CryptoVerifによるブロック暗号の安全性自動証明

○ 大熊 建司
村谷 博文
花谷 嘉一
古田憲一郎

(株) 東芝 研究開発センター

目次

- はじめに
- 自動証明の共通鍵暗号への適用
- Luby-Rackoff暗号の擬似ランダム性
- ゲーム法の適用
- CryptoVerifを適用
- おわりに

はじめに

- **安全性証明自動化の試み**
  - 形式的証明の自動化
    - 人間では処理困難な問題の取扱い
    - 証明誤りをなくす

- **自動証明のツール**
  - CryptoVerif
    - B.Blanchet
  - Applpi -- Pi-calculus in Coq
    - R.Affeldt, N.Kobayashi

はじめに

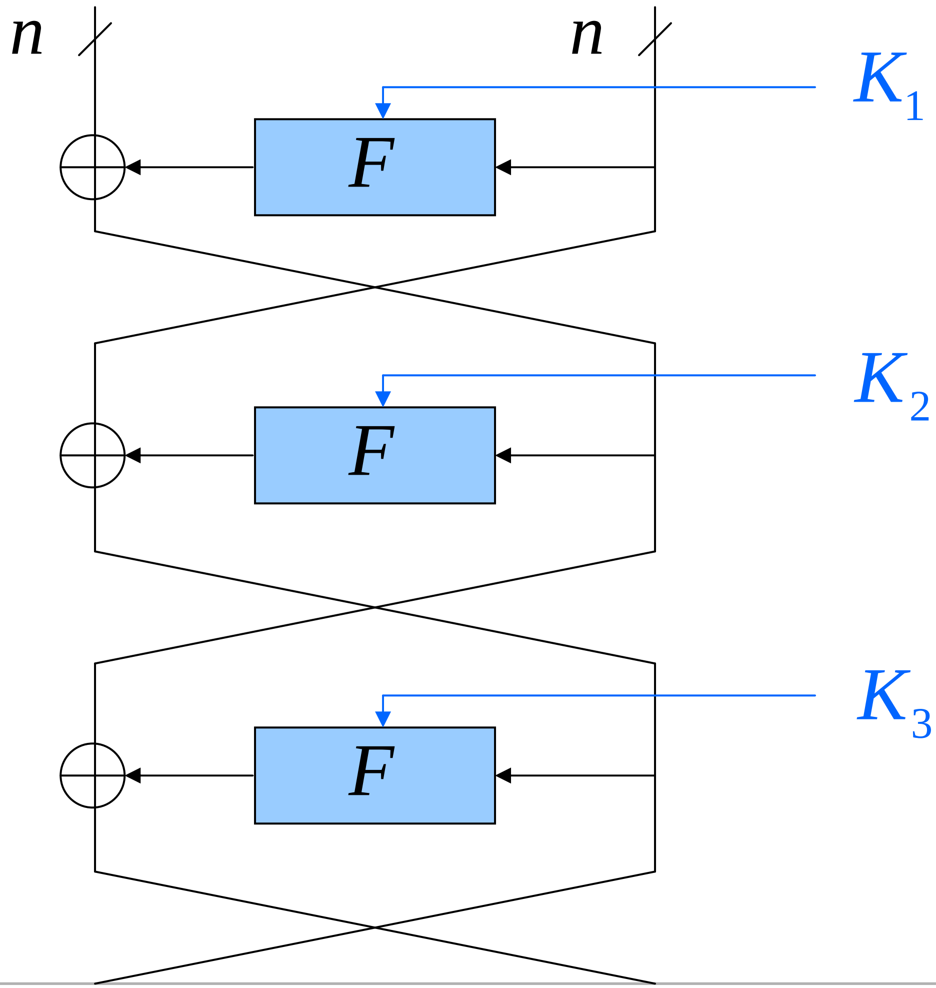
- **共通鍵暗号系に対する自動証明**
  - MACの安全性
    - CryptoVerif
      - 荒井、岡崎、不破
  - Switching Lemma
    - Applpi (Pi-calculus in Coq)
      - R.Affeldt, N.Kobayashi

Luby-Rackoff暗号に対する自動証明を適用

証明は可能か？
証明の障害は？

Feistel構造とLuby-Rackoff暗号

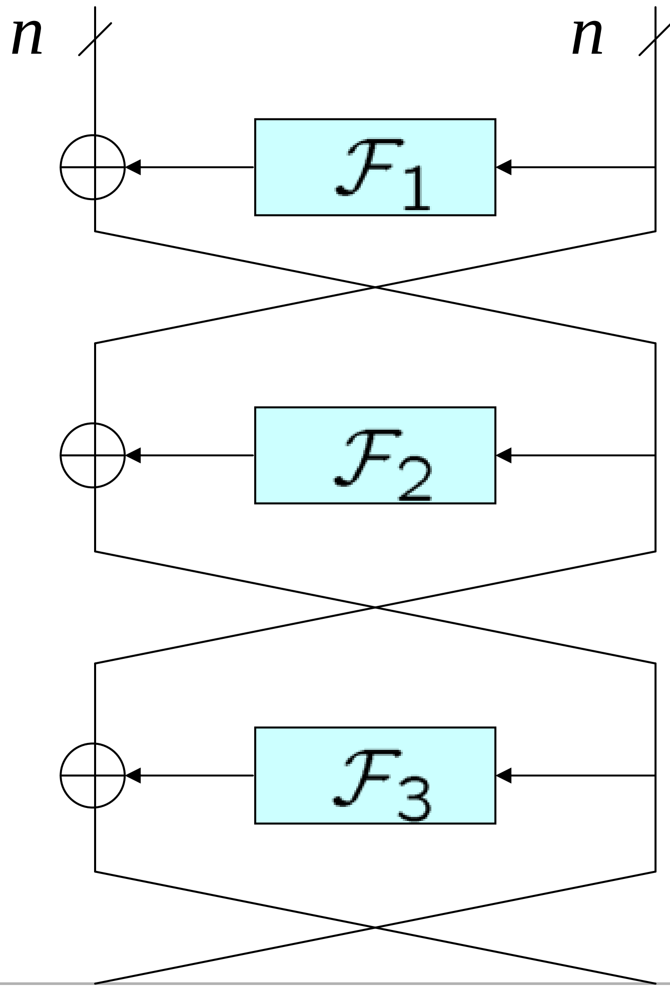
- **Feistel構造**
  - DESに代表される共通鍵ブロック暗号の構造（involution型）



鍵によらず全単射（置換）

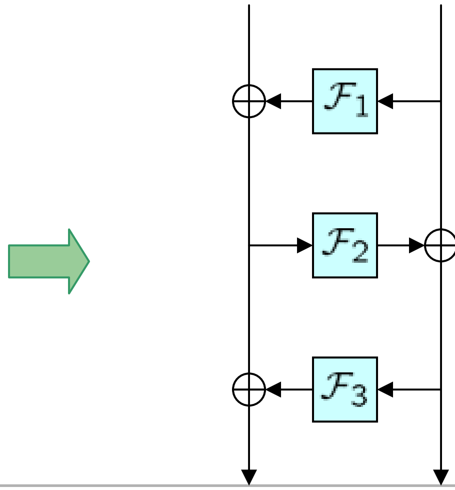
Feistel構造とLuby-Rackoff暗号

- **Luby-Rackoff暗号**
  - 鍵付きF関数を、擬似ランダム関数に置換え



Luby-Rackoff 1988

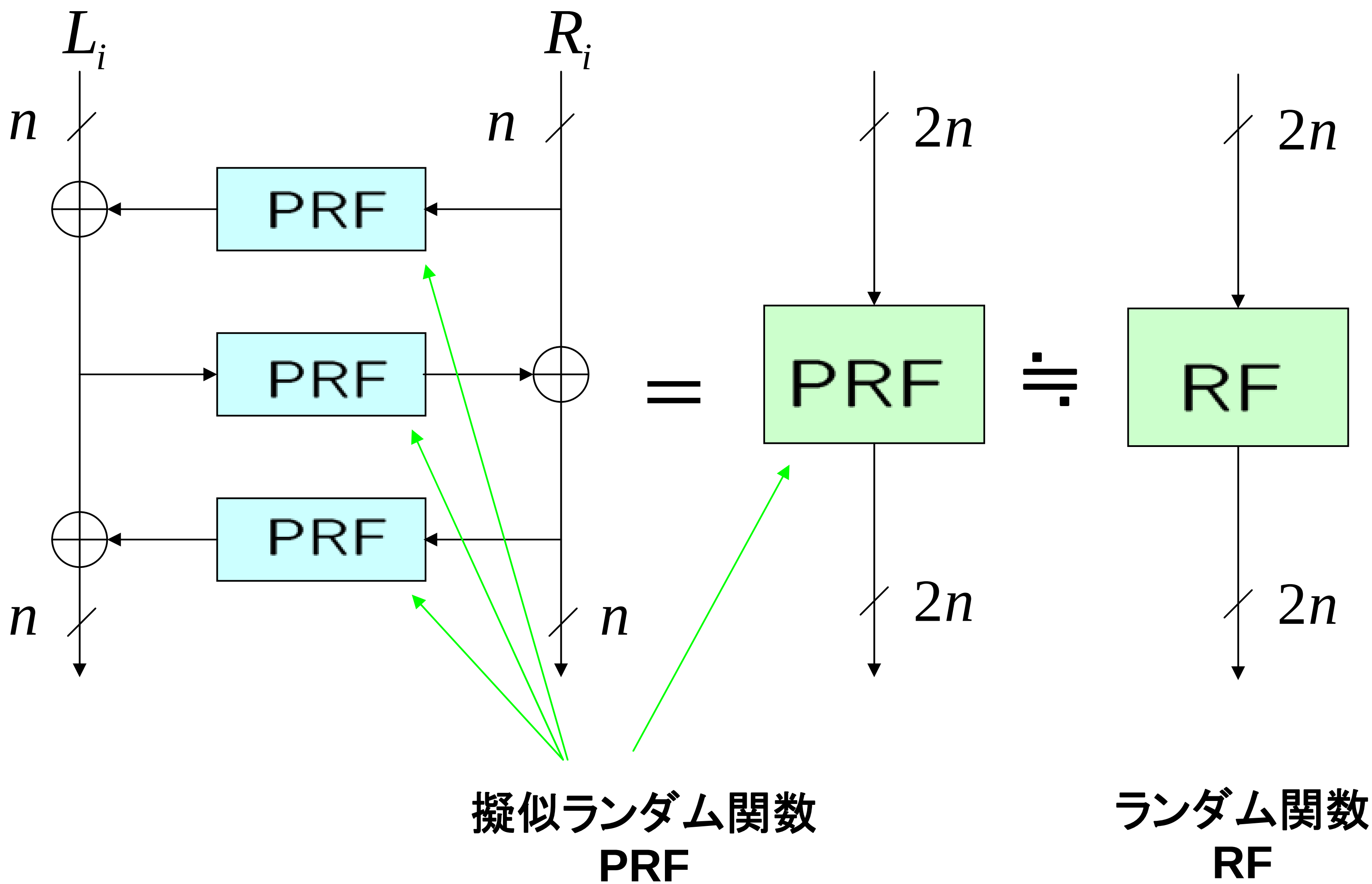
$\mathcal{F}_i$ が擬似ランダムのとき、
３段で擬似ランダム
４段で強擬似ランダム



