ブレイド群を利用した公開鍵暗号系の実装と性能評価実験

日本大学　文理学部
情報システム解析学科
谷研究室　田中裕子

目次

目次

研究の目的

- 2000年に K.H.Ko , S.J.Lee らがブレイド群を利用した公開鍵暗号系を提案
- 現在広く使われているRSAにかわる公開鍵暗号系となるのか？処理速度や安全性は？
- 実装して性能評価実験を行う

目次

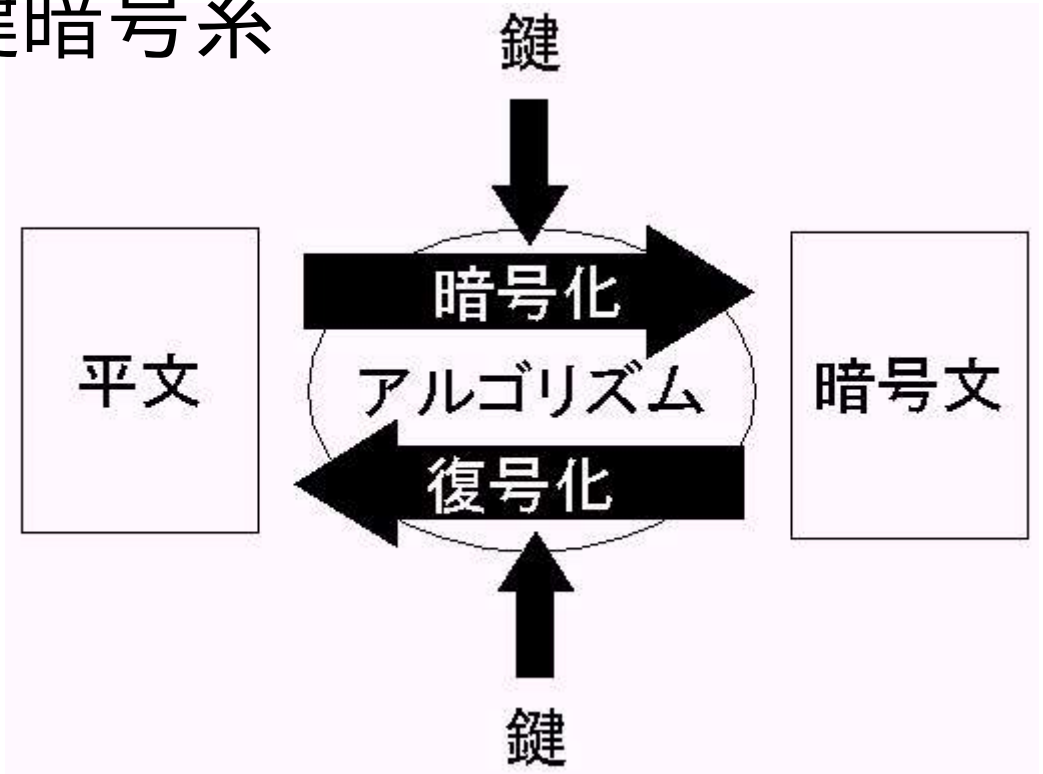
暗号系

当事者以外にはわからないように
鍵を使い決められた手順に従って文を変換する

暗号系

当事者以外にはわからないように
鍵を使い決められた手順に従って文を変換する

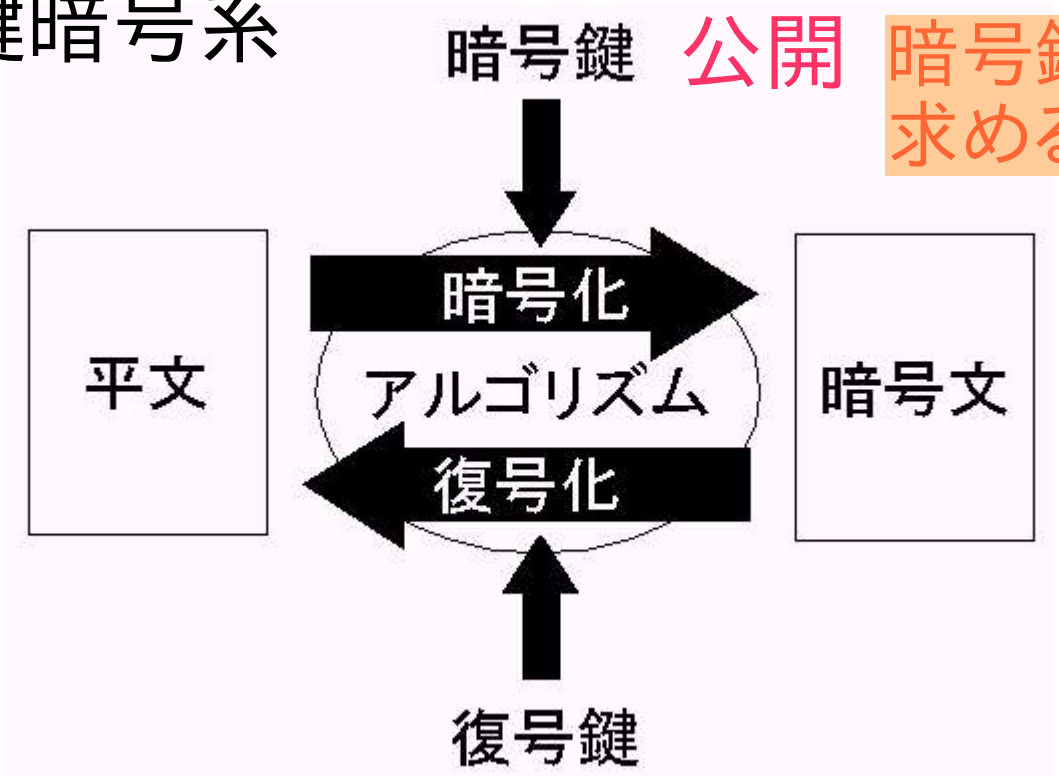
◆秘密鍵暗号系



暗号系

当事者以外にはわからないように
鍵を使い決められた手順に従って文を変換する

◆公開鍵暗号系

