学術論文

ハッシュ関数を用いた公平な電子抽選方法

A Fair Electronic Lottery Method by Using Hash Functions

キーワード：
抽選, ハッシュ関数, 第三者信頼機関(TTP), 公開掲示板(BBS)
Keyword：
Lottery, Hash Function, Trusted Third Party (TTP), Bulletin Board System (BBS)

電気通信大学　アグス ファナル シュクリ
University of Electro-Communications　Agus Fanar Syukri

ＮＴＴ情報流通プラットフォーム研究所　森田　光
NTT Laboratories　Hikaru MORITA

電気通信大学　太田　敏澄
University of Electro-Communications　Toshizumi OHTA

東京電機大学　齊藤　泰一
Tokyo Denki University　Taiichi SAITO

要　約

当選者を選択する福引や宝くじ，当番の順番を決めるアミダくじなど，社会には公平な抽選が必要になる場面が多数存在する。ジャンケンやルーレットなど，慣習的に納得できる方法はあったが，インターネットを介して結ばれた多数の個人の中で抽選するための，公平で誰もが納得できる抽選方法は余り議論されて来なかった。この抽選会を組織する運営会社に，抽選を一任することが理想的な解と考えられるが，規模が大きくなるほど不正を行うメリットが増すため，安全性を確保するには困難である。本論文では，抽選の電子化を目的に，ハッシュ関数を用いて多数の参加者が広く納得できる公平な抽選方法を提案する。

2003年11月10日受付　2004年7月21日受理

Abstract

A fair lottery, which chooses a prize winner or determines a person on duty turn, is needed in society. Although there are usual methods, which everyone can be convinced, such as tossup and roulette, fair lottery methods for casting lots in a large number of people connected through the Internet have seldom argued. An actual solution to entrust a lottery process should be managed by a company which organizes a lottery. Since a large scale lottery increases merits, it is difficult to mount guard over lottery systems. In this paper, we proposed a fair lottery method by using hash function which can be widely convinced to many participants on electronic lotteries.

1　はじめに

当選者を選択する福引や宝くじ，当番の順番などを決めるアミダくじなど，社会には公平な抽選が必要になる場合が多数存在する。ジャンケンやルーレットなど，慣習的に納得できる方法はあったが，インターネットを介して結ばれた多数の個人の中で抽選するための，公平で誰もが納得できる抽選方法は余り議論されて来なかった。

従来の類似研究には，電子サッカーくじ（Kobayashi et.al.,2000）があるが，くじの売買方法と，当りくじの安全な授受方法を提案するものであって，くじの抽選は扱わなかった。サッカーくじに限らず，競馬競輪，ナンバーくじなど，くじ購入者が，「当り」を恣意的に決められない点は抽選と共通であったが，賭け対象自体に電子的な操作を加える必要がなかったためである。

しかし，物理的な抽選方法といえるジャンケンやルーレットには，慣習的に納得できる対面環境があったが，インターネットなど対面でない電子的な環境では，誰もが納得できる電子的な抽選手段が別に必要である。ジャンケンなどの萌芽的な研究は既になされている(太田ら，1995)が，多人数間での効率的な抽選方法の研究はあまりなされていない。

また，電子的な環境では，スケールファクターも考慮すべき重要な要素である。規模が大きくなると，抽選会を運営する会社に，抽選を一任することが理想的な解であると考えられるが，くじなどの抽選では，一般的に規模が大きいほど不正を犯すメリットが増すため，安全性が攻撃される可能性は，物理的な従来型のくじよりも大きい。

一方，電子抽選に対する実社会のニーズが存在する。例えば，株式の新規公開や増資などの公募において，市場価格を大きく変動させずに適正価格を決めたいという要望がある。従来の入札方式に代わって，Book-Building方式が導入され，株購入者は証券会社により，抽選で割当てられる。しかし，抽選は一般に非公開なので，株割当が公平に行なわれたかという信用を得るのが困難であった。

対面でない電子的な環境において，D. Eastlakeが，誰もが抽選の公平性を検証できる電子抽選方法をIETF[1](以降，「IETF抽選」と呼ぶ)へ提案した(Eastlake,2000)。しかし，リストに載るだけで，候補者にとって名誉なことで，候補者のプライバシー保護が必要とされなかったため，プライバシー保護がなさられていなかった。また，提案された抽選対象は１０～１００人程度の小規模なものを前提としていた。

