(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2020-52215
(P2020-52215A)
(43)公開日　令和2年4月2日(2020.4.2)

(51)Int.Cl.		F I			テーマコード（参考）
G09C 1/00 (2006.01)		G09C	1/00	620B	5J104

審査請求 未請求　請求項の数 17　ＯＬ　（全 26 頁）

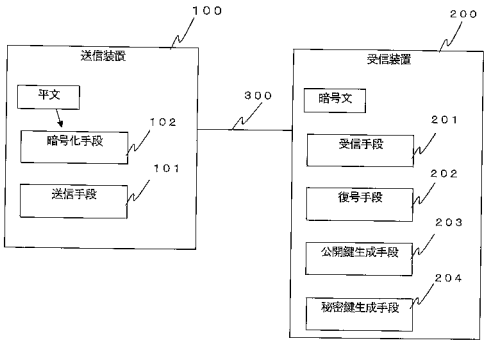
(21)出願番号　特願2018-180880 (P2018-180880)
(22)出願日　平成30年9月26日 (2018.9.26)

(71)出願人　391016358
東芝情報システム株式会社
神奈川県川崎市川崎区日進町１番地５３
(74)代理人　100090169
弁理士　松浦　孝
(74)代理人　100074147
弁理士　本田　崇
(74)代理人　100124497
弁理士　小倉　洋樹
(72)発明者　岩野　隆
神奈川県川崎市川崎区日進町１番地５３
東芝情報システム株式会社内

Ｆターム(参考)　5J104 JA28

(54)【発明の名称】公開鍵暗号システム、公開鍵暗号方法、公開鍵暗号プログラム

(57)【要約】
【課題】演算コストの低減を図ることが可能な公開鍵暗号システムを提供する。
【解決手段】公開鍵を用いて平文を暗号化し暗号文を送る送信装置１００と、送信された暗号文を受信し、前記公開鍵に対応する秘密鍵を用いて復号し平文へ戻す受信装置２００とを備える公開鍵暗号システムである。前記送信装置１００は、前記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換する暗号化手段１０２を備えており、前記受信装置２００は、前記秘密鍵を用いてベルヌーイシフト写像を実行して暗号文を平文へ変換する復号手段２０２を備えている。
【選択図】図１



【特許請求の範囲】
【請求項１】
公開鍵を用いて平文を暗号化し暗号文を送る送信装置と、送信された暗号文を受信し、前記公開鍵に対応する秘密鍵を用いて復号し平文へ戻す受信装置とを備える公開鍵暗号システムにおいて、
前記送信装置は、前記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換する暗号化手段を備えており、
前記受信装置は、前記秘密鍵を用いてベルヌーイシフト写像を実行して暗号文を平文へ変換する復号手段を備えていることを特徴とする公開鍵暗号システム。
【請求項２】
前記送信装置が、ベルヌーイシフト写像の初期値を暗号化の度に変更する送信側初期値変更制御手段を備えており、
前記受信装置が、暗号文を復号する度にベルヌーイシフト写像の初期値を、前記送信側初期値変更制御手段によって変更された初期値と同じ値に変更する受信側初期値変更制御手段と、前記受信側初期値変更制御手段により変更された初期値を用いて前記暗号文の逆元を算出する逆元算出手段とが備えられていることを特徴とする請求項１に記載の公開鍵暗号システム。
【請求項３】
前記初期値となる乱数種を生成する乱数種生成手段を有し、
前記暗号化手段は、前記乱数種生成手段により生成された乱数種と前記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換し、
前記受信装置には、前記暗号文を前記乱数種生成手段により生成された乱数種を用いて前記暗号文の逆元を算出する逆元算出手段が備えられ、
前記復号手段は、前記逆元算出手段により算出された逆元に対し、ベルヌーイシフト写像を実行して暗号文を平文へ変換することを特徴とする請求項２に記載の公開鍵暗号システム。
【請求項４】
前記乱数種生成手段は、暗号文の生成毎に新たな乱数種を生成し、前記送信側初期値変更制御手段及び前記受信側初期値変更制御手段へ与えることを特徴とする請求項３に記載の公開鍵暗号システム。
【請求項５】
前記乱数種生成手段は前記送信装置に設けられ、この送信装置では生成した乱数種を前記公開鍵を用いて前記暗号化手段においてベルヌーイシフト写像を実行して暗号化し、暗号化した乱数種を前記受信装置へ送り、
前記受信装置では、受信した暗号化された乱数種を前記復号手段で復号して用いることを特徴とする請求項４に記載の公開鍵暗号システム。
【請求項６】
前記乱数種生成手段は、暗号文の生成に際して乱数種の元となる元乱数種を１つ生成し、この元乱数種に基づき新たな乱数種を暗号文の生成毎に生成して前記送信側初期値変更制御手段へ与える一方、前記受信装置へは前記元乱数種を与える処理を行い、
前記受信装置には、暗号文の復号毎に元乱数種に基づき前記乱数種生成手段によって生成される前記新たな乱数種と同期した乱数種を生成する受信側乱数種生成手段が備えられ、この受信側乱数種生成手段により生成された乱数種を前記逆元算出手段へ与えて逆元算出を行うことを特徴とする請求項３に記載の公開鍵暗号システム。
【請求項７】
前記公開鍵を生成する公開鍵生成手段が、前記受信装置に設けられていることを特徴とする請求項１乃至６のいずれか１項に記載の公開鍵暗号システム。
【請求項８】
前記秘密鍵を生成する秘密鍵生成手段が、前記受信装置に設けられていることを特徴とする請求項１乃至７のいずれか１項に記載の公開鍵暗号システム。