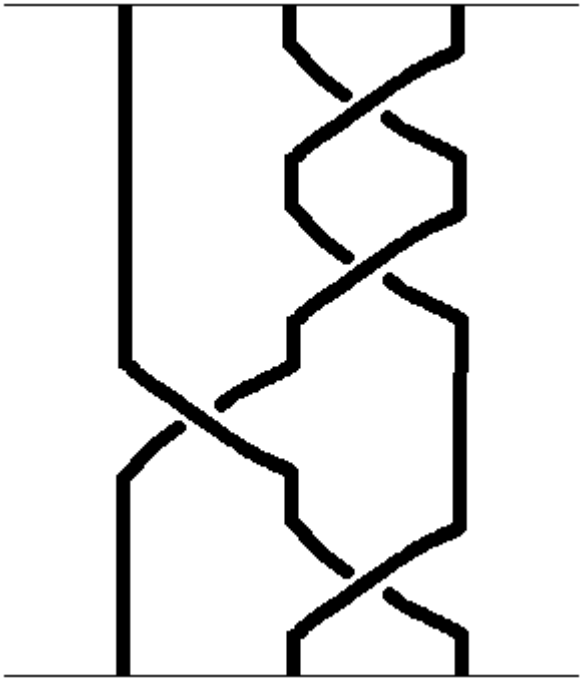


目次

ブレイド群に関する定義

◆n-ブレイド

平行な2つの面に接続しているn本の組み紐

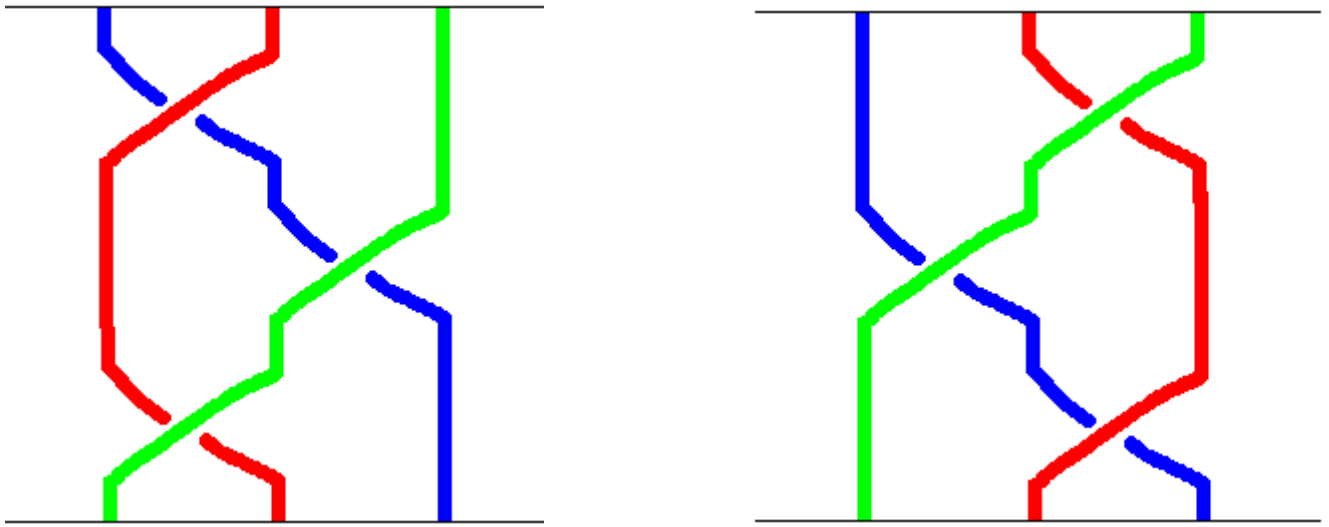


3-ブレイドの例

ブレイド群に関する定義

◆ブレイドの同値

紐が面に接続している点を固定したまま動かして同じものにできる

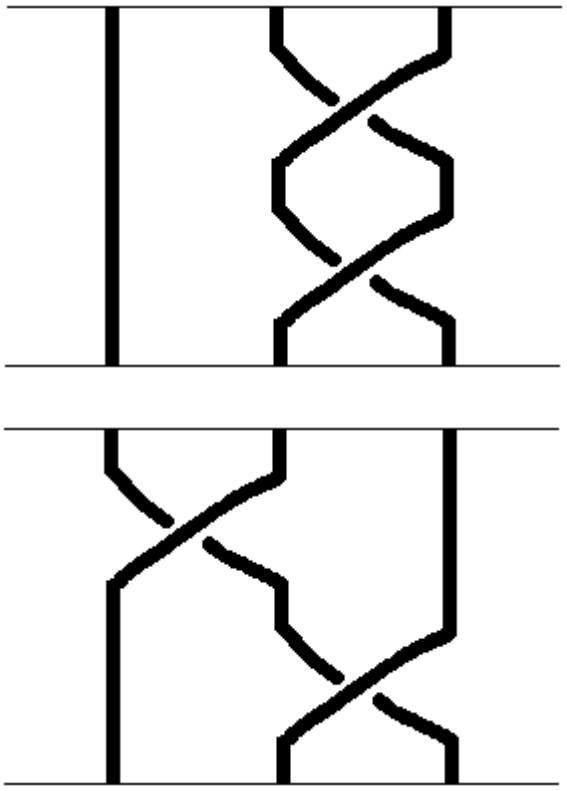


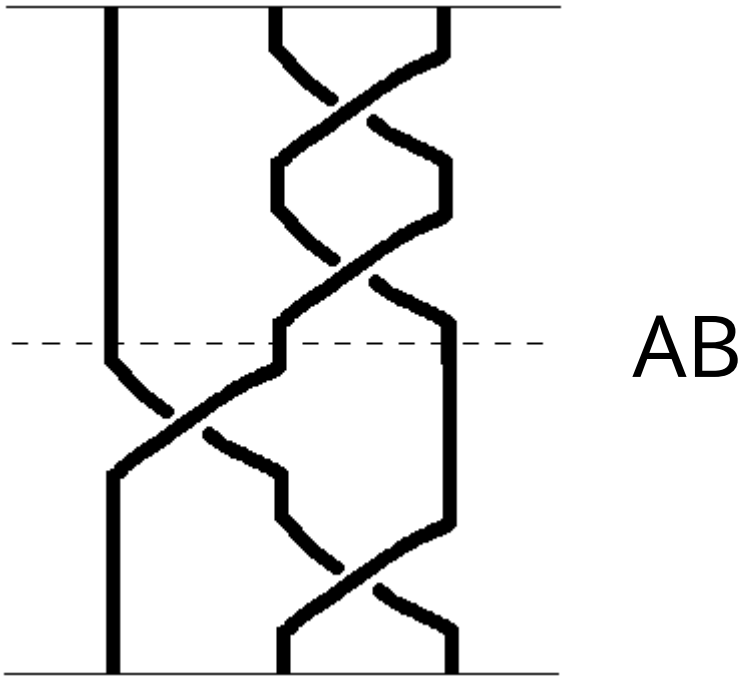
赤い紐を動かせば同じ

ブレイド群に関する定義

◆ブレイドの積

AとBの積ABはAをBの上に置いたもの



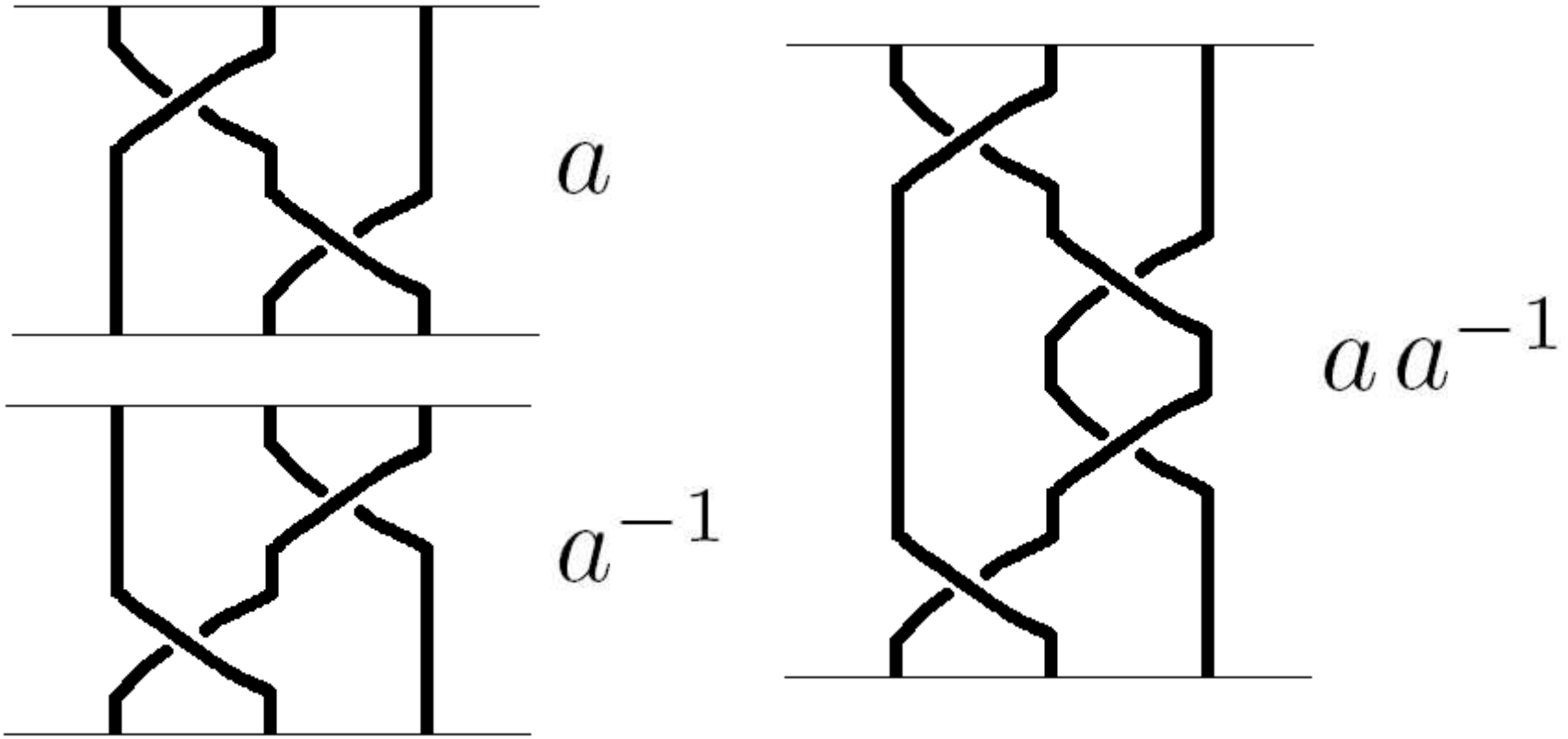


ブレイドは可換でない

ブレイド群に関する定義

◆逆元

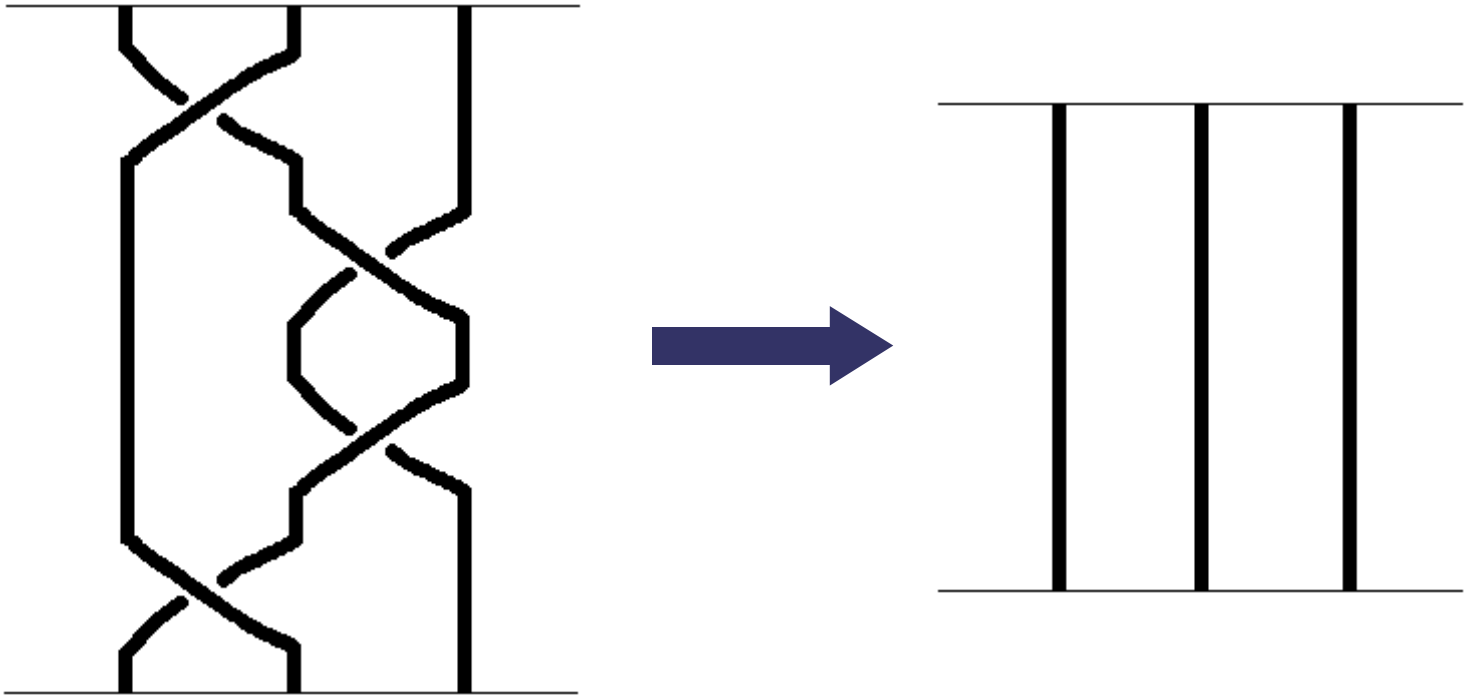