簡単のため、
はしご型で表示

擬似ランダム性

- 「3段Feistel構造」と「ランダム関数」の差が無視できる



ランダム関数とランダム置換

- ランダム関数 (Random Function [or All functions])

$$F^m = \{f : \{0,1\}^m \to \{0,1\}^m\}$$

$$\#F^m = (2^m)^{(2^m)} = 2^{m2^m}$$

異なる入力に対して独立かつ一様ランダムに出力

- ランダム置換 (Random Permutation)

$$P^m = \{f : \{0,1\}^m \to \{0,1\}^m | f(i) \neq f(j) \ , \ \forall i \neq \forall j\}$$

$$\#P^m = (2^m)! \qquad \ll \#F^m$$

ランダム関数のうち全単射

ランダム置換との差は小さい → Switching lemma

局所ランダム性（擬似ランダム）

- *k*個の入力に対する出力分布でランダム性を判断

・鍵スペースZを持つ関数族(入出力m-bit)

$$\mathcal{F}_{\mathcal{Z}} = \{f_z : \{0,1\}^m \to \{0,1\}^m \mid f_z \in \mathcal{Z}\}$$

・入力がk個のm-bit値、出力が$\{0,1\}$の関数族G^{km}

$$G^{km} = \{g \mid g : (\{0,1\}^m)^k \to \{0,1\}\}$$

関数族$\mathcal{F}_{\mathcal{Z}}$が次の性質を満たすとき、

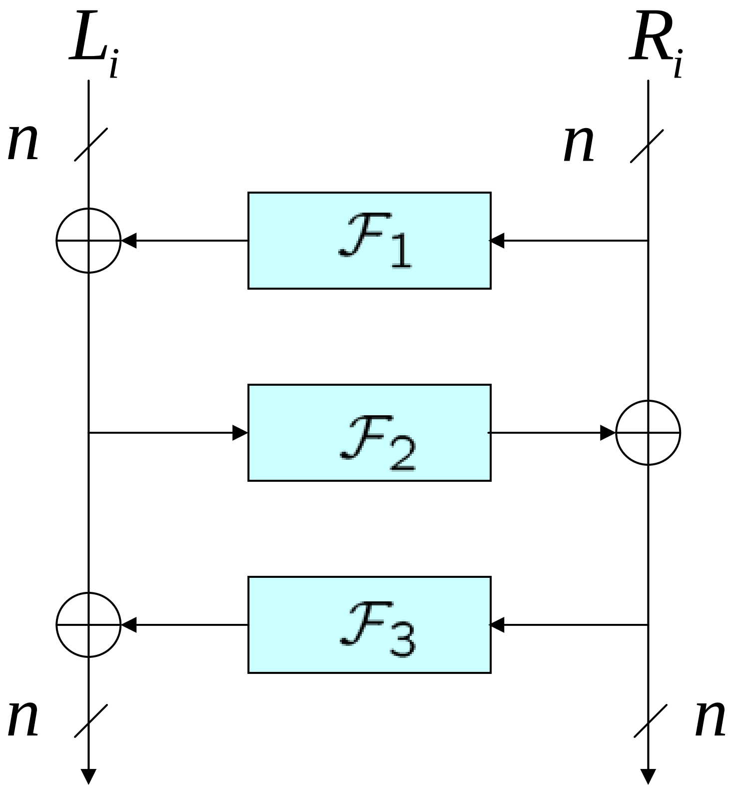
(m, k, ϵ) Locally Random Function(LRF) という

$$\forall x_1, \ldots, \forall x_k \in \{0,1\}^m \qquad \forall g : (\{0,1\}^k)^m \to \{0,1\}$$

$$\Big|\Pr[g(f(x_1), \ldots, f(x_k)) = 1,\ f \xleftarrow{R} \mathcal{F}_{\mathcal{Z}}] - \Pr[g(f(x_1), \ldots, f(x_k)) = 1,\ f \xleftarrow{R} F^m]\Big| \leq \epsilon$$