【請求項９】
公開鍵を用いて平文を暗号化し暗号文を送る送信装置と、送信された暗号文を受信し、前記公開鍵に対応する秘密鍵を用いて復号し平文へ戻す受信装置とを備える公開鍵暗号システムにより実行される公開鍵暗号方法において、
前記送信装置では、前記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換する暗号化ステップを備えており、
前記受信装置では、前記秘密鍵を用いてベルヌーイシフト写像を実行して暗号文を平文へ変換する復号ステップを備えていることを特徴とする公開鍵暗号方法。
【請求項１０】
前記送信装置には、ベルヌーイシフト写像の初期値を暗号化の度に変更する送信側初期値変更制御ステップが備えられており、
前記受信装置には、暗号文を復号する度にベルヌーイシフト写像の初期値を、前記送信側初期値変更制御手段によって変更された初期値と同じ値に変更する受信側初期値変更制御ステップと、前記受信側初期値変更制御ステップにより変更された初期値を用いて前記暗号文の逆元を算出する逆元算出ステップとが備えられていることを特徴とする請求項９に記載の公開鍵暗号方法。
【請求項１１】
前記初期値となる乱数種を生成する乱数種生成ステップを有し、
前記暗号化ステップでは、前記乱数種生成ステップにより生成された乱数種と前記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換し、
前記受信装置には、前記暗号文を前記乱数種生成ステップにより生成された乱数種を用いて前記暗号文の逆元を算出する逆元算出ステップが備えられ、
前記復号ステップは、前記逆元算出ステップにより算出された逆元に対し、ベルヌーイシフト写像を実行して暗号文を平文へ変換することを特徴とする請求項１０に記載の公開鍵暗号方法。
【請求項１２】
前記乱数種生成ステップでは、暗号文の生成毎に新たな乱数種を生成し、前記送信側初期値変更制御ステップ及び前記受信側初期値変更制御ステップへ与えることを特徴とする請求項１１に記載の公開鍵暗号方法。
【請求項１３】
前記乱数種生成ステップは前記送信装置に設けられ、この送信装置では生成した乱数種を前記公開鍵を用いて前記暗号化手段においてベルヌーイシフト写像を実行して暗号化し、暗号化した乱数種を前記受信装置へ送るステップを有し、
前記受信装置では、受信した暗号化された乱数種を前記復号ステップで復号して用いることを特徴とする請求項１２に記載の公開鍵暗号方法。
【請求項１４】
前記乱数種生成ステップは、暗号文の生成に際して乱数種の元となる元乱数種を１つ生成し、この元乱数種に基づき新たな乱数種を暗号文の生成毎に生成して前記送信側初期値変更制御ステップへ与える一方、前記受信装置へは前記元乱数種を与える処理を行い、
前記受信装置には、暗号文の復号毎に元乱数種に基づき前記乱数種生成ステップによって生成される前記新たな乱数種と同期した乱数種を生成する受信側乱数種生成ステップが備えられ、この受信側乱数種生成ステップにより生成された乱数種を前記逆元算出ステップへ与えて逆元算出を行うことを特徴とする請求項１１に記載の公開鍵暗号方法。
【請求項１５】
前記公開鍵を生成する公開鍵生成ステップが、前記受信装置に設けられていることを特徴とする請求項９乃至１４のいずれか１項に記載の公開鍵暗号方法。
【請求項１６】
前記秘密鍵を生成する秘密鍵生成ステップが、前記受信装置に設けられていることを特徴とする請求項９乃至１５のいずれか１項に記載の公開鍵暗号方法。
【請求項１７】