紐が接続している面で折り返したもの



ブレイド群に関する定義

◆単位元

どの紐も交差しないもの

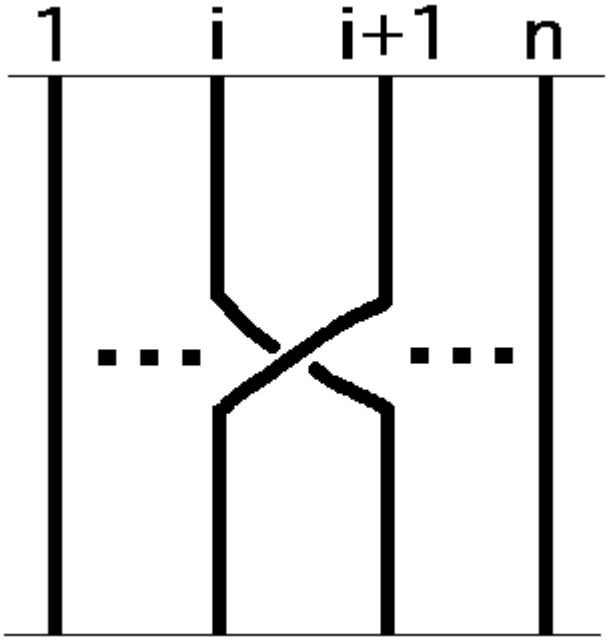


ブレイド群に関する定義

◆生成元

$\sigma_i \cdots i$ 番目と $i+1$ 番目の紐のみが交差

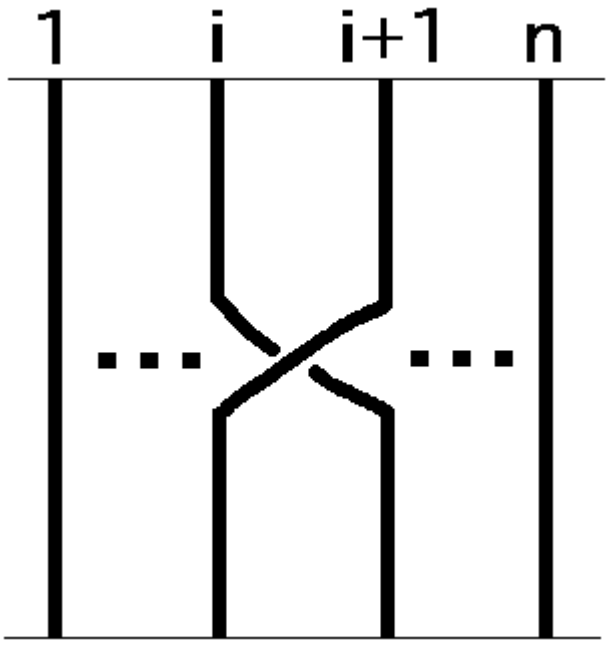
i 番目の紐が $i+1$ 番目の紐の下になる



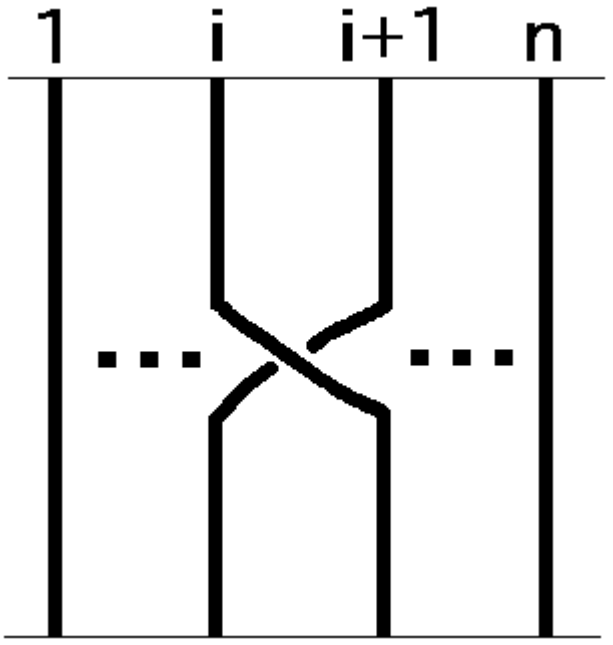
ブレイド群に関する定義

◆生成元

$\sigma_i^{-1} \cdots \sigma_i$の紐の重なり方を上下反対にしたもの



σ_i

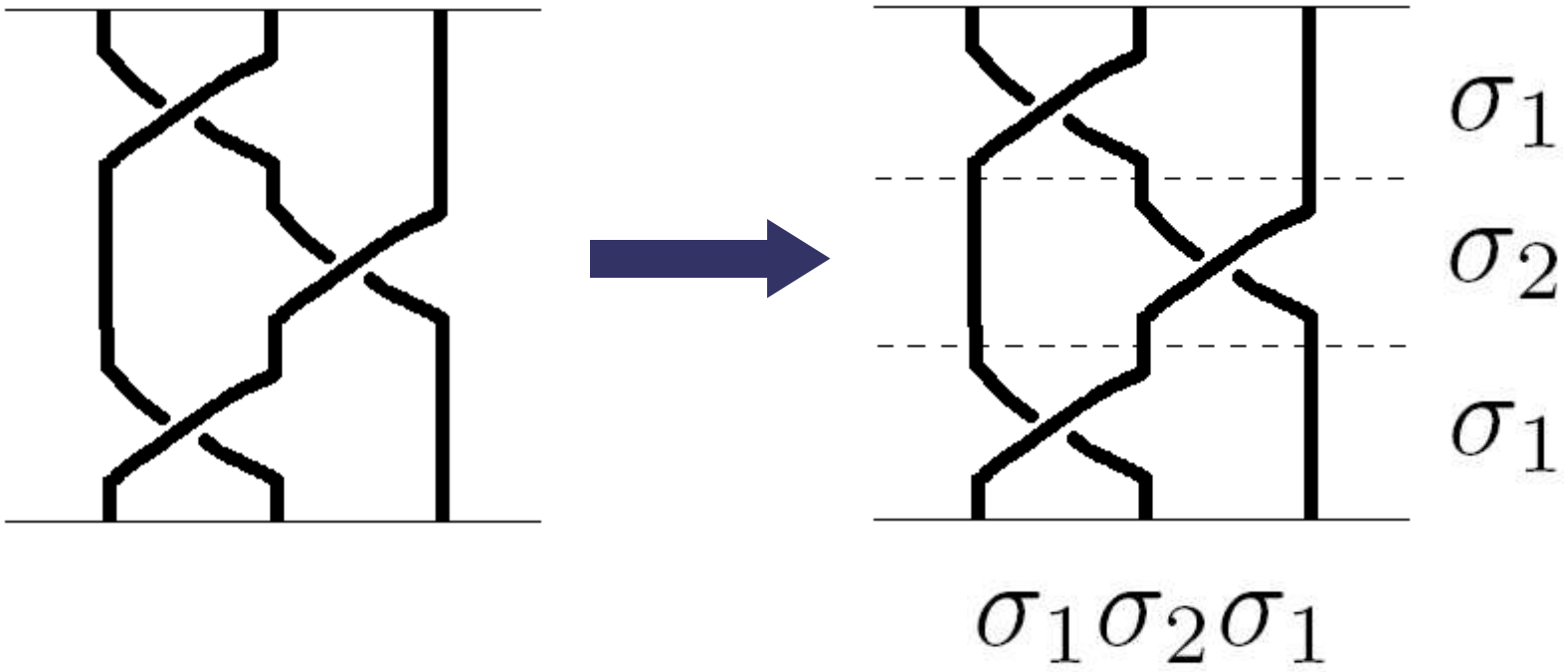


σ_i^{-1}

ブレイド群に関する定義

◆生成元

ブレイドは生成元の積で表される