本論文では，抽選の電子化を目的に，ハッシュ関数を用いて，大規模で参加者が広く納得でき，Book-Buildingの株配分の抽選のように必要とされる参加者のプライバシーを保護し，効率的で公平な抽選方法を提案する。

ハッシュ関数はドキュメントや数字などの文字列（原文）の羅列から一定長のデータに要約（生成）するための関数・手順である。ハッシュ関数は一方向関数[2]であるため，生成データから原文を推定することは不可能で，通常，署名されたメッセージダイジェストのビットコミットメントとして，使用されている。ハッシュされる原文が予め存在して，生成されたデータを用いて，その原文の認証と完全性検査を行う。しかし，本論文では，ハッシュ関数の使用方法が，ハッシュされる原文の一部を後で生成して，抽選ルールを後で覆せないように使用する。

電子的でない環境でも，くじ購入者全員が対面しない宝くじの場合は，公開会場の抽選会を催すことで，一定の信頼感を与えてきた。ここでは，この考え方を電子的に実現することを目標にした。

著者らは，Book-Building方式の抽選に対して，先行研究（アグスら，2002,2003）を行っているが，ここでは，そのコア部分と言える抽選に対する公平性に絞って議論する。以下，２節では抽選の構成と要求条件を示し，課題に対する解決方法のアプローチを３節で説明する。また，４節では提案する抽選プロトコルを説明し，５節で考察し，

6節でまとめる。

2 抽選の構成と要求条件

2.1 抽選の構成

ここで述べる抽選をモデル化し，構成要素として，主催者と応募者の二者を考える（図1参照)。主催者は，抽選を催し応募者を募る。応募者は，複数人いて，主催者に参加申込をする。なお，いろいろな抽選会が想定されるが，通常，主催者が用意した札または券などを応募者に渡す。

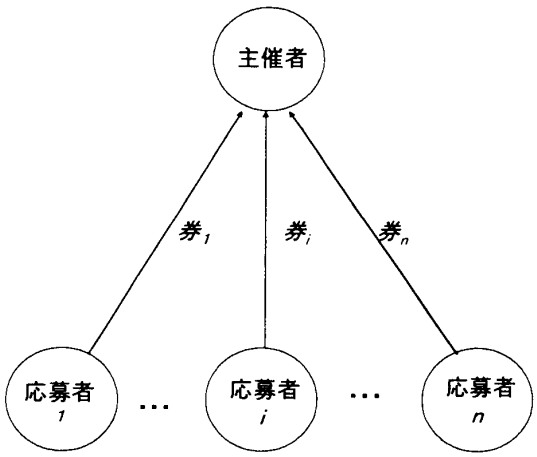

図1　抽選の構成要素

2.2 要求条件

提案プロトコルにおける抽選を実施する要求条件は以下の通りである。

応募者の匿名性: 匿名性の要件から，抽選で当ったかどうかも含めて，応募者の匿名性が守られること。

頑強性: 当選者を確定する前後に，主催者などが外れ券の可能性も含め，事前に大量の水増し券を登録したり，事後に当り券を不正に好みの応募者に配布することがない。また，主催者などにより不正に水増しされることがあっても，その不正を排除または検出できること。

公開検証可能性: 誰でも，正しく抽選されていることを納得できること。

効率性: 応募者におけるプロトコル実行にかかる計算量および通信量が少ないこと。

応募者のWalk Through性: 応募者は，投票時に一度だけ立ち会えば，公開情報だけで自分の当り外れが分かり，当りによる引換えなどが必要な場合を除き，立ち会わずに済むこと。

2.3 記号の定義

本論文では，次の記号を使う。

ID_i：　応募者iの識別子。iは$1 \leq i \leq n$である。宝くじような整理番号の変わりに，ID_iを各応募者に渡して，当選確認や抽選の公平性の確認などに使う。

R：　（誰も予測できない）乱数値。応募終了後に決定される。具体的には，公開会場におけるルーレット[3]の様に，不正や疑念が生じない環境で決定される値である。その他の乱数値としては，誰もが意図的に操作できないと思われるいくつかの試合の結果や株の最終価格を連結する値，あるいはそれらの情報を組合せる値も考えられる。