公開鍵暗号システムのコンピュータを請求項１乃至８の各手段として機能させることを特徴とする公開鍵暗号プログラム。

【発明の詳細な説明】

【技術分野】

【０００１】

この発明は、公開鍵暗号システム、公開鍵暗号方法、公開鍵暗号プログラムに関するものである。

【背景技術】

【０００２】

従来、公開鍵暗号方式としては、ＲＳＡ暗号が知られている。このＲＳＡ暗号の基本は、冪乗を行って、法（Ｍｏｄｕｌｏ）を取る演算であり、余りを掛け算していくことで演算精度幅を大きくすることなく剰余を行うなど演算コストを削減する手法や冪乗法が用いられることが知られている。即ち、この特許文献１の発明では、ＣＰＵの空き時間をＲＳＡ暗号では、法を取る計算において剰余算が必須であり、割り算を行うために演算コストがかかってしまう。

【０００３】

上記のＲＳＡ（Ｒｉｖｅｓｔ－Ｓｈａｍｉｒ－Ａｄｌｅｍａｎ）暗号処理を採用したものは、特許文献１に紹介されている。この特許文献１には、ＲＳＡ暗号では、演算負荷が大きく、処理能力の低い組み込み機器のＣＰＵでは、ＣＰＵを占有する時間が長くなるといった問題が指摘され、これを解決するために、ＣＰＵの空き時間をできる限り少なくなるように、非均等に分割し、短時間でＲＳＡ暗号処理を完了させることが開示されている。

【０００４】

また、ＲＳＡ暗号の弱点としては、決まりきった平文として“Ｙｅｓ”か“Ｎｏ”のみにより構成される場合、同じ平文を同じ公開鍵で暗号化するために行い続けると暗号文が限られるため平文が見破られるといった問題がある。これを解決するために平文に乱数を付加して毎回の暗号文を見かけ上変更して送信する手法が取り入られている。しかしながら、この手法では、受信者が復号してから乱数部分を取り除くといった手間が必要であること、また乱数の付加によって平文のサイズが増加するという問題がある。そして、このような乱数を付加する形態では、仮に公開鍵暗号の秘密鍵が第三者に知られてしまった場合には、復号後に乱数部分を取り除くことで平文が特定されてしまう欠点がある。

【０００５】

また、特許文献２には、共有鍵を安全に共有することが可能な暗号化方式が開示されている。この特許文献２においては、一時鍵発生手段２２が乱数である一時鍵ｒｉを発生し、この一時鍵ｒｉを暗号化手段１４が第１初期鍵ｋｉ１を用いて暗号化し暗号文ｃｉ１を得て、送信手段２０が暗号文ｃｉ１を受信装置５０に送信する。上記暗号化手段１４は、上記一時鍵ｒｉを第２初期鍵ｋｉ２により暗号化し、共有鍵ｋｓを得る。一方、受信装置５０は、暗号文ｃｉ１を受信し、復号化手段５６に供給し、復号化手段５６は、暗号文ｃｉ１を復号化して、元の一時鍵ｒｉを得る。暗号化手段５４が第２初期鍵ｋｉ２を用いてこの一時鍵ｒｉを暗号化することによって、送信装置１０側と同様に受信装置５０側でも共有鍵ｋｓを得ることができる。以降、この共有鍵ｋｓを用いて秘密の通信を行うというものである。