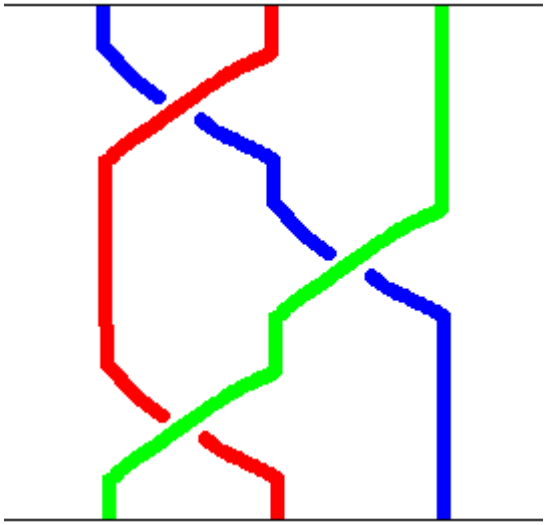


ブレイド群に関する定義

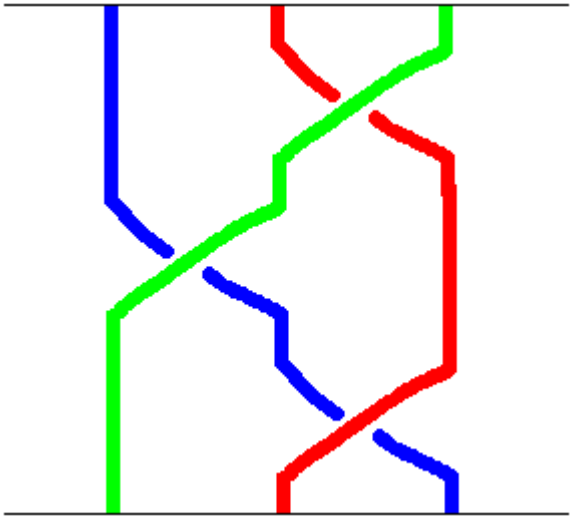
◆生成元

ブレイドは生成元の積で表される

→ 一意に決まらない → 標準形



$\sigma_1\sigma_2\sigma_1$

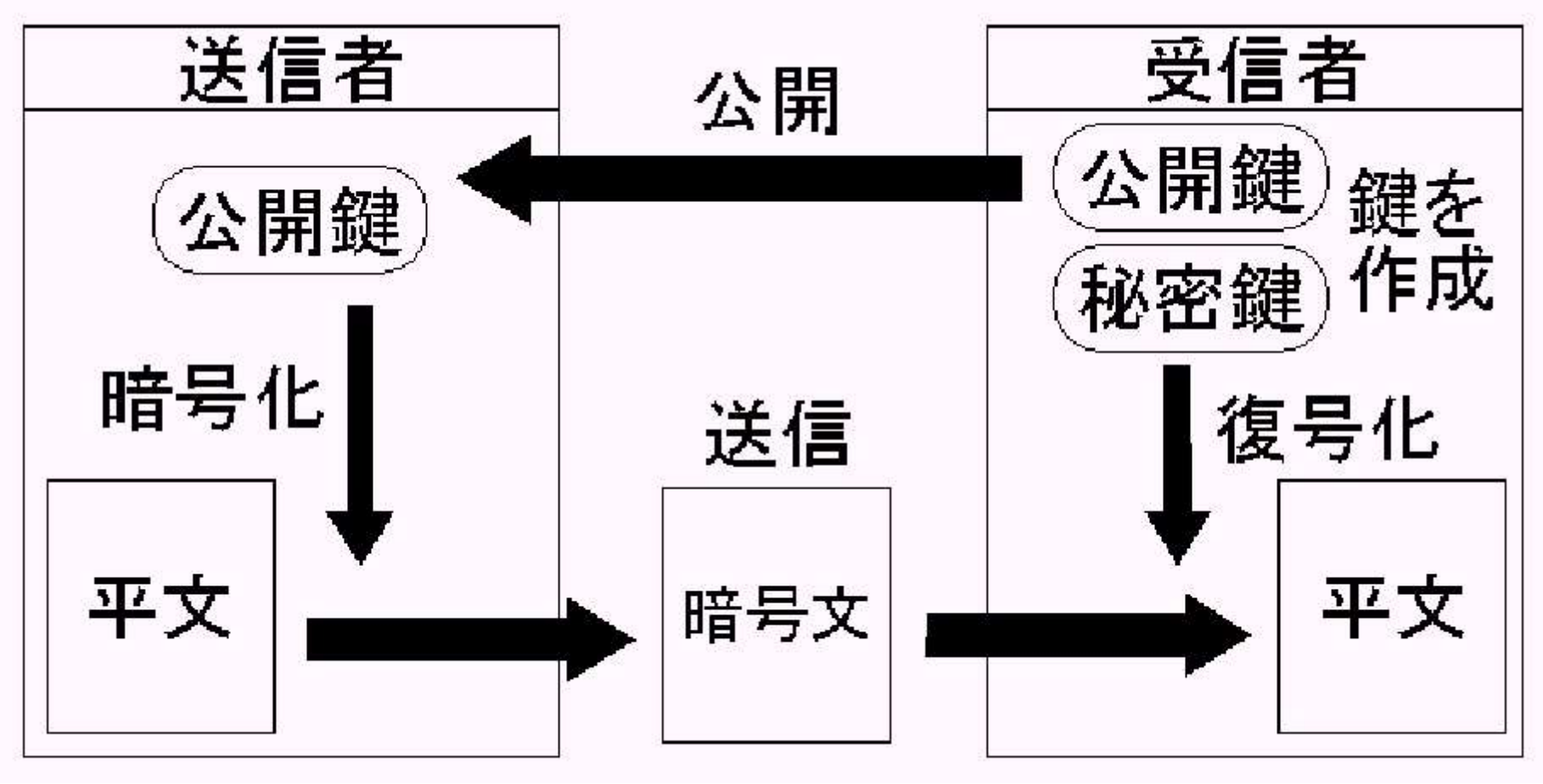


$\sigma_2\sigma_1\sigma_2$

目次

ブレイド群を利用した公開鍵暗号系

◆公開鍵暗号系のしくみ



ブレイド群を利用した公開鍵暗号系

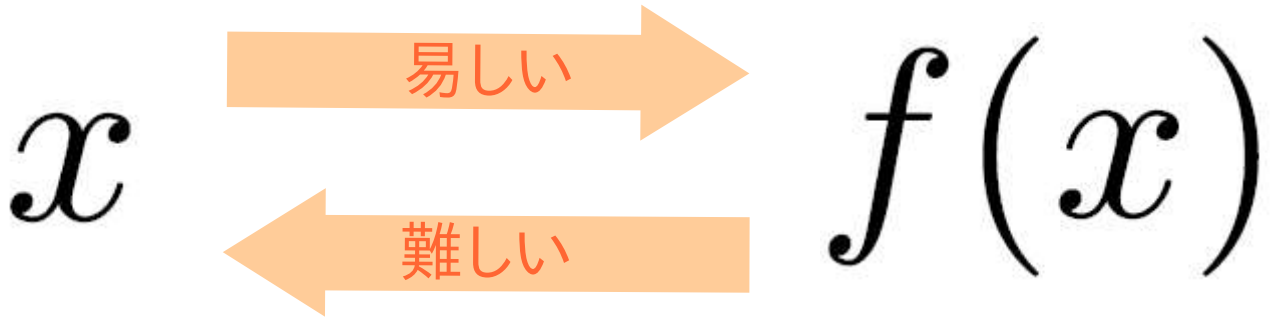
◆公開鍵暗号系が成立する条件

- 一方の鍵で暗号化したものはペアとなる
  もう一方の鍵でしか復号化できない
- 公開鍵（暗号鍵）から秘密鍵（復号鍵）を
  求めることができない → 一方向関数の利用