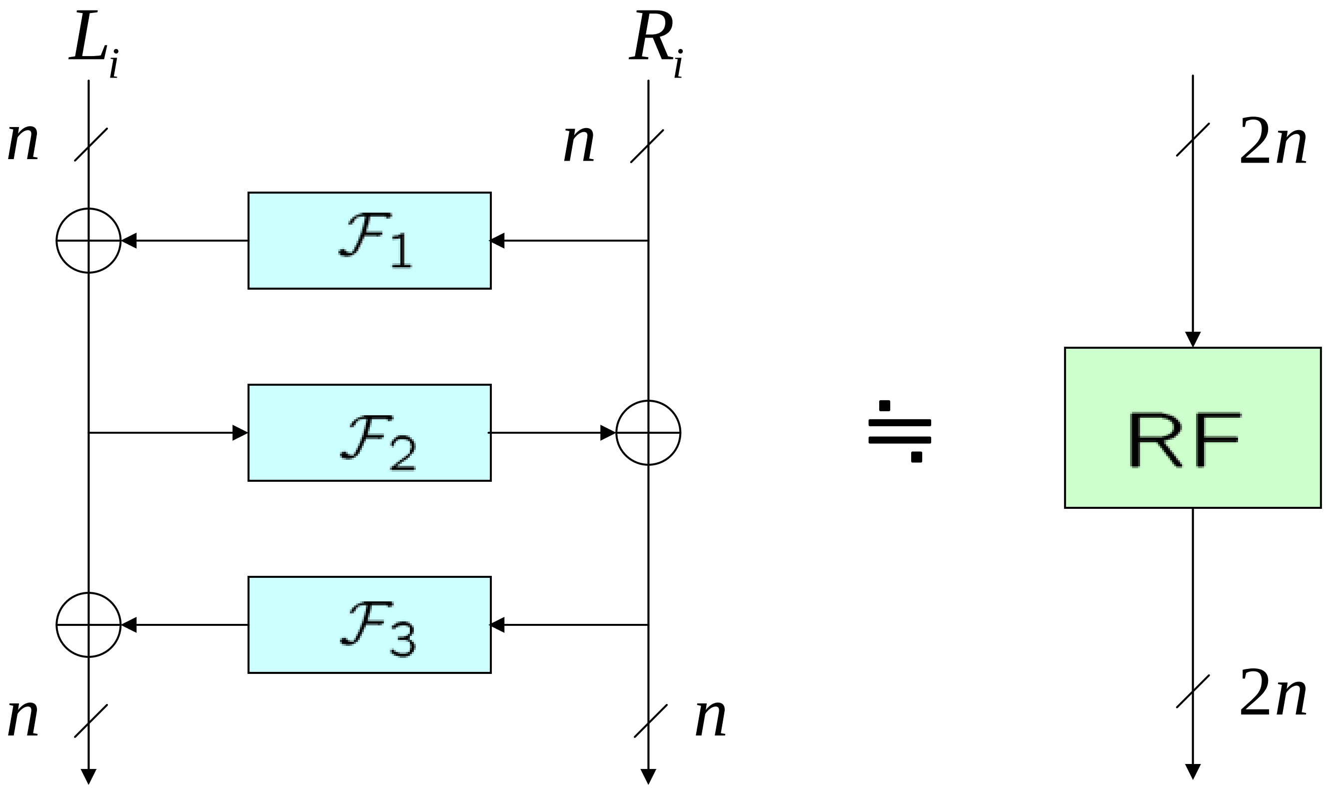
Maurer 1988

擬似ランダム関数に関する仮定



$\mathcal{F}_i$ $(i = 1, 2, 3)$ は (n, k, ϵ) LRF とする

擬似ランダム関数に関する仮定

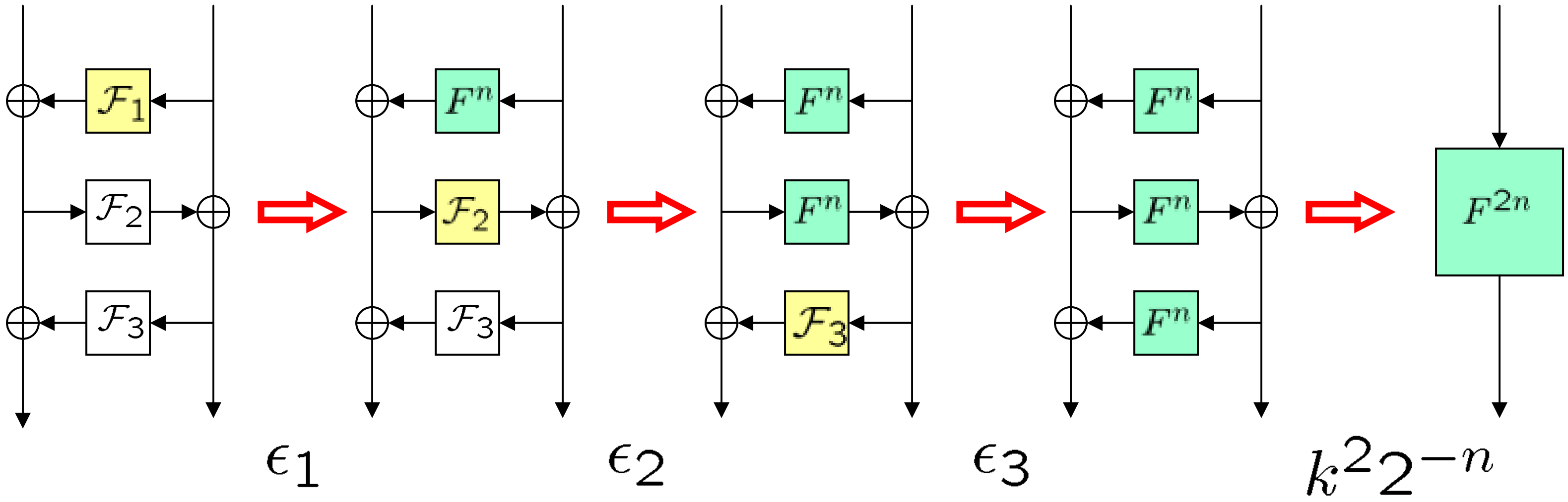


$\mathcal{F}_i \ (i = 1, 2, 3)$ は (n, k, ϵ_i) LRF とする

$\longrightarrow$ $(2n, k, \epsilon')$ LRF

$$\epsilon' = \epsilon_1 + \epsilon_2 + \epsilon_3 + k^2 2^{-n}$$

証明の概要

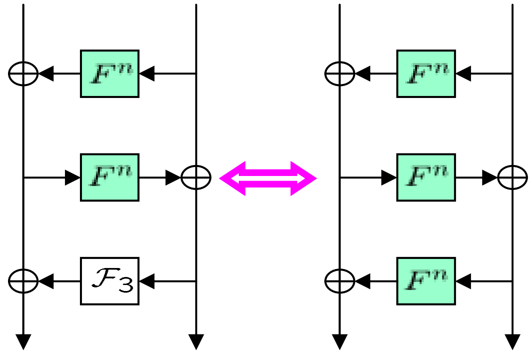

$\mathcal{F}_1$
$\mathcal{F}_2$
$\mathcal{F}_3$
F^n
F^n
$\mathcal{F}_2$
$\mathcal{F}_3$
F^n
F^n
$\mathcal{F}_3$
F^n
F^n
F^n
F^{2n}
ϵ_1
ϵ_2
ϵ_3
$k^2 2^{-n}$

観測等価性の記述

- 比較する2つの関数族間で、振る舞いが異なる場合の数え上げ方を記述

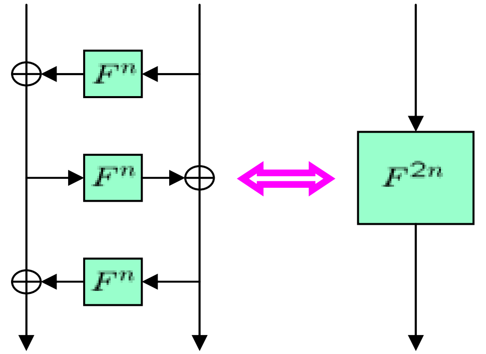
- 構造が同じで内部関数が異なる場合

  → 内部関数の振る舞いの差で記述

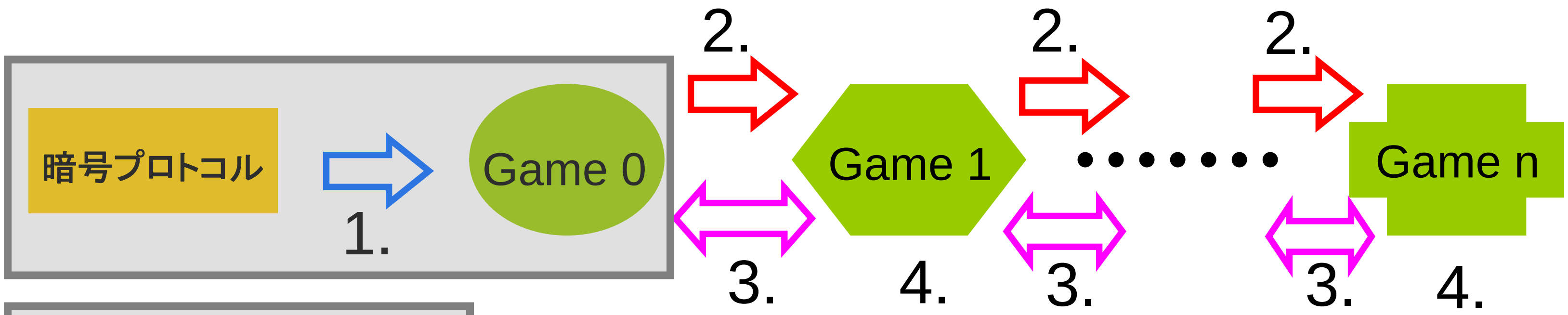


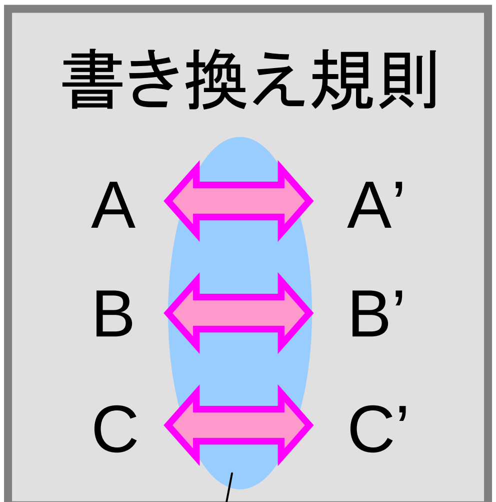
- 構造が異なる場合

  → Luby-Rackoff型がランダム関数と振る舞いが異なるケースを数え上げ

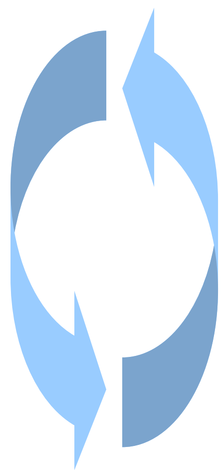


Blanchetフレームワーク





観測等価性
置き換えにより生じる差



2. 書き換え規則が適用可能か判定
   適用できる場合は書き換え,
   新たなゲームを生成.

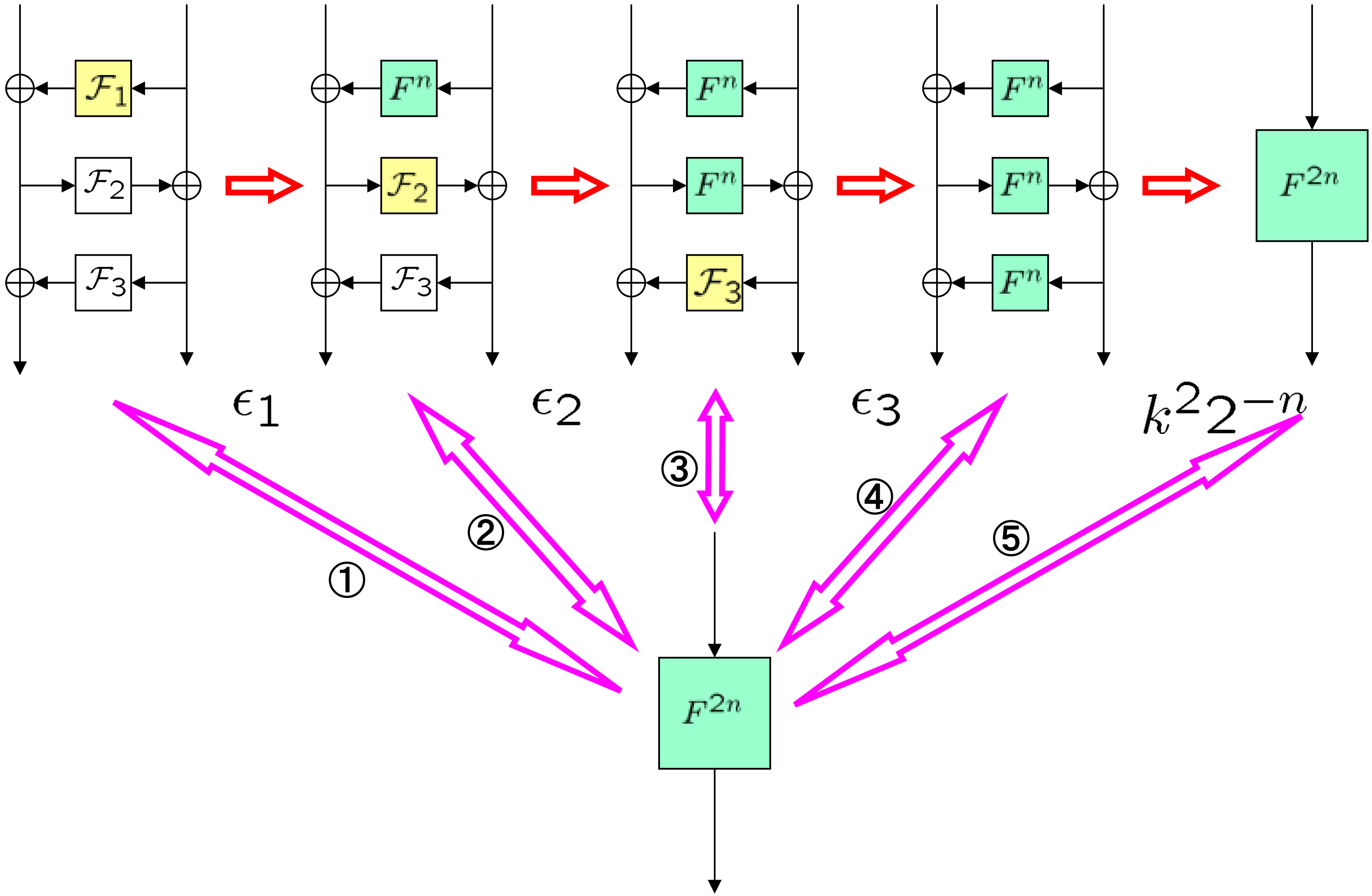
3. 適用した規則に与えられている
   観測等価性 を基に評価.

4. 安全性要件を満たすか判定.

安全な状態との差を評価.