【０００６】

特許文献３には、情報を暗号化して伝送するのに好適な暗号通信システムが開示されている。この特許文献３のものは、暗号通信システム１０１の送信装置１３１と受信装置１５１とは、それぞれ秘密鍵と公開鍵を生成するとともに、現在時刻をもとに乱数を生成し、この乱数を用いてセッション鍵を作り、ＲＳＡ暗号の技術を用いてセッション鍵を共有するものである。共有されたセッション鍵は、乱数の暗号化にも用いられ、暗号化された乱数を復号したときに元の乱数と一致することによって、セッション鍵を認証する。認証されたセッション鍵により伝送すべき情報をベクトルストリーム暗号によって暗号化し伝

送することで、通信の秘密を保つことができる。
【０００７】
特許文献４には、高速演算が可能な新規なカオス的時系列を探索し、このカオス的時系列を用いてカオス発生装置やカオス暗号装置などを実現することが開示されている。具体的には、このカオス発生装置は、変数ｎの増加に従って急激に増加する関数ｆｉ（ｎ）［ｉ＝１～Ｌ、Ｌ≧１］に対し素数ｍｉ［ｉ＝１～Ｌ、Ｌ≧１］を設定する手段、変数ｎの初期値ｎ０に対してｆｉ（ｎ０）から素数ｍｉを法として剰余ｒｉ（ｎ０）［ｉ＝１～Ｌ、Ｌ≧１］を導出する初期値演算手段、ｆｉ（ｎ＋１）の計算では剰余ｒｉ（ｎ）を利用して素数ｍｉを法として導出された剰余ｒｉ（ｎ＋１）を導出する反復演算手段、変数ｎをｎ０から順次増大させながら前記剰余ｒｉ（ｎ）［ｉ＝１～Ｌ、Ｌ≧１］を通して生成される一意の値を有するカオス的時系列Ｘｎを出力するカオス信号出力手段により構成される。
【０００８】
更に、特許文献５には、Ｒ／Ｗから乱数生成の初期値を暗号フレームに包含／送付し、乱数種を暗号フレーム毎に変化させることで、ＲＦＩＤの利便性を確保しつつ、暗号解読を困難にする暗号化方法が開示されている。この特許文献５の発明では、リーダ／ライタが、下り暗号フレームの送信の前に、前回送信した下り暗号フレームに包含した付与データを格納する第１の付与データ格納手段の格納値を元に乱数種を生成し、乱数種を初期値として第１の乱数生成手段にて生成する乱数データを元に暗号化／復号化を行うものである。一方、タグでは、下り暗号フレームの受信の前に、前回受信した下り暗号フレームに包含する付与データを格納する第２の付与データ格納手段の格納値を元に乱数種を生成し、乱数種を初期値として第２の乱数生成手段にて生成する乱数データを元に復号化／暗号化を行うものである。
【先行技術文献】
【特許文献】
【０００９】
【特許文献１】特開２０１１－７５６１１号公報
【特許文献２】特開２００６－２５４４１７号公報
【特許文献３】特開２００６－６７４１２号公報
【特許文献４】特開２００５－１７６１２号公報
【特許文献５】特開２００９－１０５９７号公報
【発明の概要】
【発明が解決しようとする課題】
【００１０】
上記の従来の暗号化の手法によると、秘匿性の観点からのものが多く、処理の過程に含まれる割算など演算コストが高い処理の低減につながるようなものではなかった。本実施形態では、演算コストの低減を図ることが可能な公開鍵暗号システム、公開鍵暗号方法、公開鍵暗号プログラムを提供する。
【課題を解決するための手段】
【００１１】
実施形態に係る公開鍵暗号システムは、公開鍵を用いて平文を暗号化し暗号文を送る送信装置と、送信された暗号文を受信し、前記公開鍵に対応する秘密鍵を用いて復号し平文へ戻す受信装置とを備える公開鍵暗号システムにおいて、前記送信装置は、前記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換する暗号化手段を備えており、前記受信装置は、前記秘密鍵を用いてベルヌーイシフト写像を実行して暗号文を平文へ変換する復号手段を備えていることを特徴とする。
【図面の簡単な説明】
【００１２】
【図１】本発明に係る公開鍵暗号システムの第１の実施形態の構成を示すブロック図。
【図１Ａ】本発明に係る公開鍵暗号システムの第２の実施形態の構成を示すブロック図。

