(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2005-292412
(P2005-292412A)
(43)公開日　平成17年10月20日(2005.10.20)

(51)Int.Cl.[7]	F I	テーマコード（参考）
G09C 1/00	G09C 1/00 650B	5J104

審査請求 未請求　請求項の数 8　ＯＬ　（全 10 頁）

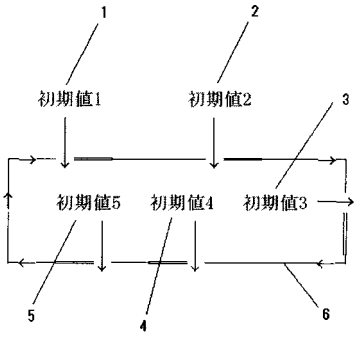
(21)出願番号　特願2004-106329 (P2004-106329)
(22)出願日　平成16年3月31日 (2004.3.31)

(71)出願人　591106864
富士通エフ・アイ・ピー株式会社
東京都江東区青海２丁目４５番
(74)代理人　100109726
弁理士　園田　吉隆
(74)代理人　100101199
弁理士　小林　義教
(72)発明者　山岸　国夫
東京都江東区青海２丁目４５番
富士通エフ・アイ・ピー株式会社内

Ｆターム(参考)　5J104 AA01 AA18 FA09 NA06 NA09 NA20

(54)【発明の名称】ストリーム暗号のための乱数かき混ぜ方法およびプログラム

(57)【要約】
【課題】
乱数発生アルゴリズムにおいて、１６～６４Ｋバイト程度のバッファ容量および演算で十分ランダムな乱数列を発生させる。
【解決手段】
ストリーム暗号における乱数かき混ぜ方法であって、（１）２次元マトリクステーブルの各行の１列目に第１の乱数列を入れる初期値生成段階と、（２）該２次元マトリクステーブルの各行に、各行の１列目を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、（３）該２次元マトリクステーブルから第１の数値を取り出す乱数取出し段階と、（４）第１の数値から次に取り出すべき第２の数値のアドレスを生成するアドレス生成段階と、（５）取り出した第１の数値が入っている場所に新たな乱数を入れる段階と、を有する乱数かき混ぜ方法およびプログラムを提供する。
【選択図】　図１



【特許請求の範囲】
【請求項１】
ストリーム暗号における乱数かき混ぜ方法であって、
（１）２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、
（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、
（３）該２次元マトリクステーブルから第１の数値を取り出す乱数取出し段階と、
（４）第１の数値から次に取り出すべき第２の数値のアドレスを生成するアドレス生成段階と、
（５）取り出した第１の数値が入っている場所に新たな乱数を入れる段階と、を有する乱数かき混ぜ方法。
【請求項２】
前記（５）の段階の新たな乱数は第１の数値が入っている行の第２の乱数列のうち、２次元マトリクスに入っていない乱数列である請求項１に記載の方法。
【請求項３】
ストリーム暗号における乱数かき混ぜ方法であって、
（１）２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、
（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、
（３）該２次元マトリクステーブルから第１の数値を取り出す乱数取出し段階と、
（４）取り出した第１の数値の入っている場所に新たな乱数を入れる段階と、
（５）該新たな乱数から次に取り出すべき第２の数値のアドレスを生成する段階と、を有する乱数かき混ぜ方法。
【請求項４】
コンピュータ上のバッファを２次元マトリクスとして使用する請求項１ないし３に記載の方法。
【請求項５】
新たな乱数およびアドレスを生成する段階は、所定の乱数列と演算により行う請求項１または２に記載の方法。
【請求項６】
ストリーム暗号における乱数かき混ぜ方法であって、
（１）２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、
（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、
（３）（１）、（２）の段階を複数の２次元マトリクスについて行う段階と、
（５）これらのマトリクスから第１の数値を取り出す段階と、
（６）取り出した第１の数値の入っている場所に新たな乱数を入れる段階と、
（７）所定の乱数列と演算により第２の数値のアドレスを生成する段階と、を有する乱数かき混ぜ法。
【請求項７】
ストリーム暗号における乱数かき混ぜプログラムであって、
（１）バッファに設定された２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、
（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、
（３）該２次元マトリクステーブルから第１の数値を取り出す乱数取出し段階と、
（４）第１の数値から次に取り出すべき第２の数値のアドレスを生成するアドレス生成