フレームワークに基づく自動検証ツールCryptoVerif

証明の概要


$\mathcal{F}_1$
$\mathcal{F}_2$
$\mathcal{F}_3$
F^n
F^{2n}
ϵ_1
ϵ_2
ϵ_3
$k^2 2^{-n}$
①
②
③
④
⑤

CryptoVerif を適用するにあたっての問題点

- **BlanchetのCryptoVerifは**
  – ある関数のクラスから一つの関数を生成するのは困難
  → 複雑な条件分岐の扱いは不向き

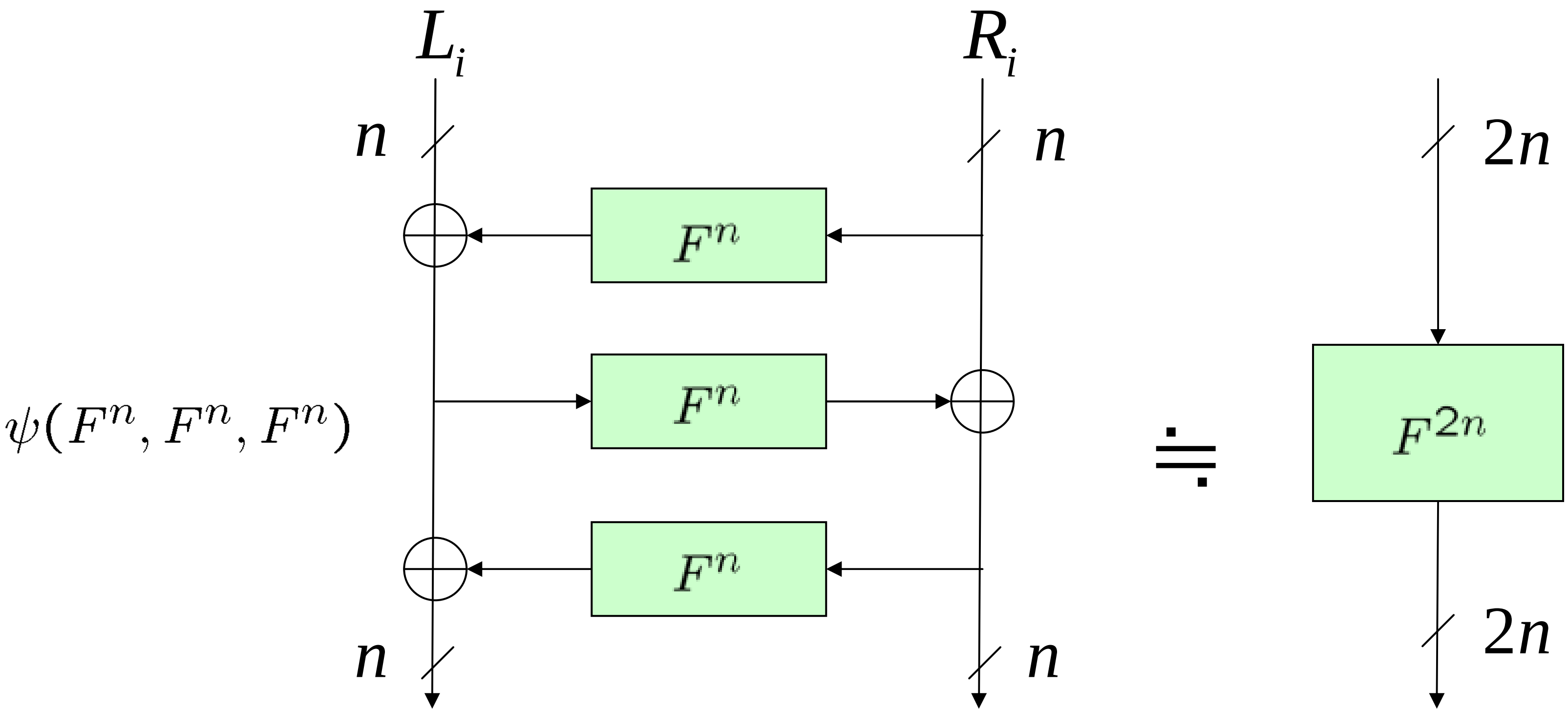
- **対策は？**

1) 機能の追加
   - prover本体の変更が必要

2) 観測等価性の利用とゲームの書き方の工夫
   - proverの変更は必要ないが…
     → 記述の書き方が悪いと終了しない

Luby-Rackoff 1988 の Lemma 1

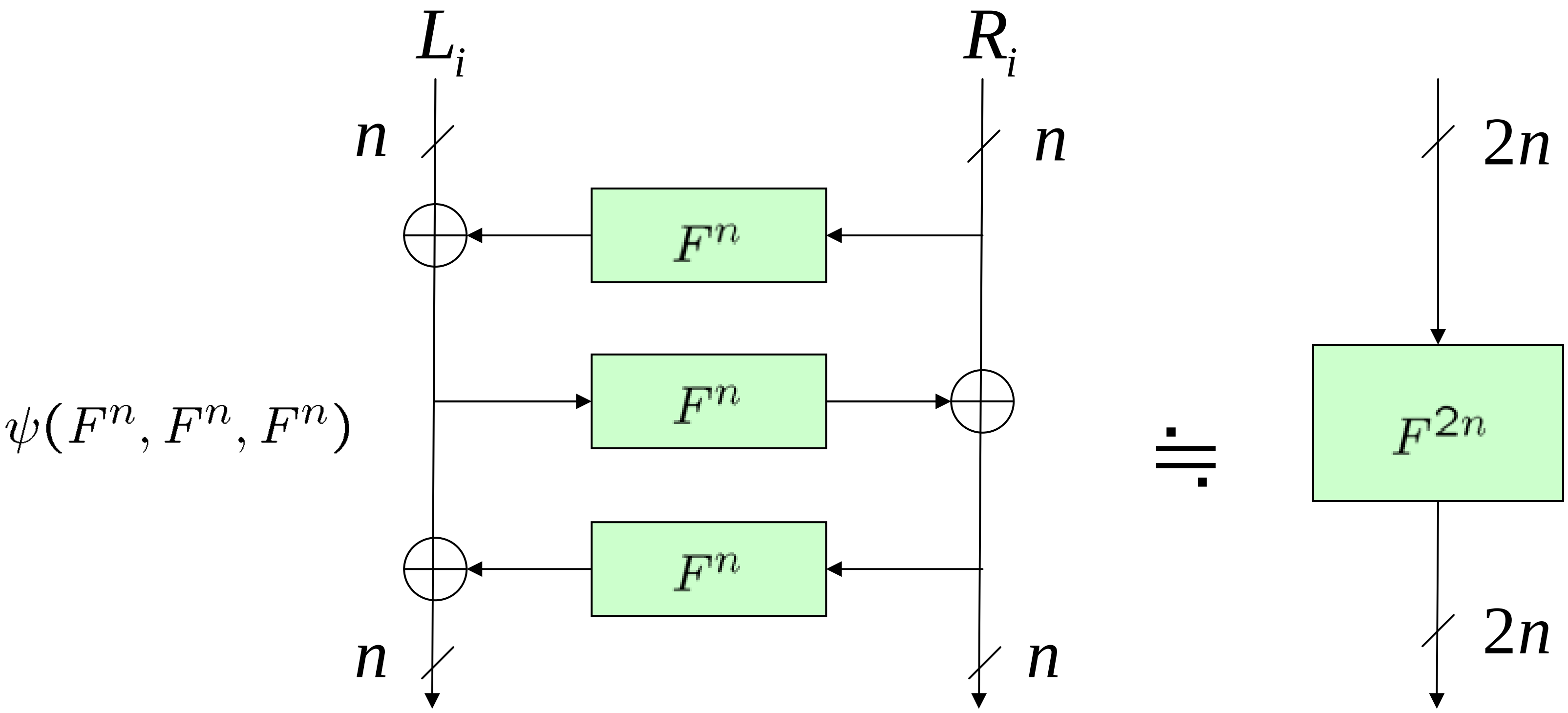
- **3段Feistel構造　～　局所擬似ランダム**