ブレイド群を利用した公開鍵暗号系

◆一方向関数

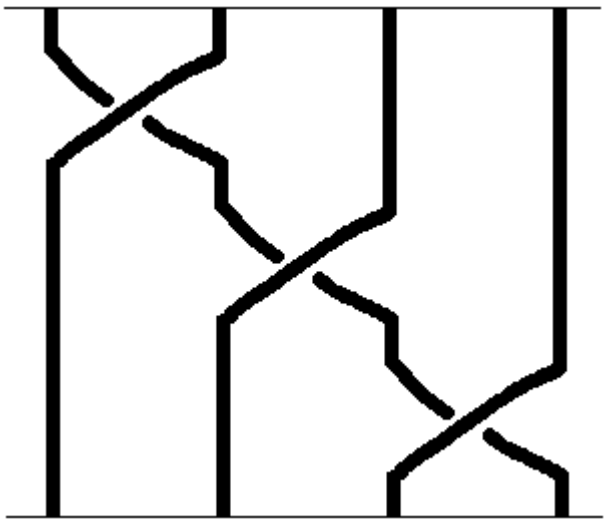
1. x を入力して $f(x)$ を出力する多項式時間アルゴリズムが存在する
2. f の逆関数を計算するすべての確率的多項式時間アルゴリズムはわずかな確率でしか成功しない