【図１Ｂ】本発明に係る公開鍵暗号システムの第３の実施形態の構成を示すブロック図。
【図１Ｃ】本発明に係る公開鍵暗号システムの第４の実施形態の構成を示すブロック図。
【図１Ｄ】本発明に係る公開鍵暗号システムの第５の実施形態の構成を示すブロック図。
【図２】本発明に係る公開鍵暗号システムの実施形態において用いるベルヌーイシフト写像のマップを示す図。
【図３】本発明に係る公開鍵暗号システムの実施形態において用いるベルヌーイシフト写像の結果の時系列を示す図。
【図４】傾き係数Ａ＝２の時のベルヌーイシフト写像と合同算術の時系列を示す図。
【図５】傾き係数Ａ＝５のベルヌーイシフト写像マップを示す図。
【図６】係数Ａ＝５の時のベルヌーイシフト写像と合同算術の時系列を示す図。
【図７】ベルヌーイシフト写像と合同算術のＣ言語によるソースコードを示す図。
【図８】図７のベルヌーイシフト写像と合同算術による演算結果を示す図。
【図９】一次元不定方程式の整数解によって秘密鍵ＳＫとなる“ｘ”と“ｙ”を求める過程を示す図。
【図１０】本発明に係る公開鍵暗号システムの第１の実施形態の動作を示すフローチャート。
【図１１】ベルヌーイシフト写像による平文３通りの暗号化の結果）と復号の結果を示す図。
【図１２】ベルヌーイシフト写像による公開鍵暗号の暗号化・復号イメージを示す図。
【図１３】本発明に係る公開鍵暗号システムの第３の実施形態の動作を示すフローチャート。
【図１３Ａ】本発明に係る公開鍵暗号システムの第４の実施形態の動作を示すフローチャート。
【図１４】ベルヌーイシフト写像の初期値X_0を変更したときのX_i
【図１５】本発明に係る公開鍵暗号システムの第５の実施形態の動作を示すフローチャート。
【発明を実施するための形態】
【００１３】
以下添付図面を参照して、本発明に係る公開鍵暗号システム、公開鍵暗号方法、公開鍵暗号プログラムの実施形態を説明する。
【００１４】
本発明の実施形態として、公開鍵を元に暗号化を行い、秘密鍵を保有している受信装置により復号を行うベルヌーイシフト写像による公開鍵暗号の演算を用い、安全で演算コストが低い公開鍵暗号システムを提案する。従来の代表的な公開鍵暗号の一つとして、ＲＳＡ暗号が知られている。ＲＳＡ暗号で用いられている数理は、フェルマーの小定理を素数以外の数値に適用できるように拡張したオイラーの定理に基づくものである。
【００１５】
本願発明者は、一次元写像として知られるベルヌーイシフト写像を整数演算化し、傾きを一律に変更して反復する値を追跡すると合同算術の定理で知られるフェルマーの小定理と同期することを発見した。この点に着目するとオイラーの定理もベルヌーイシフト写像の反復周期と同期するため、ＲＳＡ暗号の合同算術による演算をベルヌーイシフト写像による公開鍵暗号として適応可能である。
【００１６】
ＲＳＡ暗号は、法演算（ｍｏｄｕｌｏ）である合同算術を基本としており、この合同算術には割り算が必須である。これに対し、本発明の実施形態に係るベルヌーイシフト写像による公開鍵暗号アルゴリズムでは割り算を行わず、引き算を１回のみ行えば良いため、合同算術を用いるものと比較して演算コストを削減できる。また、ＲＳＡ暗号の弱点としては、決まり切った平文が“Ｙｅｓ”か“Ｎｏ”のみにより構成される場合、同じ平文を同じ公開鍵で暗号化を行うと暗号文が限定されるため、平文が見破られやすいといった問題があることは前述したことである。これを解決するため、平文に乱数を付加して送信す

