0

9

3.2 消去過程 (1/5)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	0	1	0	0	1	1
4	0	0	1	0	1	1	0	0
5	0	1	1	0	0	1	0	0
6	0	1	1	1	1	0	1	0
7	1	0	0	0	0	1	1	0
8	0	1	0	0	1	1	0	1

枢軸	3	0	0	0	0	0	0	0

10

3.3 消去過程 (2/5)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	0
4	0	0	1	0	1	1	0	0
5	0	1	0	0	0	1	0	1
6	0	1	0	1	1	0	1	1
7	1	0	0	0	0	1	1	0
8	0	1	1	0	1	1	0	0

枢軸	3	2	0	0	0	0	0	0

11

3.4 消去過程 (3/5)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	0
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
6	0	1	0	1	1	0	1	1
7	1	0	0	0	0	1	1	0
8	0	0	1	1	1	1	1	0

枢軸	3	2	1	0	0	0	0	0

12

3.5 消去過程 (4/5)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	0
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
6	0	0	1	1	0	1	0	1
7	1	0	0	0	0	1	1	0
8	0	1	0	1	0	0	0	0
枢軸	3	2	1	4	0	0	0	0

13

3.6 消去過程 (5/5)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	0
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
6	0	1	1	1	0	1	0	0
7	1	1	0	0	0	1	1	1
8	0	1	0	1	0	0	0	0
枢軸	3	2	1	4	0	5	0	0

14

3.7 消去結果

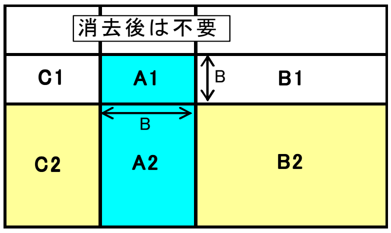
交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	0
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
7	1	1	0	0	0	1	1	1
6	0	1	1	1	0	1	0	0
8	0	1	0	1	0	0	0	0
枢軸	3	2	1	4	0	5	7	0

従属行1: 1, 2, 4, 5, 6

従属行2: 2, 4, 8

15

4. ブロック·ガウス消去法



1. A1,A2の消去
2. B2=B2−A2･B1 (mod 2)で消去
3. C2=C2−A2･B1 (mod 2)で消去

16

4.1 ブロック消去前の行列

交換	行列A							
1	0	1	0	1	0	0	1	1
2	0	1	1	0	0	0	0	1
3	1	1	1	1	0	0	0	0
4	0	0	1	0	1	1	0	0
5	0	1	1	0	0	1	0	0
6	0	1	1	1	1	0	1	0
7	1	1	1	1	0	1	1	0
8	0	1	0	0	1	1	0	1
枢軸	0	0	0	0	0	0	0	0

17

4.2 ブロック消去過程 (1/4)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	1
4	0	1	1	0	1	1	0	0
5	0	1	0	0	0	1	0	0
6	0	1	0	1	1	0	1	0
7	1	0	0	1	0	1	1	0
8	0	0	1	0	1	1	0	1
枢軸	3	2	1	0	0	0	0	0

18

4.3 ブロック消去過程 (2/4)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	1
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
6	0	1	0	1	1	0	1	1
7	1	0	0	0	0	1	1	0
8	0	0	1	1	1	1	1	0

枢軸								
枢軸	3	2	1	0	0	0	0	0

19

4.4 ブロック消去過程 (3/4)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	1
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
6	0	1	0	1	0	1	1	1
7	1	0	0	0	0	1	1	0
8	0	0	1	1	0	0	1	0

枢軸								
枢軸	3	2	1	4	0	5	0	0

20

4.5 ブロック消去過程 (4/4)

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	1
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
6	0	1	1	1	0	1	0	0
7	1	1	0	0	0	1	1	1
8	0	1	0	1	0	0	0	0

枢軸								
枢軸	3	2	1	4	0	5	0	0

21

4.6 ブロック消去結果

交換	行列A							
3	1	1	1	1	0	0	0	0
2	0	1	1	0	0	0	0	1
1	0	1	1	1	0	0	1	1
4	0	1	1	1	1	1	1	0
5	0	1	0	0	0	1	0	1
7	1	1	0	0	0	1	1	1
6	0	1	1	1	0	1	0	0
8	0	1	0	1	0	0	0	0

枢軸								
枢軸	3	2	1	4	0	5	7	0

従属行1:　1, 2, 4, 5, 6
従属行2:　2, 4, 8

22

5. 大次元行列の場合の工夫

(1) 32ビット整数又は64ビット整数を使用し、
32要素又は64要素を一括処理
→排他和演算を使用し一括処理

(2) 疎行列直接解法の使用
非ゼロ要素だけを記憶し、記憶領域及び計算量を削減する。番号付けが重要

(3) ブロックランチョス法を使用した反復解法
記憶領域は最も少なくできる。

23