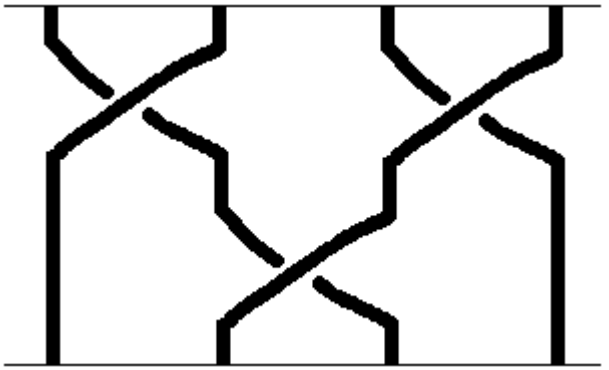
ブレイド群を利用した公開鍵暗号系

◆ブレイド群における一方向関数

$y = axa^{-1}$ となる a が存在するとき x と y は共役



x

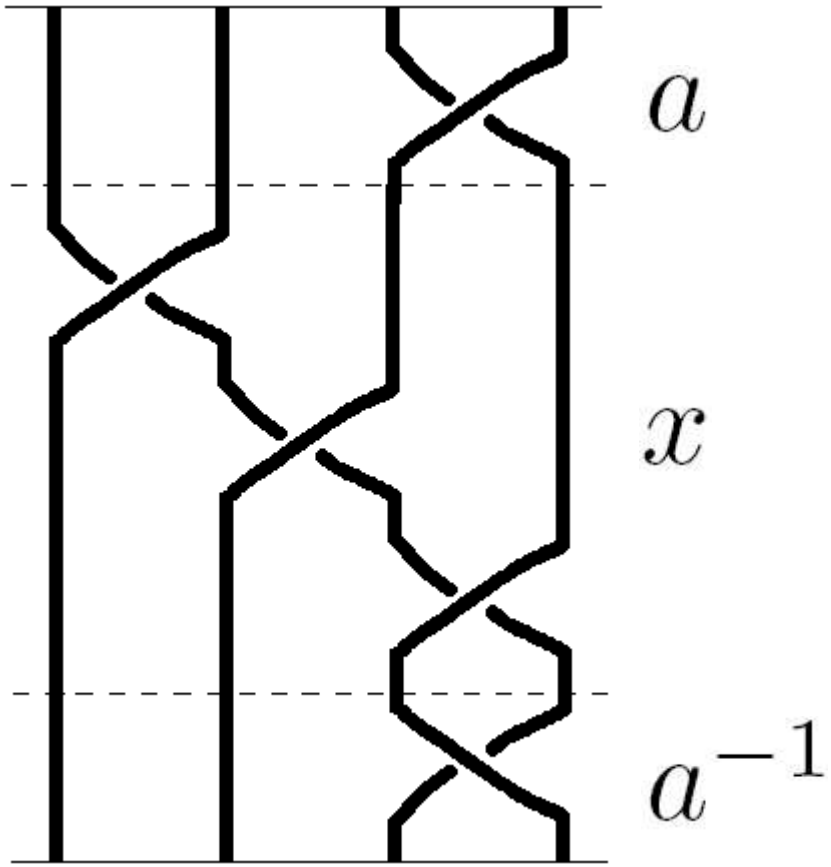


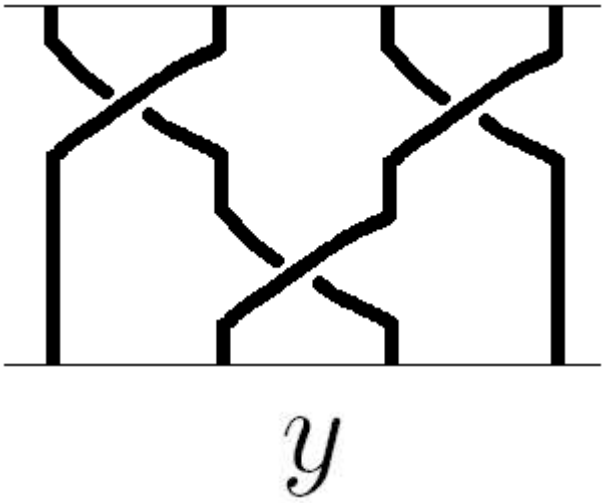
y

ブレイド群を利用した公開鍵暗号系

◆ブレイド群における一方向関数

$y = axa^{-1}$ となる a が存在するとき x と y は共役

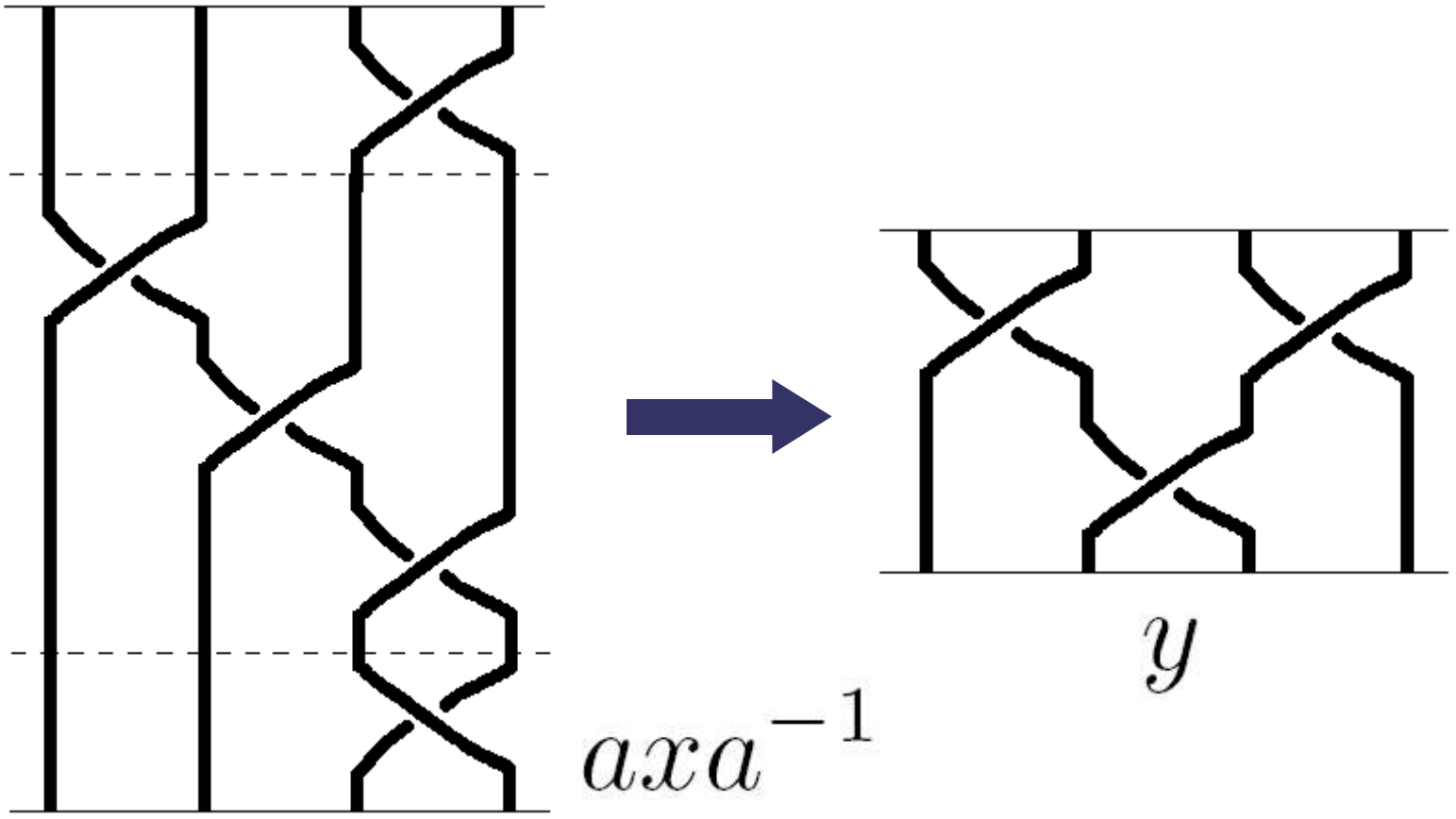




ブレイド群を利用した公開鍵暗号系

◆ブレイド群における一方向関数

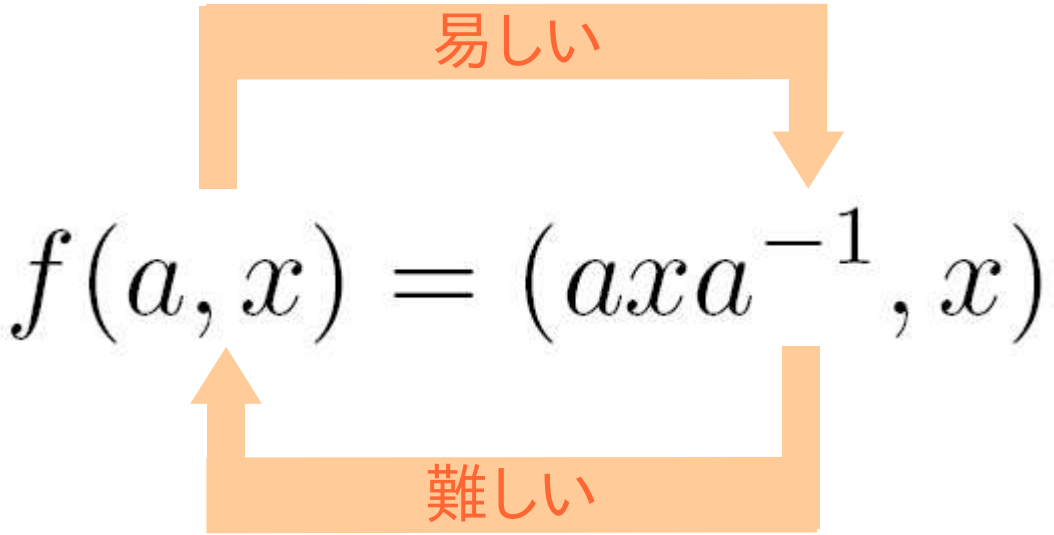
$y = axa^{-1}$ となる a が存在するとき x と y は共役



ブレイド群を利用した公開鍵暗号系

◆ブレイド群における一方向関数

$y = axa^{-1}$ となる a が存在するとき x と y は共役



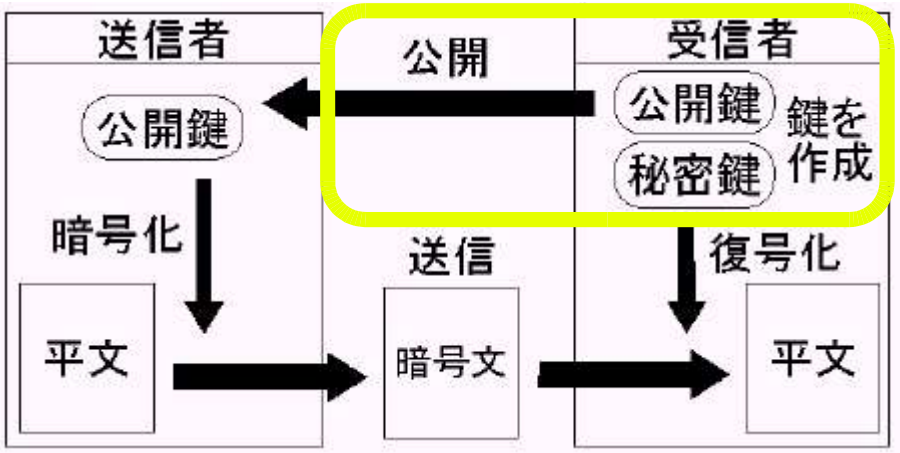
ブレイド群を利用した公開鍵暗号系

◆鍵の生成

公開鍵 1. $l+r$ 本の紐からなるブレイド x

秘密鍵 2. x の左側の l 本の紐からなるブレイド a

公開鍵 3. $y = axa^{-1}$ を計算