る手法があるが、受信側の手間が増加することについても述べた。

【００１７】

上記に対し本実施形態に係るベルヌーイシフト写像よる公開鍵暗号の演算では、初期値X_0を変更できる自由度を有する。即ち、初期値X_0を初期ベクターとして変更することで、同じ平文を同じ公開鍵で暗号化しても暗号文のパターンが毎回変動されるので、同じ平文を使用していることが第三者に悟られ難くなることが期待できる。

【００１８】

公開鍵暗号方式は、共通鍵暗号方式の鍵の配送の問題から共通鍵を通信相手に渡す際に公開鍵暗号を利用するハイブリット型暗号として用いられているが、同じ共通鍵を使い回し続ける場合は同じ暗号文が出力されるため安全性が小さくなる。実施形態に係るベルヌーイシフト写像を用いた公開鍵暗号システムでは、初期値X_0を乱数で毎回変更することで毎回異なる暗号文が生成できる構成が採れるため同じ鍵を使用していることが第三者にわからなくなり安全性を高めることが期待できる。このため、同じパスコードを使い回す用途にも好適となる鍵交換方法となる。本実施形態の暗号の安全性は、ＲＳＡ暗号と同様に大きな桁の素数同士の掛け算の合成数は、素因数分解に莫大な計算量がかかることを根拠とする。

【００１９】

次に、図を用いて実施形態を説明する。各図において、同一の構成要素には同一の符号を付して重複する説明を省略する。第１の実施形態に係る公開鍵暗号システムは図１に示すように、送信装置１００と受信装置２００とを備えて構成される。

【００２０】

送信装置１００と受信装置２００とは、コンピュータ機能と通信機能を備える装置であるならば、特に限定されず、パーソナルコンピュータやワークステーション、スマートフォン、ＰＤＡ（パーソナルディジタルアシスタント）などによって構成することができる。送信装置１００は、公開鍵を用いて平文を暗号化し暗号文を送るものであり、送信のための送信手段１０１を備えている。受信装置２００は、送信された暗号文を受信し、上記公開鍵に対応する秘密鍵を用いて復号し平文へ戻すものであり、受信のための受信手段２０１を備えている。

【００２１】

送信装置１００と受信装置２００とは、有線または無線の伝送路３００により接続されている。送信装置１００は、暗号化手段１０２を備えている。暗号化手段１０２は、上記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換するものである。受信装置２００は、復号手段２０２を備えている。復号手段２０２は、上記秘密鍵を用いてベルヌーイシフト写像を実行して暗号文を平文へ変換するものである。

【００２２】

受信装置２００には、上記公開鍵を生成する公開鍵生成手段２０３と上記秘密鍵を生成する秘密鍵生成手段２０４が備えられている。上記図１の構成を備える本実施形態の公開鍵暗号システムは、基本的にベルヌーイシフト写像を用いるものであるから、ベルヌーイシフト写像の説明から始める。

【００２３】

○ベルヌーイシフト写像について

ベルヌーイシフト写像は以下の式（１）で定義される。

【数１】

$$x_{i+1} = \begin{cases} 2x_i & (x_i < 0.5) \\ 2x_i - 1 & (0.5 \leq x_i) \end{cases} \qquad i = 0,1,2,\ldots \qquad (1)$$

ベルヌーイシフト写像のマップを図２に示し、式（１）による横軸をｉ、縦軸をx_{i+1}とした時系列を図３に示す。

【００２４】

図３では、初期値x_0＝０．２３４とし、“x_0＜０．５”であるため、“x_1＝２x_0”の演算を行い、x_1＝０．４６８が得られる。x_1については、“x_1＜０．５”であるため、“x_2＝２x_1”の演算を行い、x_2＝０．９３６が得られ、以降“x_3，x_4，…”と開区間（０，１）を反復し遷移してゆく。

【００２５】

次に、ベルヌーイシフト写像と合同算術との同期について説明する。初等整数論の合同算術における重要な定理として、フェルマーの小定理が知られている。Ｐを素数とし、ＡをＰの倍数でない整数（ＡとＰは互いに素［最大公約数が１］、つまりＰが素数であればよい）とするとき、以下の式が成り立つことが知られている。