段階と、
（５）取り出した第１の数値が入っている場所に新たな乱数を入れる段階と、を有する乱数かき混ぜプログラム。
【請求項８】
ストリーム暗号における乱数かき混ぜプログラムであって、
（１）バッファ上に設定された２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、
（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、
（３）該２次元マトリクステーブルから第１の数値を取り出す乱数取出し段階と、
（４）取り出した第１の数値の入っている場所に新たな乱数を入れる段階と、
（５）該新たな乱数から次に取り出すべき第２の数値のアドレスを生成する段階と、を有する乱数かき混ぜプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は暗号のための乱数発生方法であり、特に乱数かき混ぜ法に関する。
【背景技術】
【０００２】
ストリーム暗号などでは一般に乱数を用いて暗号化が行われている。しかしながら現実の乱数は疑似乱数であり、またその発生方法のため、例えば周期性を有するなどの特定の性質を持っていることが多い。よってこのような特定の性質を有していると、その性質を手がかりに乱数自体が解読されることになり、ストリーム暗号においては暗号化した文を解読される危険性を有している。
【０００３】
特定の性質を持たない乱数の発生方法としては非線形演算やハッシュ化などを行う方法が知られている。また一連の乱数を記憶装置に格納し、自身か他の乱数によって読み出し順序を決定する方法も知られている。
【０００４】
しかしながら非線形演算やハッシュ化などを行なう方法は処理に時間がかかり、この処理が乱数の性質を変化させる可能性がある。
【特許文献１】特許第２９５０４８号
【特許文献２】特開平１１－３３８３４７号
【特許文献３】共立出版 アルゴリズム辞典 Ｐ１０３～１０４ かき混ぜ法
【発明の開示】
【発明が解決しようとする課題】
【０００５】
したがって解読されにくく、また比較的少ない演算処理で乱数を発生させる方法が望まれていた。
【課題を解決するための手段】
【０００６】
上述の問題に鑑み本発明は、ストリーム暗号における乱数かき混ぜ方法であって、（１）２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、（３）該２次元マトリクステーブルから第１の数値を取り出す乱数取出し段階と、（４）第１の数値から次に取り出すべき第２の数値のアドレスを生成するアドレス生成段階と、（５）取り出した第１の数値が入っている場所に新たな乱数を入れる段階と、を有する乱数かき混ぜ方法を提供する。これらの段階を経ることによって元の乱数の性質を保ち十分ランダムな乱数列を生成することが出来る。また（５）の段階において新たに入れる乱数は、２次元マトリクスに入っていない乱

数である。これによって、解読の危険性を大幅に減少させることができる。また、連続する乱数をバッファに入れて利用することは、乱数の狭域的な利用法であり、初期値を複数生成する方法は、乱数の周期上に大域的に分散させることである。このことから、十分なランダム性を有することが出来る。図１に示したように乱数の周期６上に初期値１、初期値２・・・初期値５を分散して配置しバッファを作成する。

【０００７】

またストリーム暗号における乱数かき混ぜ方法であって、（１）２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、（３）該２次元マトリクステーブルから第１の数値を取り出す乱数取出し段階と、（４）取り出した第１の数値の入っている場所に新たな乱数を入れる段階と、（５）該新たな乱数から次に取り出すべき第２の数値のアドレスを生成する段階と、を有する乱数かき混ぜ方法を提供する。つまり、２次元マトリクスから数値を取り出した後に、新たな乱数を入れ、この乱数から次に取り出す数値のアドレスを得るものである。これによって、前記の取り出した数値からアドレスを生成する方法と比べて、アドレスが判り難くなり、解読困難性が増すという利点がある。

【０００８】

さらに別の方法として、ストリーム暗号における乱数かき混ぜ方法であって、（１）２次元マトリクステーブルの各行の最初の列に第１の乱数列を入れる初期値生成段階と、（２）該２次元マトリクステーブルの各行に、各行の最初の列を初期値とする第２の乱数列を入れる２次元乱数マトリクス設定段階と、（３）（１）、（２）の段階を複数の２次元マトリクスについて行う段階と、（５）これらのマトリクスから第１の数値を取り出す段階と、（６）取り出した第１の数値の入っている場所に新たな乱数を入れる段階と、（７）所定の乱数列と演算により第２の数値のアドレスを生成する段階と、を有する乱数かき混ぜ法を提供する。これは複数の２次元マトリクスは３次元マトリクスと同じであり、３次元のマトリクスを使用することによりランダムネスの高い乱数列を生成することが出来る。