Th：　主催者が公開する当選者選定の閾値。具体的に，Book-Buildingの株分配では,応募者らをリストにして,当選と落選の2つグループに分ける時，閾値Thがそのグループ分けの境界を指す。

$x||y$：　xとyのデータの連結で，xとyは同じ型(文字列)データである。

$H(x)$：xのハッシュ値。$H(\cdot)$はハッシュ関数。

3 アプローチ

公平な抽選を行うために，後で決定する乱数を用いて，応募者の順序を付けて，初めの方から当選者を選定する。乱数さえ公平に選定されれば，その乱数を入力する関数の出力から，各応募者に対して値を生成し，その値の大小で応募者の順序

付けができる。

後の処理で，抽選の正しさを検証できる様に，BBS（公開掲示板）を導入する。これは，BBSの情報が公開され，誰もが見ることができ，そのBBSに公開された情報に基づいて抽選の公平性を検証するためである。この結果，主催者などにより不正があっても，その不正を検出できるようになる。

また，各応募者のプライバシー（個人名など）を秘匿する一方，不正追及の手段も残す防止策として，TTP（第三者信頼機関）を導入する。提案する抽選システムにおいて，TTPが主催者や応募者などに信頼され，不正しない，または他のものと結託し，不正しないとする。しかし，例えTTPが不正をしても，主催者や応募者らやBBSなどの働きにより，その不正を発覚する手段もある。
従って，全応募者が，BBSに各自分のID_iを登録することにより，事後の登録を抑止する。これは事後に，不正の水増し抽選券が出回ることを防止する意味から設定される。つまり，当選者が確定した後，主催者が当り券を不正に好みの応募者に配布することができない（**事後の不正水増し防止機能**）。

しかし，応募者などが外れ券の可能性も含め，事前に大量の水増し券を登録し，当り券を入手する不正も考えられる。これに対して，一人一登録しかできないような前提にするために，TTPを導入する。

例えば，TTPは抽選券を購入した者を正規な応募者と見なす。また，購入でなく，個人情報の本人確認をした者を正規な応募者とする場合など，実際の抽選の環境によって，TTPが確認する情報は異なる。不正な応募があったかどうかは，BBS上の情報をTTPが監視することで検出できる（**事前の不正水増し防止機能**）。

匿名性確保の観点からも，TTPを活用できる。もともと匿名性がある貨幣による購入の場合は不要であるが，個人情報による本人確認の場合，個人情報はTTPだけが持ち，他へ漏らさないことにする（応募者の匿名性）。つまり，通常は，各応募者は*ID*だけで区別される存在で，匿名性が確保される。一方，不正が発覚した場合，*ID*の持ち主である本人が特定され，個人情報が開示される。

4 提案プロトコル

提案する抽選プロトコルの詳細を以下に説明する(図２参照)。

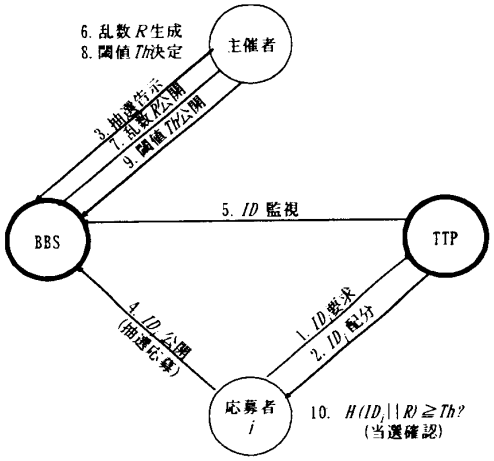

図２　提案プロトコル

4.1 構成

提案する抽選プロトコルの構成は図－２に示す。主催者は一人存在する。応募者は複数人いるが，ここでは応募者iに注目して，議論を進める。同様に，BBSとTTPが複数存在させることも可能だが，ここでは，説明の簡単のため,BBSとTTPは各々一つにする。なお，図－２の中に書いている数字はStep番号に対応する。

4.2 応募処理

Step-1. ***ID*要求：** 応募者iがTTPにID_iを要求する。TTPは，応募者iを本人確認する。本人確認の手段は，ディジタル署名やバイオメトリックス技術などが考えられる。