$\psi(F^n,F^n,F^n)$ は $(2n,k,k^2 2^{-n})$**LRF**

Luby-Rackoff 1988 の Lemma 1

- **3段Feistel構造　～　局所擬似ランダム**

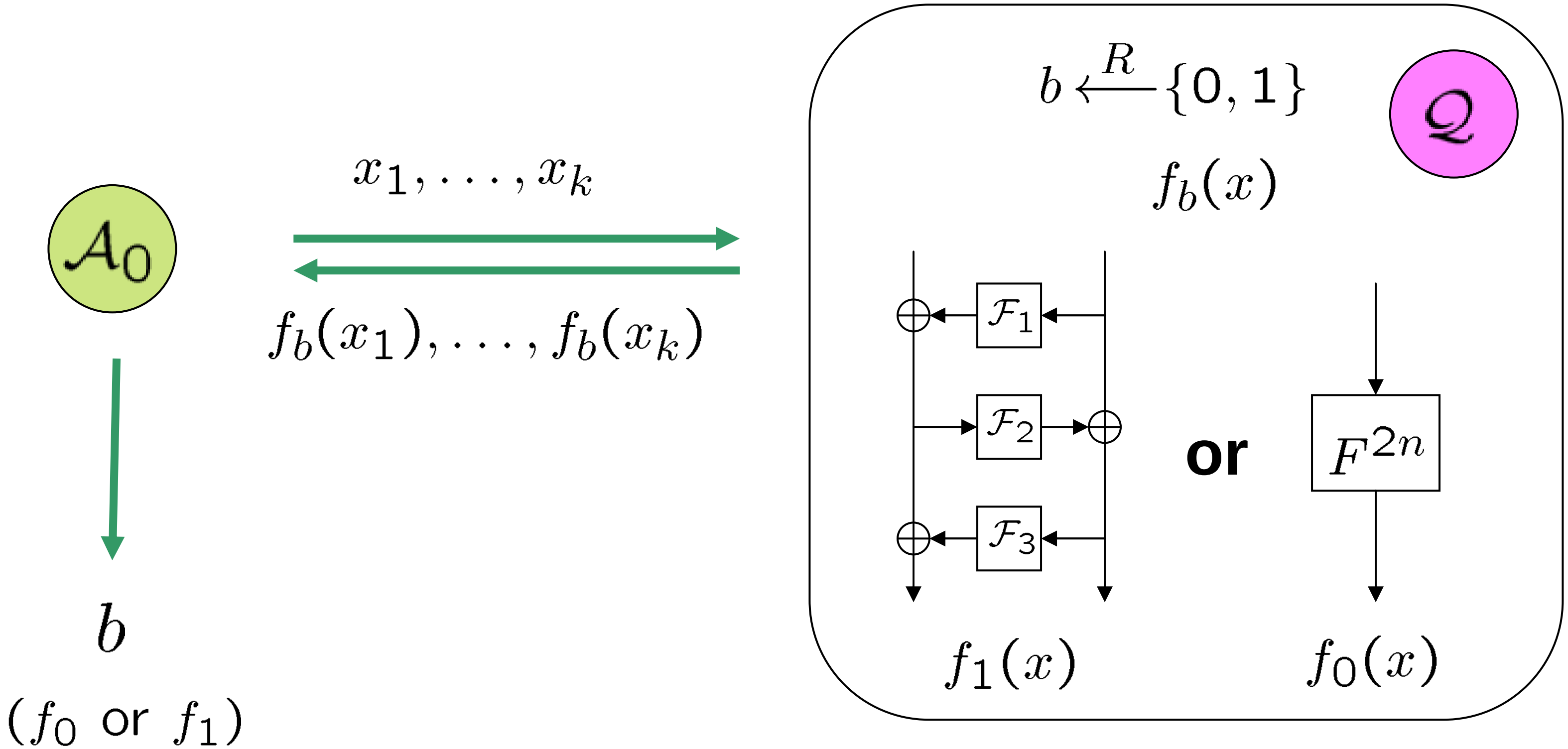


$\psi(F^n, F^n, F^n)$ は $(2n, k, k^2 2^{-n})$**LRF**

CryptoVerifでは2つの差を直接扱うことは困難
→　観測等価性とOne-session secrecy の利用

One-session Secrecyの利用

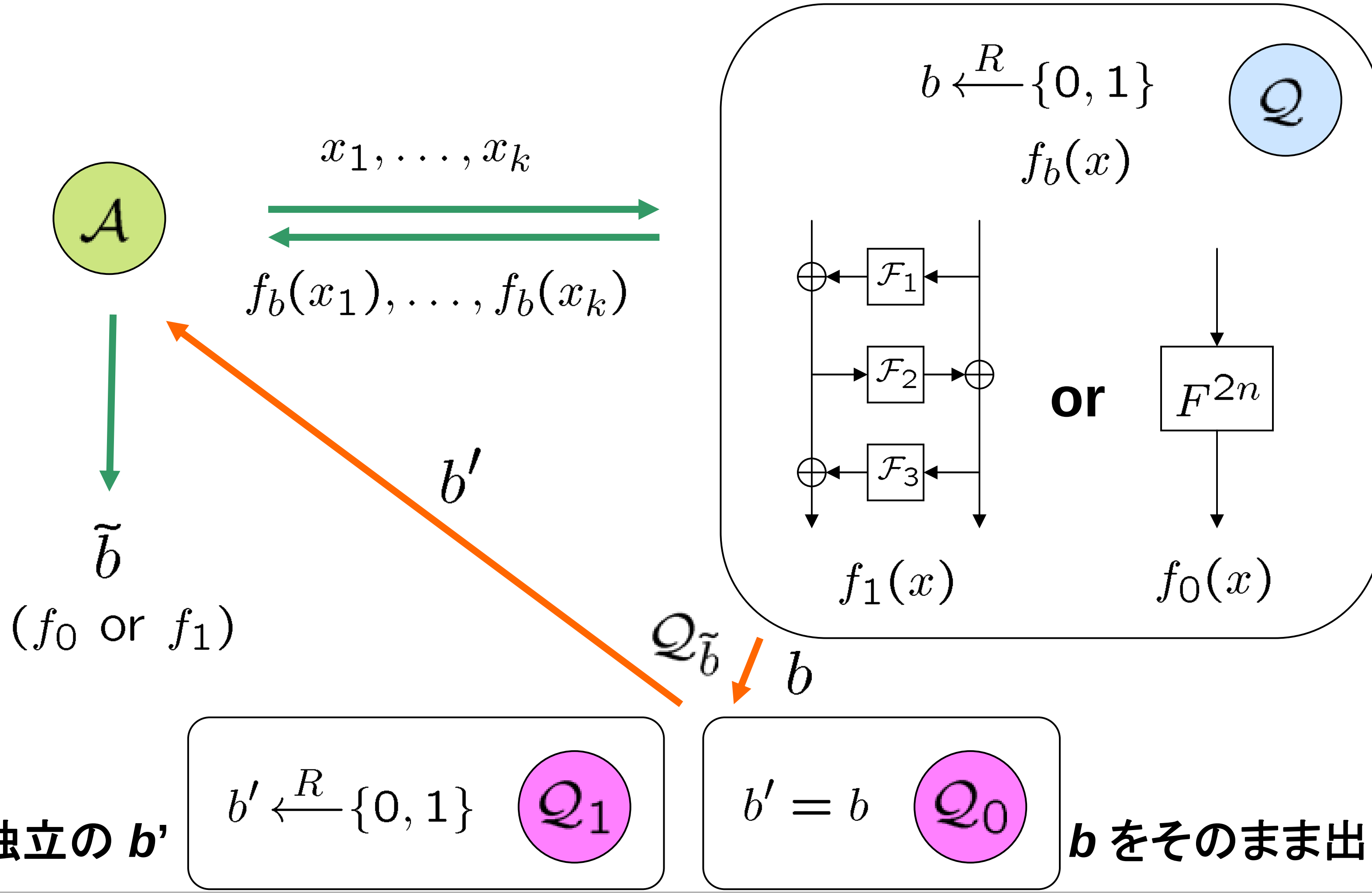
A_0: 2種類の関数 $f_b(x)$ **を識別しようとする**



One-session Secrecy

***A*: 2種類の f_b を識別**

識別できない
→ *b'*のone-session secrecy

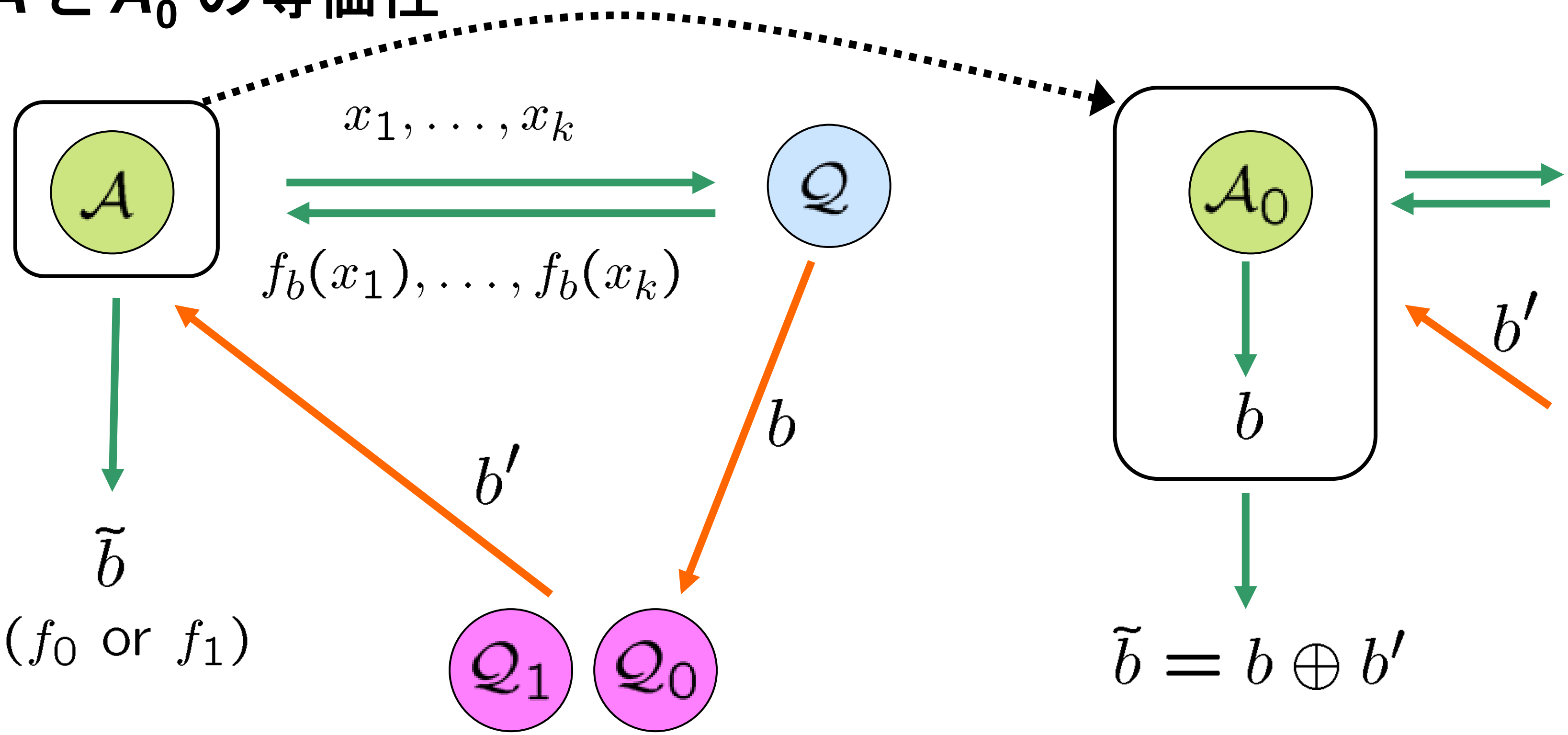


b* と独立の *b'