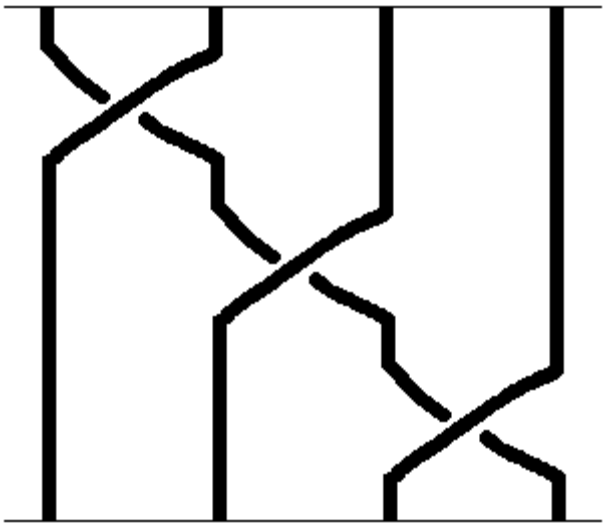


ブレイド群を利用した公開鍵暗号系

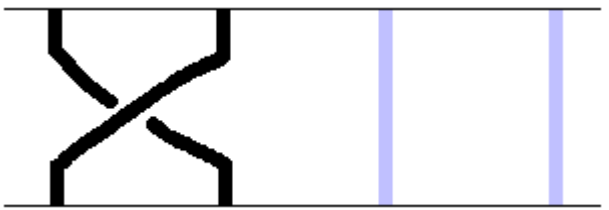
◆鍵の生成

公開鍵 1. $l+r$ 本の紐からなるブレイド x

秘密鍵 2. x の左側の l 本の紐からなるブレイド a



$x = \sigma_1\sigma_2\sigma_3$



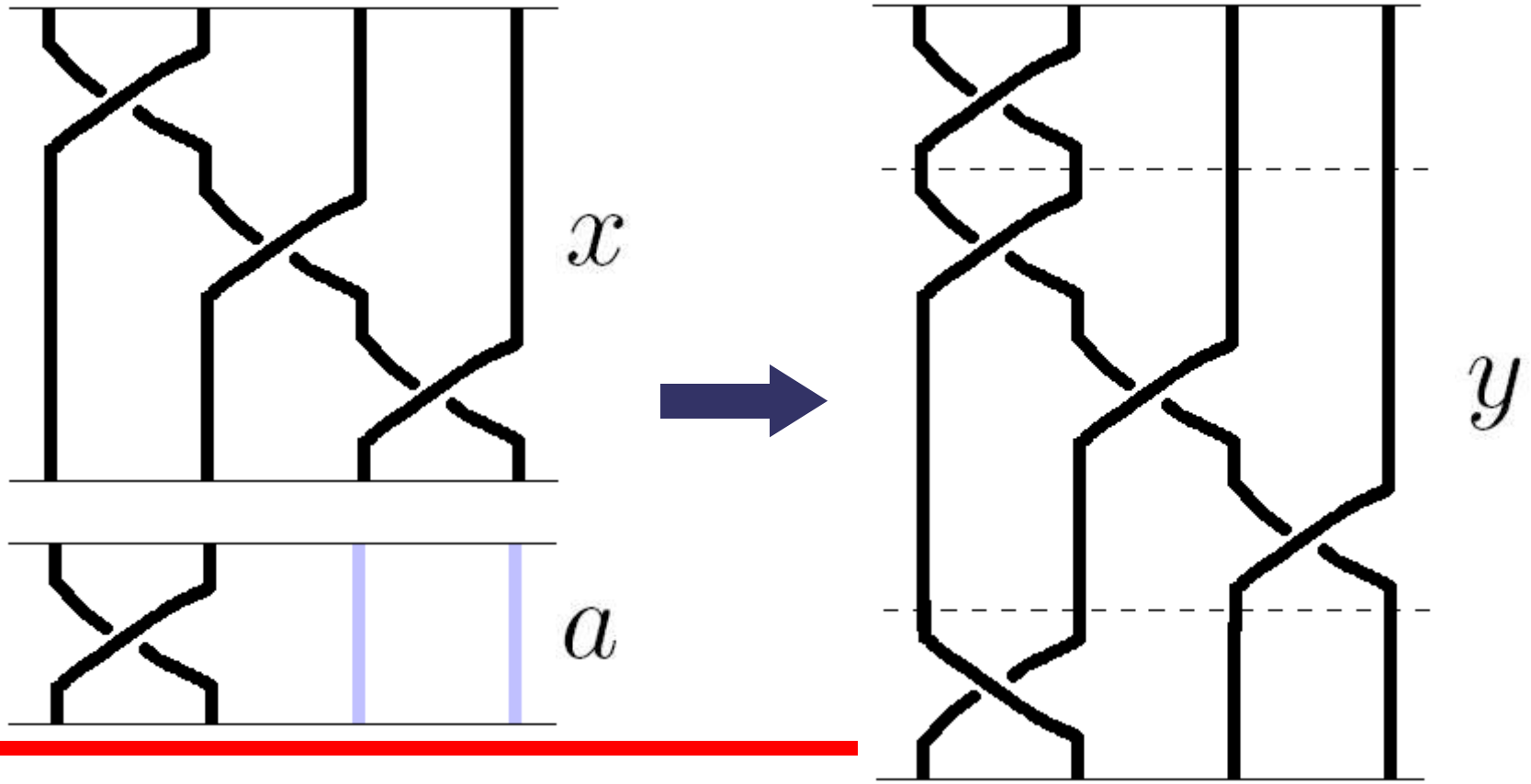
$a = \sigma_1$

ブレイド群を利用した公開鍵暗号系

xとyは共役
→ xとyからaは求められない

◆鍵の生成

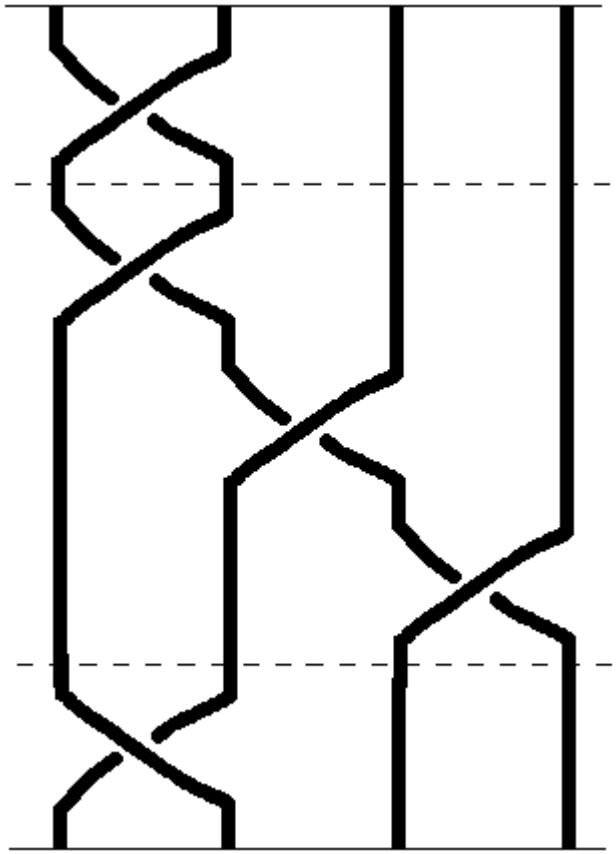
公開鍵 3. $y = axa^{-1}$ を計算



ブレイド群を利用した公開鍵暗号系

◆鍵の生成

公開鍵 3. $y = axa^{-1}$ を計算



$$x = \sigma_1\sigma_2\sigma_3$$

$$a = \sigma_1$$

$$y = axa^{-1} = \sigma_1\sigma_1\sigma_2\sigma_3\sigma_1^{-1}$$

標準形

$$\sigma_1^{-1}\sigma_2^{-1}\sigma_3^{-1}\sigma_1^{-1}\sigma_2^{-1}\sigma_1^{-1}$$
$$\sigma_3\sigma_1\sigma_2\sigma_3\sigma_2\sigma_1\sigma_1\sigma_2\sigma_3$$