Step-2. ***ID*配分：** TTPが応募者iを本人確認できれ

ば，応募者iにID_iを与える。応募者iとそのID_iの対応を記録する。

Step-3. 抽選告示： 主催者はBBSに抽選の公募を告示する。公募では選定するルールも公開する。具体的な抽選ルールとして，例えば，「応募者の中で順序付けて，その上位何人かを当選者とする」などの当選条件の情報を事前に公開しておく。その情報とは，ハッシュ関数の種類，乱数の決定方法，順序付け方法などである。

Step-4. 応募： 応募者iは，ID_iをBBSに，匿名通信路[4]を介して，公開することで，主催者へ抽選を応募する。

Step-5. *ID*監視： TTPは，登録された*ID*が管理されているものであることを，BBS上の情報を監視することで確認する（**架空名義による事前水増し登録の防止**）。

4.3 抽選処理

公平な抽選処理を行うために，主催者が応募処理後で乱数を決定する。決定した乱数を，ハッシュ関数の入力の一部分として用いて，応募者の順序を付けて，当選者を選定する。

当選者の選定方法は二つ種類がある。まず，応募者を順序付けて，当選者を選定する方法である。次に，応募者の中から，必要な数だけ(グループ)当選者を選定する方法である。これらの方法を次に説明する。

●順序付けの場合

ごく一部の人を当選者とするための順序付けの方法を次に述べる。順序付けた中の小さい番号を当選者とする。

Step-6. 乱数生成： 主催者は，公平に乱数Rを生成する。

Step-7. 乱数公開： 主催者は，決定した乱数RをBBS上に公開する。

●抽選によるグループ分けの場合

以下に，応募者の中から，必要な数だけ当選者を選び出す方法について述べる。基本的に，順序付けの場合と同じであるが，順序付けた後で，上位の何個を選定するなどと決める。選定される数に応じ，リストの中に，ある境界となる値（以降「閾値Th」と呼ぶ）以上のものを当選者とするなどと決める。

Step-6. 乱数生成： 順序付けの場合のStep-6と同様の処理である。

Step-7. 乱数公開： 順序付けの場合のStep-7と同様の処理である。

Step-8. 当選者選定閾値決定： 主催者が，応募者の中から当選者のグループを選定するために，閾値Thを決定する。この閾値Thの決定方法は以下に示す。

Step-8-1. 決定された乱数Rとハッシュ関数Hを用い，組$(ID_i, H(ID_i \| R))$を生成する。

Step-8-2. Step-8-1で生成された，全てi $(1 \leq i \leq n)$の組$(ID_i, H(ID_i \| R))$を第二要素のハッシュ値に関して昇順(公募告示で“昇順”と示されたと仮定する)に並べたリストを作る。

Step-8-3. リストの上位から必要な分だけ組を取り，その中の組$(ID_i, H(ID_i \| R))$のID_iが当選された者の*ID*であるとし，最も小さなハッシュ値$H(ID_i \| R)$を選定閾値Thと設定する。

Step-9. 当選者選定閾値公開： 主催者が決定した閾値ThをBBSに公開する。なお，閾値Thは，各応募者などが確認し易いように与えるものであり，BBS上で，全体の順番のリストを示すことでも，選定の正しさを検証することができる。

4.4 当選者確認処理

●順序付けの場合

Step-10. 当選確認・公平性確認： 応募者iは，BBSに公開された乱数Rを入手し，$H(ID_i \| R)$を計算し，発表にされている当選番号ないし当選条件と照合する。また，応募者iは自分以外の応募者に対して，BBSに公開された

$ID_{j\,(i\neq j)}$とRを入手し，$H(ID_j \| R)$を計算することで，主催者が選定した応募者の数と当選者を確認できる。

●抽選によるグループ分けの場合

Step-10. 当選確認・公平性確認： 応募者iは，BBSに公開された乱数Rと閾値Thを入手し，$H(ID_i \| R)$を計算し，Th以上か以下で，主催者に選定されるか否かを確認する。また，応募者iはBBSに公開された$ID_{j\,(i\neq j)}$，RとThを入手し，$H(ID_j \| R)$を計算し，$H(ID_j \| R) \geqq Th$と照合することで，主催者が選定した応募者の数と当選者を確認できる。