【０００９】

このような方法はコンピュータプログラムとして実現することにより、例えばストリーム暗号で使用する乱数として解読されにくい乱数列を生成することが出来る。

【発明を実施するための最良の形態】

【００１０】

以下に本発明の実施例を説明する。まず本実施例では２次元マトリクスとして３行４列を用いる。また行列のインデックスは０から始めるものとする。つまり各セルのインデックスは表１のようになり、最初の行、最初の列はそれぞれインデックスは０から始まる。また各行には初期値に続く単調数列を用いる。また取り出した数値から次の位置を決定するには、以下の演算を行う。

十の位を列：ｍｏｄ（行，３）＝０～２・・・（１）

一の位を行：ｍｏｄ（列，４）＝０～３・・・（２）

【表１】

0,0	0,1	0,2	0,3
1,0	1,1	1,2	1,3
2,0	2,1	2,2	2,3

【００１１】

まず初期値として数列１１、８３、６５を各行の最初の列に入れ、行方向には各数値を

初期値として単調数列を入れる。よって最初のマトリクスは表２のようになる。
【表２】

11	12	13	14
83	84	85	86
65	66	67	68

【００１２】
この状態で最初に（０，０）＝１１を取り出す。１１を取り出したため、（０，０）は行方向の数列において次の数値、つまり１５が入る。よって表３のようになる。このとき入れた数値１５を用いて次に取り出す位置を計算する。（１）の式により、ｍｏｄ（１，３）＝１、（２）の式によりｍｏｄ（５，４）＝１となり、（１，１）が次に取り出すべき場所である。
【表３】

15	12	13	14
83	84	85	86
65	66	67	68

【００１３】
従って次に（１，１）＝８４を取り出し、（１，１）には同じ行の次の数値、すなわち８７を入れて表４のようになる。同様にして、新たに入れた数値８７から次の取り出し位置を入れる。（１）式よりｍｏｄ（８，３）＝２、（２）式よりｍｏｄ（７，４）＝３よって（２，３）が次の取り出し位置である。ここまでで取り出された数列は、１１、８４である。
【表４】

15	12	13	14
83	87	85	86
65	66	67	68

【００１４】
（２，３）に入っている数値６８を取り出し、ここに新たに同じ行の次の数値、すなわち６９を入れる。同様にｍｏｄ（６，３）＝０、ｍｏｄ（９，４）＝１、（０，１）＝１２が次の数値である。取り出された数列は、１１、８４、６８となる。

【表５】

15	12	13	14
83	87	85	86
65	66	67	69

【００１５】
（０，１）＝１２を取り出し、ここに同じ行の次の数値を入れる。このとき１５はすでに１回目の取り出しで使用しているので、次の１６を入れ表６のようになる。次に取り出す位置は、ｍｏｄ（１，３）＝１、ｍｏｄ（６，４）＝２、であり数値は（１，２）＝８５である。

【表６】

15	16	13	14
83	87	85	86
65	66	67	69

【００１６】
８５を取り出した後、この場所に８８を入れる。次に取り出す位置はｍｏｄ（８，３）＝２、ｍｏｄ（８，４）＝０、であり数値は（２，０）＝６５である。

【表７】

15	16	13	14
83	87	88	86
65	66	67	69

【００１７】
６５を取り出した後、次に入れる数値は７０である。次の取り出し位置は、ｍｏｄ（７，３）＝１、ｍｏｄ（０，４）＝０、であり数値は（１，０）＝８３である。

【表８】

15	16	13	14
83	87	88	86
70	66	67	69

【００１８】
８３を取り出した後次に入れる数値は８９であり、取り出し位置は、ｍｏｄ（８，３）＝２、ｍｏｄ（９，４）＝１となり、数値は（２，１）＝６６である。
【表９】

15	16	13	14
89	87	88	86
70	66	67	69

【００１９】
６６を取り出した後に７１を入れ、次の取り出し位置、および数値は、ｍｏｄ（７，３）＝１、ｍｏｄ（１，４）＝１、（１，１）＝８７である。ここまでで取り出された数列は、１１、８４、６８、１２、８５、６５、８３、６６となる。
【表１０】