***b* をそのまま出力**

One Session Secrecy

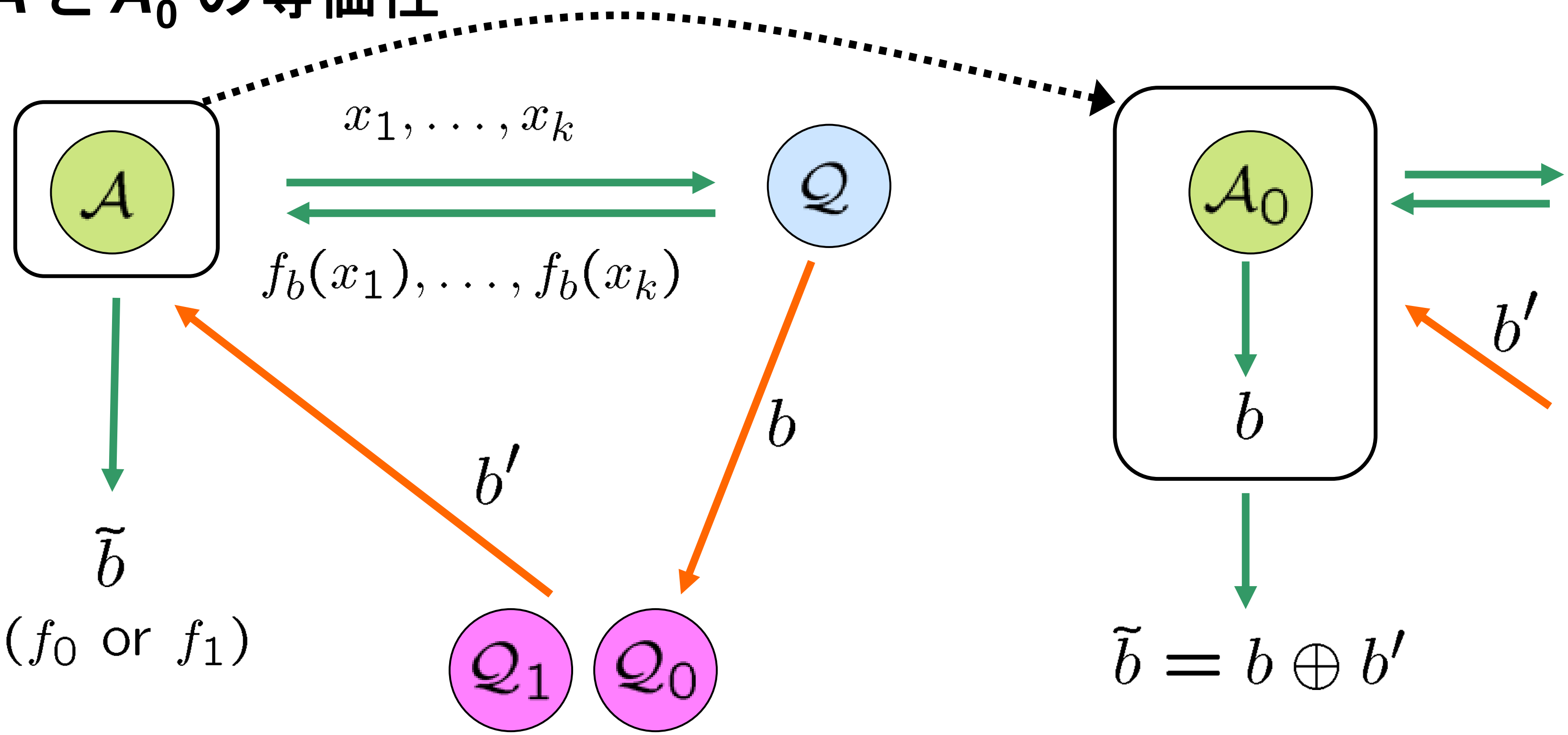
***A* と *A*$_0$ の等価性**



$$\sum_{\tilde{b}\in\{0,1\}} \left| \Pr\left[\tilde{b} \xleftarrow{R} \mathcal{A}^{\mathcal{Q},\mathcal{Q}_0}\right] - \Pr\left[\tilde{b} \xleftarrow{R} \mathcal{A}^{\mathcal{Q},\mathcal{Q}_1}\right]\right| = \frac{1}{2} \sum_{b\in\{0,1\}} \left| \Pr\left[b \xleftarrow{R} \mathcal{A}_0^{f_0}\right] - \Pr\left[b \xleftarrow{R} \mathcal{A}_0^{f_1}\right]\right|$$

One-session Secrecy

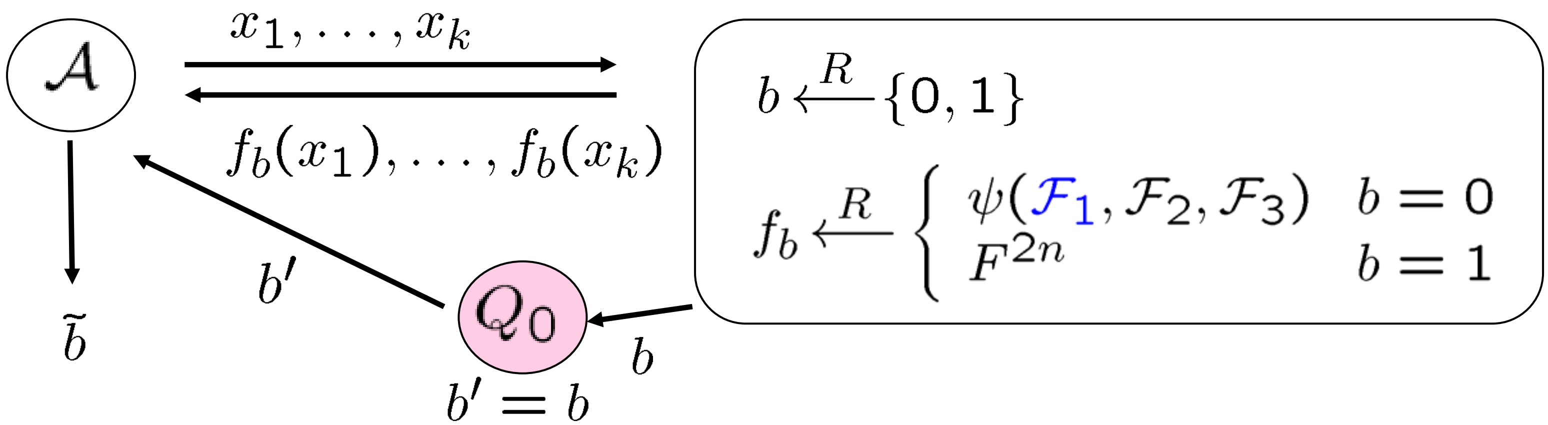
A と *A*$_0$ の等価性

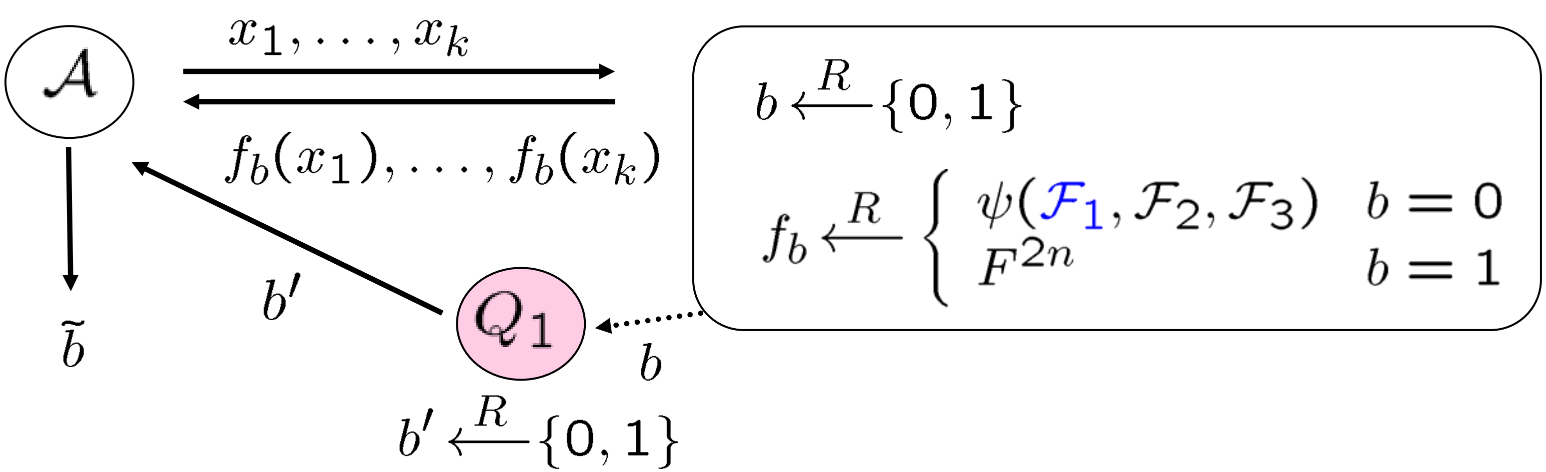


A_0 が存在 → A が存在

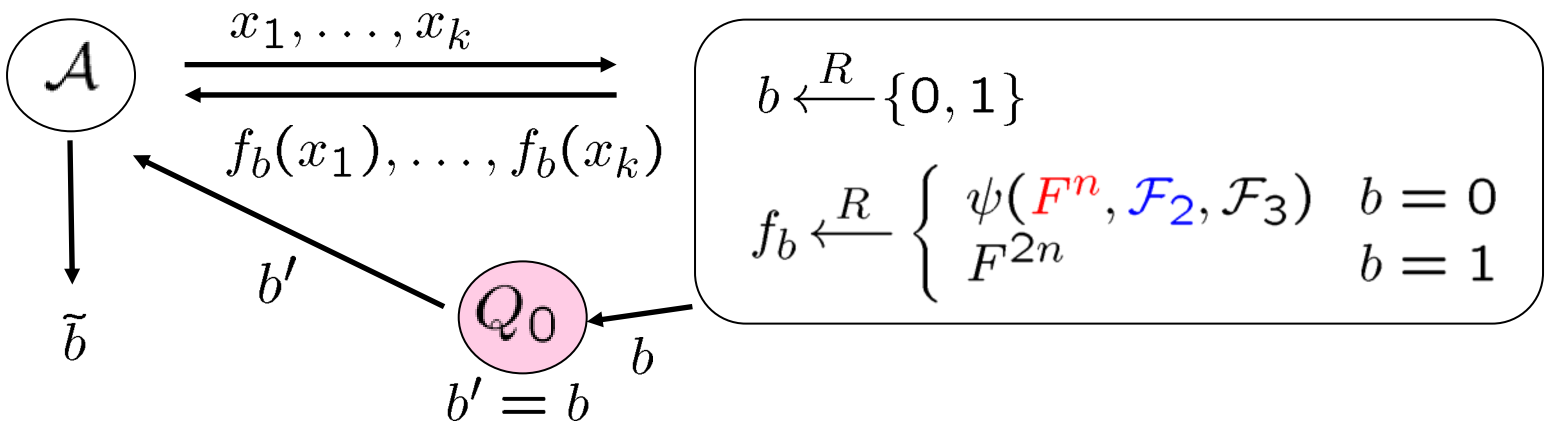
A が存在しない → A_0 が存在しない → $f_0(x)$ と $f_1(x)$ の識別不可能性