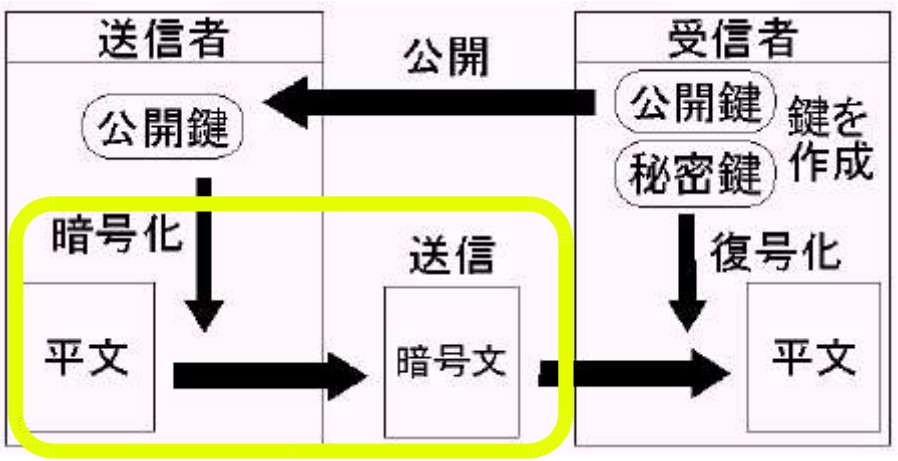
ブレイド群を利用した公開鍵暗号系

◆暗号化　公開鍵 x と y を受信

1. x の右側の r 本の紐からなるブレイド b

送信　2. $c = bxb^{-1}$

送信　3. $d = H(byb^{-1}) \oplus m$　m:送信したい平文



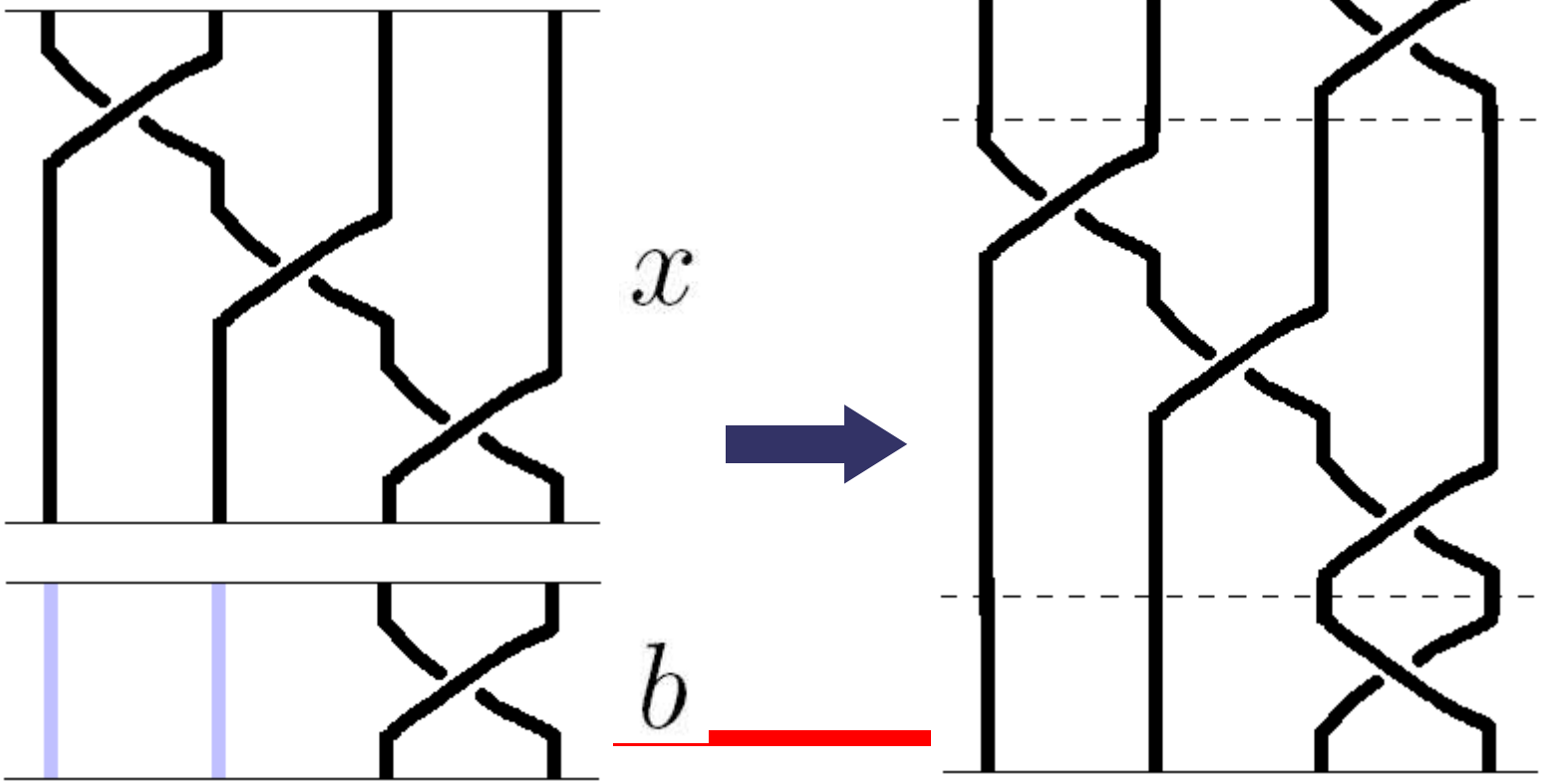
ブレイド群を利用した公開鍵暗号系

◆暗号化

xとcは共役
→ xとcからbは求められない

1. x の右側の r 本の紐からなるブレイド b

送信 2. $c = bxb^{-1}$



ブレイド群を利用した公開鍵暗号系

◆暗号化

送信　3.　$d = H(byb^{-1}) \oplus m$　m:送信したい平文

ハッシュ関数

- 原文を固定長のデータに要約
- ハッシュ値から原文を再現することはできない
- 同じハッシュ値を持つ異なるデータを作成することは困難

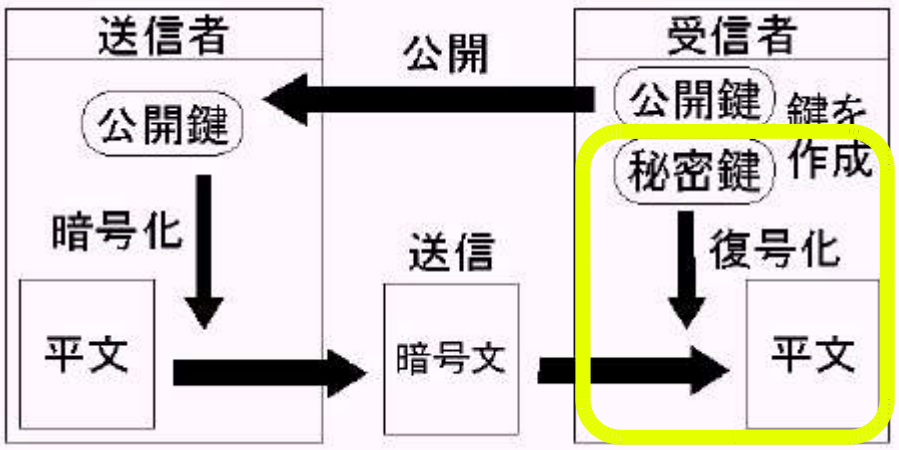
ブレイド群を利用した公開鍵暗号系

◆復号化　暗号文ｃとｄを受信

1. $m = H(aca^{-1}) \oplus d$

秘密鍵

x の左側の l 本の紐からなるブレイド a



ブレイド群を利用した公開鍵暗号系

◆復号化のしくみ

正しくもとの文に復号化できるのか??

$$m = H(aca^{-1}) \oplus d$$

$c = bxb^{-1}$ より

$$aca^{-1} = abxb^{-1}a^{-1}$$

$$= baxa^{-1}b^{-1}$$

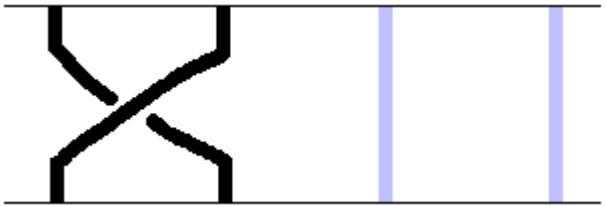
ブレイドは可換でないのでは??

$y = axa^{-1}$ より

$$= byb^{-1}$$

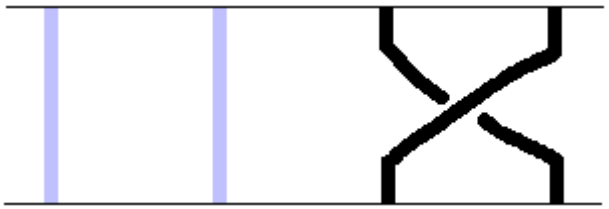
ブレイド群を利用した公開鍵暗号系

◆復号化のしくみ



a

x の左側の l 本の紐からなるブレイド

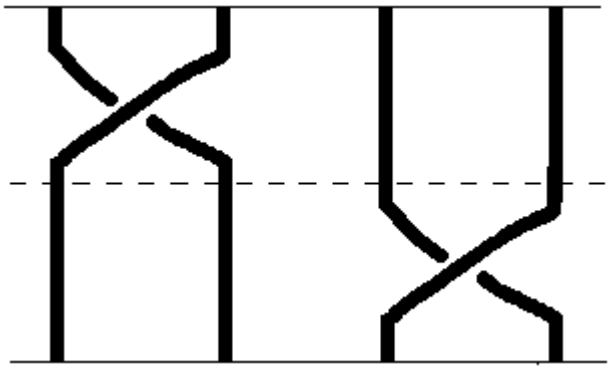


b

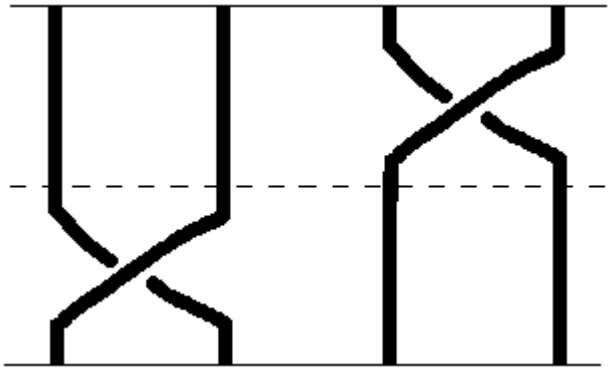
x の右側の r 本の紐からなるブレイド

ブレイド群を利用した公開鍵暗号系

◆復号化のしくみ



$a\,b$



$b\,a$

この暗号系においてaとbは可換である

$$aca^{-1} = byb^{-1}$$

ブレイド群を利用した公開鍵暗号系

◆復号化のしくみ

$$\boxed{m = H(aca^{-1}) \oplus d}$$

$aca^{-1} = byb^{-1}$ より

$$H(aca^{-1}) \oplus d$$

$$= H(byb^{-1}) \oplus d$$

$d = H(byb^{-1}) \oplus m$ より

$$= H(byb^{-1}) \oplus H(byb^{-1}) \oplus m = m$$

元の文に正しく
復号化されている

目次

研究の成果と今後の課題

- ブレイド群を利用した公開鍵暗号系を実装
- 性能評価実験

実行環境

《CPU》Pentium 4(2.7GHz)

《OS》Fedora Core 2 Linux

《搭載メモリ》1GB

《コンパイラ》g++ 3.3.3

研究の成果と今後の課題

◆ブレイド群を利用した公開鍵暗号系を実装

◆性能評価実験

実験方法

- 平文を固定して鍵の生成･暗号化･復号化に要する時間を測定
- 鍵をランダムに生成して20回測定したときの平均を求めた

研究の成果と今後の課題

◆実験結果

紐の本数とブレイドの長さ(生成元の個数)を変えた場合

長さ/本数	4	5
50	0.03	0.15
100	0.06	0.70
200	0.09	2.23
300	0.23	5.30
400	0.45	8.52
500	0.64	14.80

（単位:秒）

研究の成果と今後の課題

◆問題点

現在紐の本数を6本以上にできない

現在使われているRSA暗号系と同程度にまで
安全性を高めるためには紐の本数を増やす必要あり

研究の成果と今後の課題

◆問題点

現在紐の本数を6本以上にできない

原因:

- 標準形を構成するアルゴリズムは基本的に多項式時間であるが今回の実装では一部が指数時間になっている
- その部分の高速なアルゴリズムは知られている

研究の成果と今後の課題

◆今後の課題

- 標準形を構成するプログラムの改良
- 紐の本数を増やして暗号系の安全性を高める
- RSA暗号系との処理速度比較実験

おわり