15	16	13	14
89	87	88	86
70	71	67	69

【００２０】
８７を取り出した後に９０を入れ、次に取り出す位置、数値は、ｍｏｄ（９，３）＝０，ｍｏｄ（０，４）＝０、（０，０）＝１５である。
【表１１】

15	16	13	14
89	90	88	86
70	71	67	69

【００２１】
１５を取り出した後に１７を入れ、次に取り出す位置、数値は、ｍｏｄ（１，３）＝１、ｍｏｄ（７，４）＝３、（１，３）＝８６である。
【表１２】

17	16	13	14
89	90	88	86
70	71	67	69

【００２２】
８６を取り出し、９１を入れ、次に取り出す位置、数値は、ｍｏｄ（９，３）＝０、ｍｏｄ（１，４）＝１、（０，１）＝１６である。
【表１３】

17	16	13	14
89	90	88	91
70	71	67	69

【００２３】
１６を取り出し、１８を入れ、次に取り出す位置、数値は、ｍｏｄ（１，３）＝１、ｍｏｄ（８，４）＝０、（１，０）＝８９である。
【表１４】

17	18	13	14
89	90	88	91
70	71	67	69

【００２４】
このようにして取り出された数列は、１１、８４、６８、１２、８５、６５、８３、６６、８７、１５、８６、１６、８９となり各行に入れた元の数列は単調数列にもかかわらず、十分ランダムな数列となる。本実施例では取り出しがランダムに行われることを示すために各行に入れる数値は単調数列にしたが、これを乱数列にしても十分ランダムな数列を作ることが出来る。乱数を用いる場合は最初の段階でマトリクスの最初の列には第１の乱数列を入れ、この各乱数を初期値として行方向に第２の乱数列を設定することが好ましい。
【００２５】
以上のように　本実施例による取り出し順序を表１５に示した。表の各要素の数字は取り出した順序を示している。各行方向の数値並び方はランダムであり、不連続である。よ

って取り出し方が十分ランダムであることがわかる。

【表１５】

1, 10	4, 12		
7, 13	2, 9	5	11
6	8		3

【００２６】

また、上記実施例では新たに入れた数値で次の取り出し位置を求めたが、取り出した数値を用いて次の取り出し位置を計算しても同様な乱数列を生成することが出来る。本発明の方法はバッファサイズが例えば６４Ｋ、１６Ｋバイトのように小さなサイズで乱数を発生させることが出来、さらに非線形演算やハッシュ化などの処理が不要であるため高速に乱数を発生させることが出来ることが特徴である。またこのような２次元マトリクスを複数個用いて、３次元マトリクスとして同様な方法で乱数を発生させることが出来る。この場合でも各要素に対応できるように取り出し位置を計算できる。また各行に入れる乱数列の生成アルゴリズムがわかっていても、数値の取り出しは各行からランダムに行われるため最終的に生成された乱数は解読困難である。

【００２７】

ここで本発明の技術的利点を説明すると、例えば岩手大学、小島宏美著、「計算機による乱数の発生とその応用」によれば、従来の乱数かき混ぜ法では、１、２次元マトリクスでは乱数の一様性は保たれるが、アルゴリズムによっては２次元、３次元マトリクスで乱数の一様性が無くなることが指摘されている。しかしながら本発明の乱数かき混ぜ法をストリーム暗号に応用する場合、１次元のバッファサイズ、２次元のバッファサイズを十分大きく取れば一様性は上がり、乱数列のなかで次の乱数を予測するのが困難になる。さらに別系列の乱数アルゴリズムを３次元として設定することにより、暗号利用における予測困難性、解読困難性が強化される。

【００２８】

ここで本発明の乱数列を入れるバッファサイズについて説明する。本発明において１次元のバッファサイズは大きい方がよい。２次元マトリクスは初期値設定は大きい方がよく、３次元の乱数アルゴリズムは、Ｍ系列の次数の異なる原始多項式を採用すればよい。例えば４バイトの乱数を得るとすると、バッファサイズは大きい程よいが、６４～１０２４Ｋバイトでも十分効果を期待できる。１次元を６４バイト、２次元を６４バイト、３次元を４バイトとすると、４＊６４＊６４＊４となり、６４Ｋバイトとなる。また、３次元的に４バイトの乱数を生成するには３２次以上のＭ系列原始多項式を採用すればよい。これは例えば、

$f(x)=X^{p}+X^{q}+1$，（ｐ、ｑ）＝（３３，１３）（３５，２）（４１，３）（４７，５）などである。

【図面の簡単な説明】

【００２９】

【図１】図１は本発明の乱数かき混ぜ法およびプログラムの、バッファを作成する図である。。

【符号の説明】

【００３０】

１～５　２次元マトリクスの初期値

６　周期的乱数列

【図１】