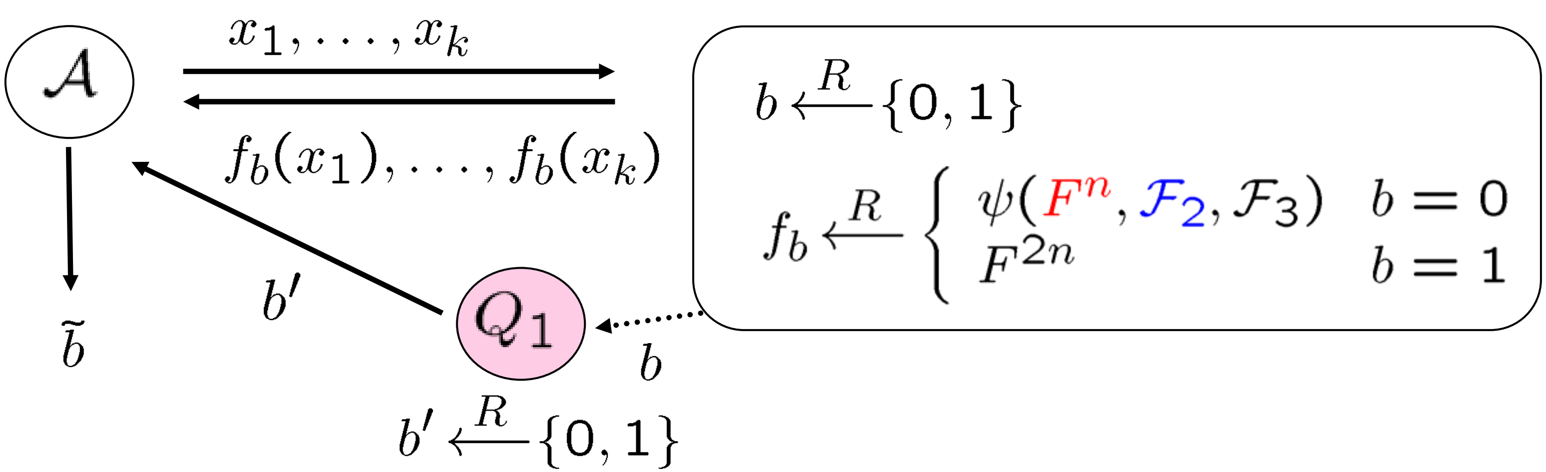
Game 0


$\mathcal{A}$
$x_1, \ldots, x_k$
$f_b(x_1), \ldots, f_b(x_k)$
$b \xleftarrow{R} \{0,1\}$
$f_b \xleftarrow{R} \begin{cases} \psi(\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3) & b = 0 \\ F^{2n} & b = 1 \end{cases}$
b'
Q_0
b
$\tilde{b}$
$b' = b$


$\mathcal{A}$
$x_1, \ldots, x_k$
$f_b(x_1), \ldots, f_b(x_k)$
$b \xleftarrow{R} \{0,1\}$
$f_b \xleftarrow{R} \begin{cases} \psi(\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3) & b = 0 \\ F^{2n} & b = 1 \end{cases}$
b'
Q_1
b
$\tilde{b}$
$b' \xleftarrow{R} \{0,1\}$