5 考察

5.1 安全な応募処理方法

提案プロトコルにおいて，IDが単なる識別数字などのパラメータの場合，第三者にID_iが推測される可能性があり，他人へ当選券が渡されるなどの危険が生じる。

従って，より安全な方式として，公開鍵暗号方式を用いて，応募処理を行うバリエーションも考えられる。その実現には，Step-1で，各応募者が自分のディジタル署名用の（応募者による公開の）検証鍵を応募者からTTPに渡し，Step-2で，IDの代わりに，その検証鍵のTTPによって署名された公開鍵証明書を各応募者に渡すという方法である。

その時，Step-5では，TTPの公開鍵もBBSに公開されるとし，その公開鍵とTTPによる証明書により，応募者の正当性を誰もが確認できるようにし，本人確認が必要なときは，その公開鍵に対応する署名を生成できる人が正規の応募者であるとする。

5.2 当選者確認方法

応募者が主催者にIDだけを示して確認される場合，当選者に他人がなりすましできる。提案されたプロトコルの当選者確認処理では，本人確認のために，TTPを介在させた方が良いが，応募処理時にTTPへ何を預けるかにより，当選者確認処理が異なる。主な方法とその違いを以下に示す。

TTPからレシートを貰っている場合: 抽選券購入時のレシートを示して，本人であることを証明する。但し，レシートの照合用の情報は公開しない前提とする。

個人情報を渡している場合: 本人を確認できる身分証明書，パスワード，暗証番号，バイオメトリックスなどで本人確認させる。

公開鍵PK_iを渡している場合: 当選金を支払う者が，適当な乱数を当選者に渡し，その乱数に対する署名を，当選者が生成する。当選金を支払う者は，登録されている公開鍵PK_iで検証して確認する。

検証鍵S-Keyを渡している場合: 応募者がBBSに登録するIDとしてハッシュ関数$H(x)$[5]を選び，この所有者を確認際に，当選金を支払う者に，xを示す。

参加資格として，匿名性のある金を渡しただけで，応募者の資格が与えられている場合，レシート（当選者確認方法(1)）か公開鍵署名を使う方式（当選者確認方法(3)）しか採用できない。
また，このときのTTPの役割は応募者の登録業務だけであり，宝くじの販売所の役割に止まる。このような場合，TTPの安全性に対する積極的な存在理由は無いので，BBSや主催者と一体化することが可能である。

当選者確認方法(3)，公開鍵PK_iを渡す前提の場合，当選者確認の処理でTTPが存在しなくても良いので，他の当選者確認方法[6]に無い長所がある。しかし，公開鍵暗号方式による署名処理は，通常，共通鍵暗号方式やハッシュ関数の処理より低速であるという短所が知られている。

この対策として，S-Keyの応用（当選者確認方法(4)）で，ハッシュ連鎖などの技法を使う方法が提案されている(Kobayashi et.al.,2000)。つまり，BBSなどに登録するIDとして$H(x)$を選び，xを示すことで，本人確認を行う。

ハッシュ関数の処理はディジタル署名に比べ，

通常速いので負荷が低減される。また，TTPも応募処理の公開鍵証明書としての生成が不要になるので，負荷が軽減される。さらに，ハッシュ関数の使用は，署名方式の安全性を弱めるものではない(Stinson,1996)。

5.3 順序付けの関数

順序付け関数として，ハッシュ関数以外の関数でも，出力が一様分布である関数であれば良い。しかし，その仕組みが簡単でなく，思わぬ恣意的な分布が隠れているとも限らない。この点において，生成されるハッシュ値は制御できないことが確かめられているため，ハッシュ関数を採用する方が適当である。

5.4 要求条件の検証

2.2節の要求条件との対応関係を検証する。

応募者の匿名性: 抽選で当ったかどうかも含めて，応募者は原則的に，識別子*ID*だけで区別される。その*ID*と応募者との対応関係はTTPしか知らないので，匿名性は保証される。

頑強性: *ID*がTTPと関連付けられるため，主催者などによる架空*ID*の水増しができない。TTPがBBSに登録される*ID*を監視するので，TTPに個人情報が登録されていない*ID*をBBSに公開できない[7]ためである。

公開検証可能性: 乱数*R*が未知であることにより，応募者にも主催者にとっても，ハッシュ関数$H(\cdot \| R)$の出力が予測不可能である。リストの順序付けが予測不可能であることより，当選者の公平性が保たれる。主催者と一部の応募者に結託があったとしても，主催者はその応募者を優遇できない。また，誰もが，BBSに公開された情報（*ID*, *R*, *Th*など）に基づいて，抽選の公平性を検証できるため，公開検証可能である。