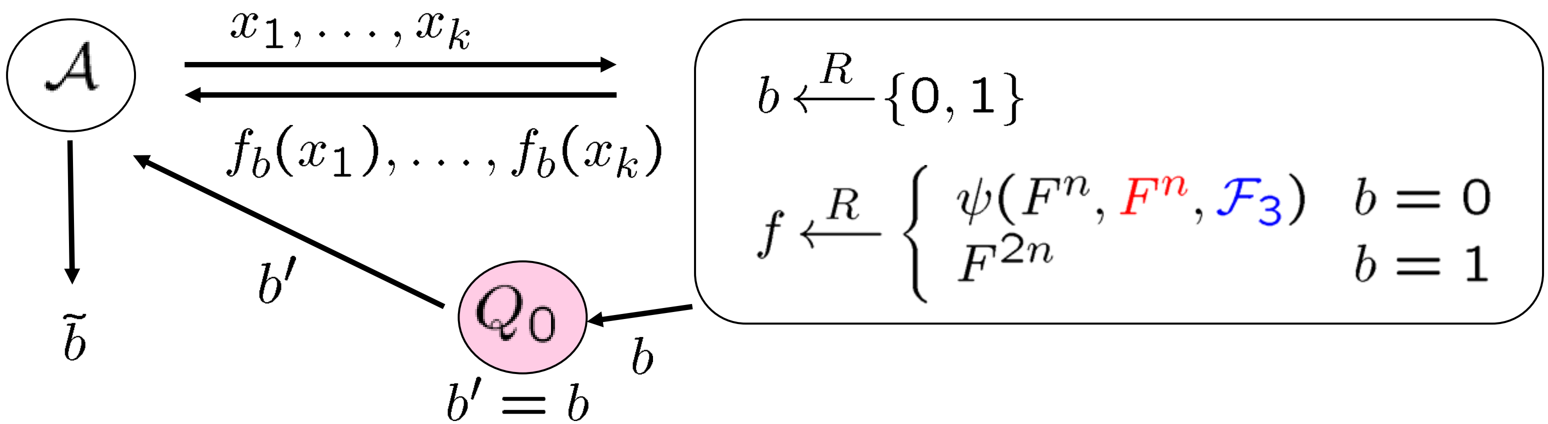
Game 1

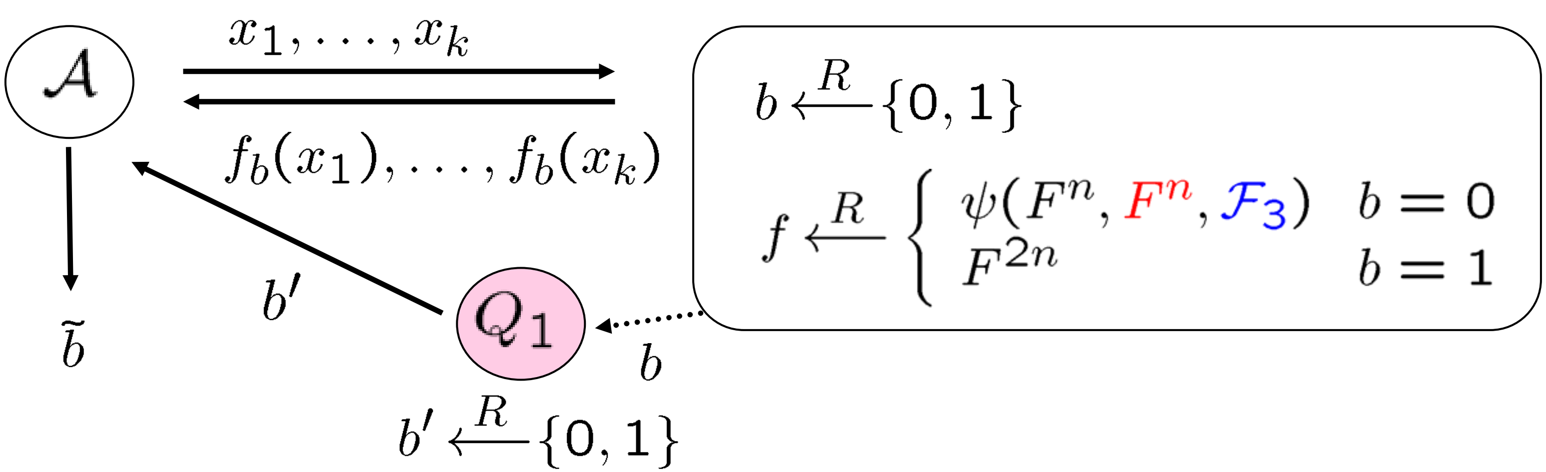




Game 0との差： ϵ_1

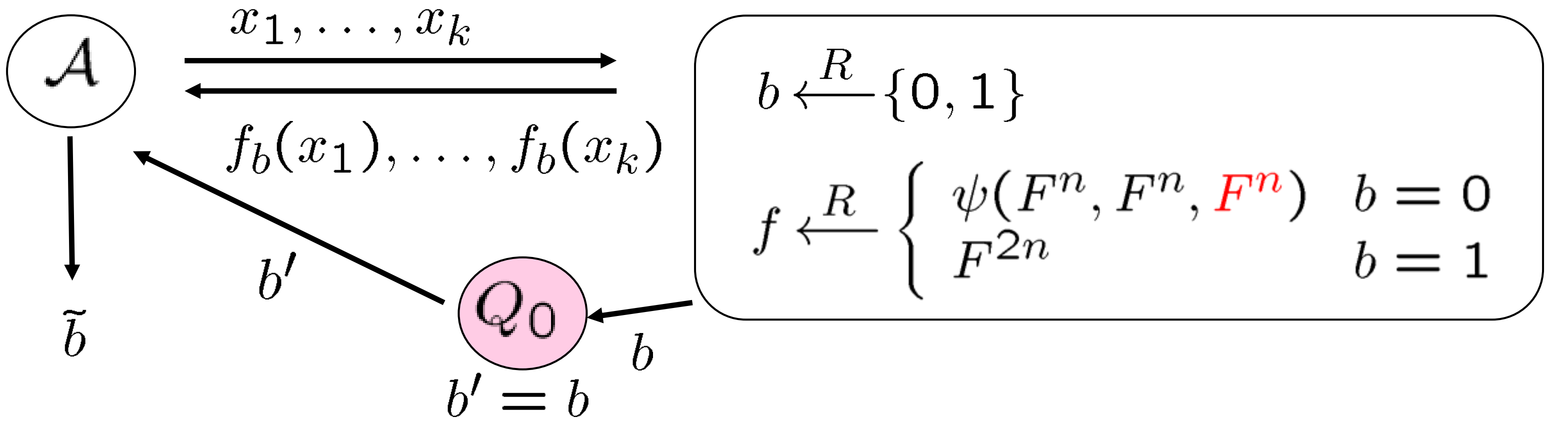
Game 2

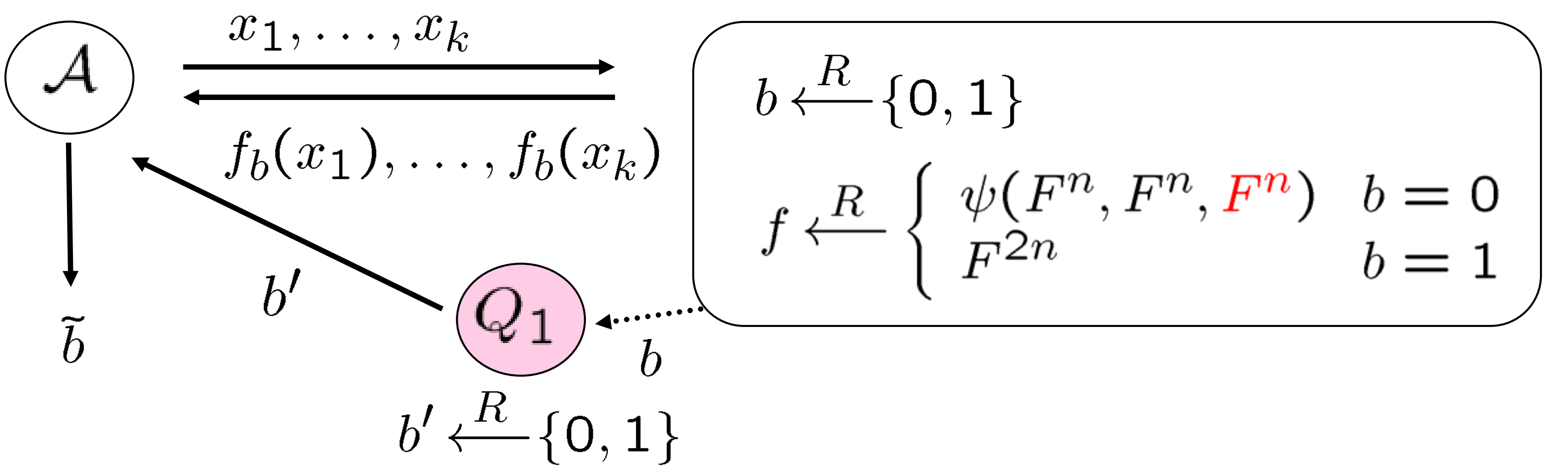




Game 1との差： ϵ_2

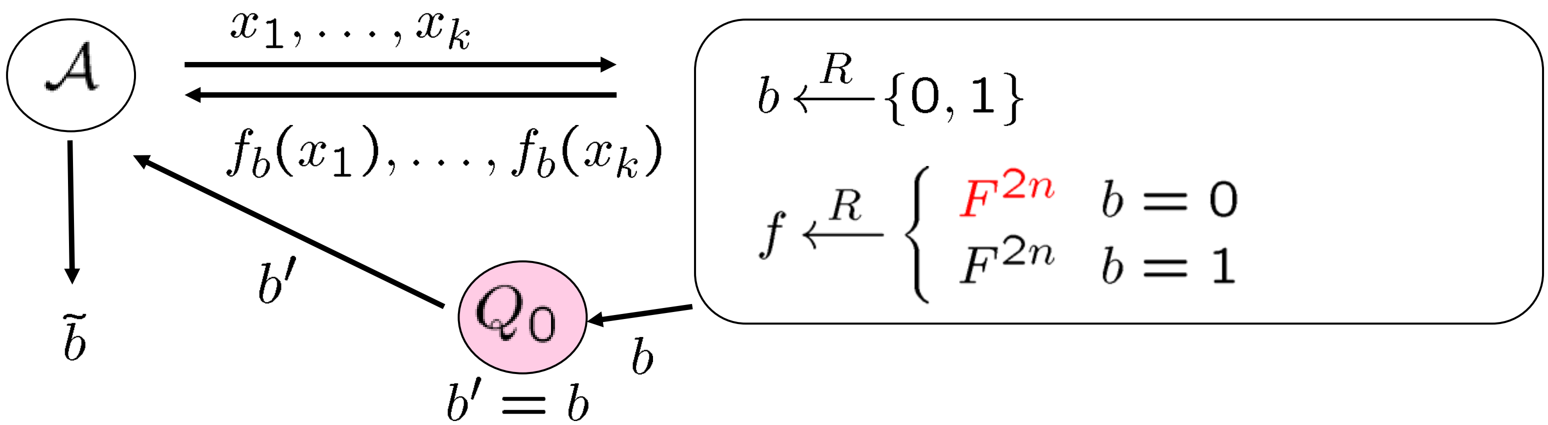
Game 3

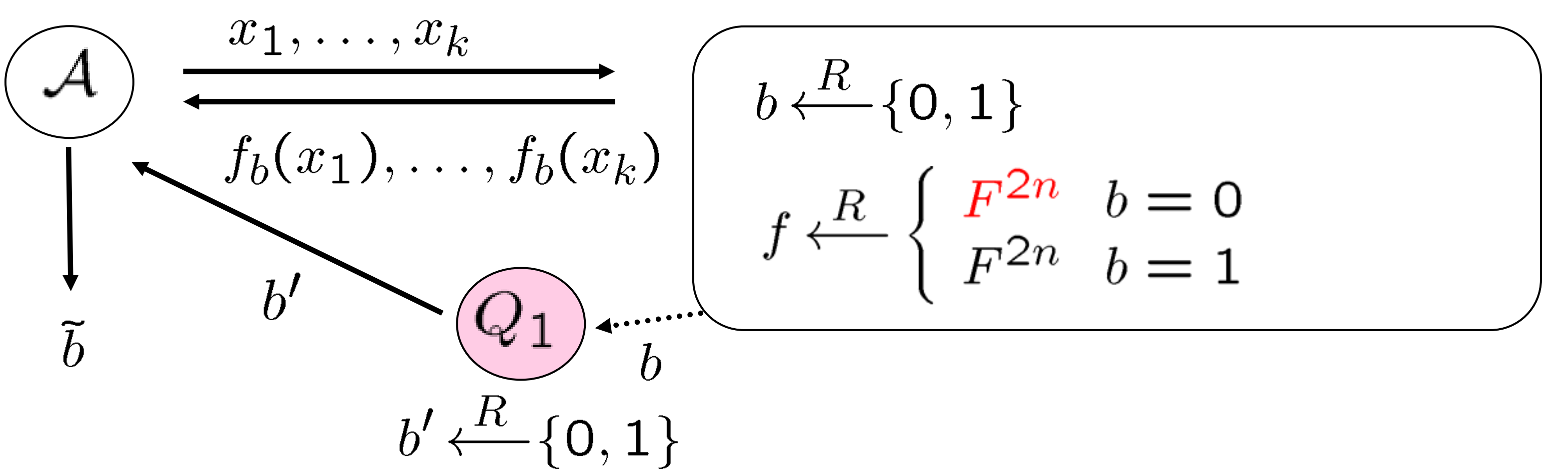




Game 2との差： ϵ_3

Game 4



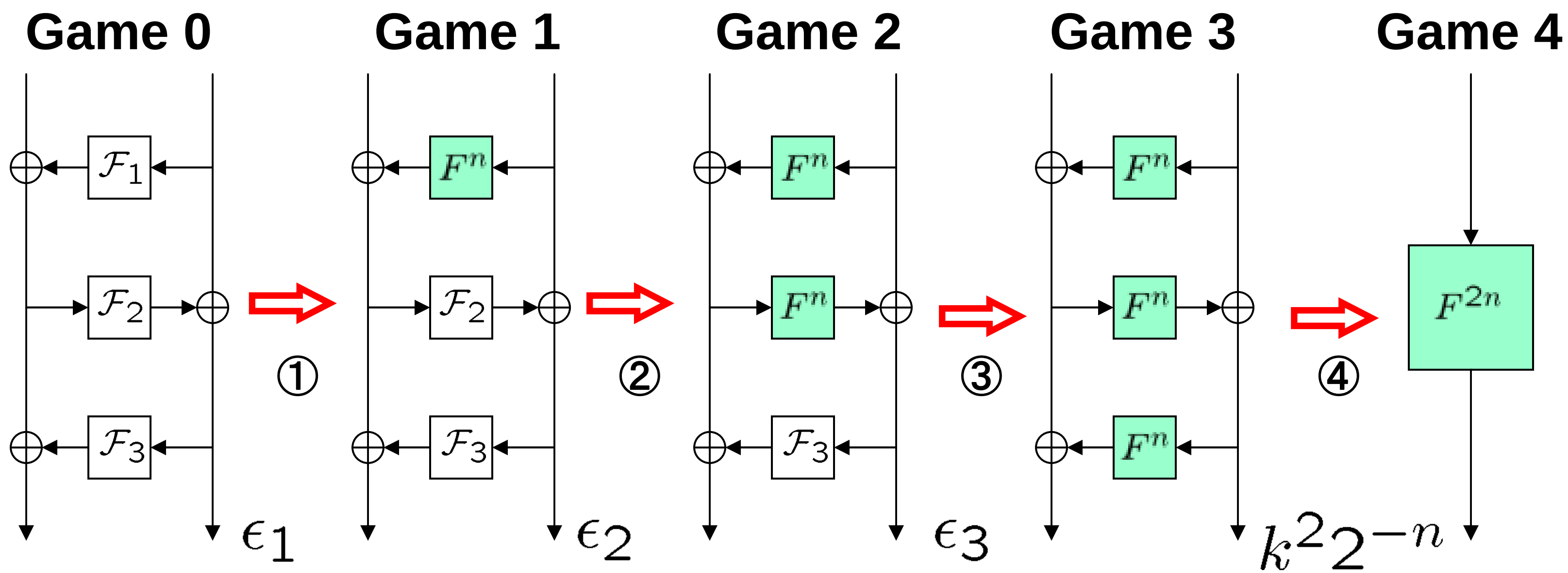


Game 3との差： $k^2 2^{-n}$

実験結果

安全性	①	②	③	④
判定結果	○	○	○	×

形式的には出来るはずだったが…



まとめ

- **Luby-Rackoff暗号のBlanchet法への適用可能性を検討**
  - 方法
    - 関数族の入出力分布の判別法を定式化
      - 観測等価性
      - One-session Secrecy
  - 結果
    - 共通鍵ブロック暗号系に対する安全性評価にもBlanchetのアプローチが有効
    - CryptoVerif による動作検証はまだ出来ていない
  - 今後の課題
    - 強擬似ランダム性（DES型４段）
    - Tweakable 暗号（DES型４段以上、SPN型）
    - Blanchet 以外のアプローチ

TOSHIBA

Leading Innovation >>>