効率性: 応募者のプロトコル実行にかかる計算量は，確認時のハッシュ関数による確認一回である。TTPに公開鍵を登録するオプション処理があったとしても，登録時のディジタル署名処理一回と，当選した場合の本人確認用のディジタル署名一回程度しか負荷がかからない。

応募者のWalk Through性: 応募者は募集時に，一度だけIDを登録すれば，公開情報だけで自分の当り外れが分かり，「当り」などによる引換えが必要な場合を除き，立ち会わずに済む。

5.5 IETF抽選との比較

IETF抽選は，毎年４０～６０人の候補者の中から１０人の理事を選定するとき使われ，本提案の順序付け抽選の処理(Step-6～Step-8)を選定する数の回数で実行しなければならない。これより，本提案の抽選によるグループ分けの場合で，１回の実行で済む。

本提案した方法にはIETF抽選と同様に，ハッシュ関数を抽選ルールとして用いたが，提案方法は以下の特徴がある。

プライバシー保護： IETF抽選はプライバシー保護が必要ないが，Book-Buildingの株配分方法のように，匿名性保護が必要な抽選方法として，TTPとBBSを導入して，実現可能である。

効率性と規模： IETF抽選は，選定する数だけで実行しなければならないが，本提案の抽選によるグループ分けの場合で，規模によらずに，１回のプロトコル実行で済む。

6 まとめ

電子環境における抽選処理として，抽選の主催者が，ハッシュ関数を用いて,応募者の順序付けを行って抽選を実行する公平な抽選方法を提案した。この提案方法によれば，応募者のプライバシー(匿名性)が保護され，当選者決定の処理がガラス張りになり，効率的で公平な抽選ができ，大人数の応募者などが納得できる。

謝　辞

査読者の方々より，本論文の位置付けや記述に関して有益な指摘や助言を数多く頂き，深く感謝いたします。

また，電気通信大学大学院情報システム学研究科の菅野哲氏より，文章の修正と表現に関して貴重なコメントや指摘を頂き，ここに感謝の意を表します。

注

1)IETF (Internet Engineering Task Force)はインターネット上の情報通信に関する標準化を促進するために設立された団体である。

2)関数それ自身の計算は簡単だが，逆関数の計算が非常に難しい関数である。

3)ここでは，公開ルーレットを用いたが，乱数Rは生成方法の例の一つに過ぎない。誰もが予測できなく，不正や疑念が生じない環境で決定されれば,どんな乱数でも可能である。

4)送信者を特定できないネットワークである。例えばMix-netなどがある。

5)ハッシュ関数$H(\cdot)$を決めるのは主催者で，ハッシュされる中身xを決めるのは応募者自身である。

6)当選者確認方法(1),(2)と(4)には，TTPの存在が必要である。

7)TTPに個人情報が登録されていない*ID*をBBSに公開できたとしても，TTPがBBSに公開された*ID*を監視するので，TTPが主催者にその*ID*を参加解除させることができる。

参考文献

アグス・ファナル・シュクリ，森田 光，税所 哲郎，齊藤 泰一 (2002)「Book-Buildingにおけるハッシュ関数を用いた株式の公平な配分方法」信学技報，ISEC-2002-97, pp.7-12.

アグス・ファナル・シュクリ，森田 光(2003)「ハッシュ関数を用いた公平な抽選方法」，２００３年第九回社会情報システム学シンポジウム，pp.117-121，平成１５年１月.

太田 和夫，黒澤 馨，渡辺 治(1995)『情報セキュリティの科学－マジック・プロトコルへの招待』，ブルーバックスB-1055，講談社.

Kunio Kobayashi, Hikaru Morita, Mitsuari Hakuta, and Takanori Nakanowari (2000) ” An Electronic Soccer Lottery System that uses bit commitment”, IEICE Trans. Inf & Sys, Vol. E83-D, no.5, pp.980-987.

D. Eastlake 3rd (2000) ” Publicly Verifiable Nomcom Random Selection”, Request for Comments (RFC) 2777. (available at http://www.ietf.org/rfc/rfc2777.txt)

Douglas R.Stinson (1996), 櫻井 幸一(翻訳) 『暗号理論の基礎』，共立出版.