【数２】

$$A^P \equiv A \pmod P$$

【００２６】

例えば，$2^{11} \equiv 2$（ｍｏｄ１１）の場合2^{11}＝２０４８に対して，素数１１で割る演算を行うと、余りが２となることを示し、１１の法（ｍｏｄｕｌｏ）をとった剰余は２であることを表す式である。また、“Ｐ－１”を乗数とした（両辺をＡで割る）場合には、以下式（２）のように余りが１となる。

【数３】

$$A^{P-1} \equiv 1 \pmod P \qquad (2)$$

【００２７】

ここで、式（１）の初期値x_0に“１／１１”を設定し、反復演算したx_iの値と、式（２）に“Ａ＝２”を設定し、法として素数Ｐ＝１１を設定し、左辺の指数Ｐを０～１１に振った余りの値は、図４に示すようになる。ここで、式（２）について指数が大きくなるほど桁が莫大になるため、合同算術では法（ｍｏｄｕｌｏ）をとることで、ある値Ｑ同士を掛け算した結果の余りにもう一回ある値Ｑを掛け算して、余りを出すと元の掛け算（Q^3）の余りに等しくなるといった性質があり、桁を抑えて演算できるため、このテクニックを使って計算している。

【００２８】

図４（Ａ）に示すように、ベルヌーイシフト写像は分数で計算を行っており、この計算におけるX_iの分子の数値が、図４（Ｂ）の合同算術による余りの数値と同じになっており、同期していることが判る。これは、割り切れない数は仮分数（2^i／１１）が、例えばｉ＝６のとき、６４／１１＝５５／１１＋９／１１＝５＋９／１１となり、帯分数は１１の倍数として、５が括りだされ、真分数（９／１１）における分子の値“９”は１１の割り算の余りとしても示されるからである。以上の図４から、x_{10}の時点で１／１１とX_0に戻り、ベルヌーイシフト写像は周期長１０で繰り返されることが判る。

【００２９】

次に、式（２）のＡが２以外の数値のときに対応するベルヌーイシフト写像の式を考える。例としてＡ＝５の場合を考えて、これと同期するベルヌーイシフト写像は以下式（３）の５つからなる式になる。式（３）のマップを図５に示す。

【数４】

$$x_{i+1}=\begin{cases}5x_i & (x_i < 1/5)\\ 5x_i-1 & (1/5 \le x_i < 2/5)\\ 5x_i-2 & (2/5 \le x_i < 3/5)\\ 5x_i-3 & (3/5 \le x_i < 4/5)\\ 5x_i-4 & (4/5 \le x_i)\end{cases} \qquad i=0,1,2,\ldots \qquad (3)$$

【００３０】

式（３）に、初期値X_0＝１／１１を与えて反復演算を行った場合の値と、式（２）において“Ａ＝５”を設定し、法として素数Ｐ＝１１を設定した値の遷移を、図６に示す。図６を参照するとＡ＝２の場合と同様に、図６（Ａ）に示される式（３）の反復演算におけるX_iの分子と、図６（Ｂ）に示される合同算術による余りの数値が同じ値となっており、同期している。これも、分数の演算では真分数の分子が合同算術の余りを指しているからであり、傾きＡがどのような値でも同様に真分数の分子と合同算術の余りが同じ値になることが判る。そして、X_5＝１／１１となっており、初期値X_0に戻るため周期長は５で繰り返されることが判る。これらを考察すると、ベルヌーイシフト写像の反復値の真分数の分子は合同算術の余りに相当し、合同算術の余りと同義であることが判る。以上から、傾きＡの冪乗の法を取る合同算術は、ベルヌーイシフト写像に代替できることが示された。

【００３１】

ここまでベルヌーイシフト写像は開区間（０，１）を扱っていたが、整数演算が行えるよう区間を素数Ｐ倍して開区間（０，Ｐ）にすることで、冪乗の合同算術の余りに相当するX_iを得ることができる。傾きを“Ａ”とし、開区間（０，Ｐ）に拡大し整数演算化したベルヌーイシフト写像は各傾きＡの一次式をまとめて、次の式（４）のように、簡潔に表すものとする。

【数５】

$$X_{i+1}=AX_i-PM_k \quad (PM_k/A \le X_i < PM_{k+1}/A) \qquad (4)$$
$$(\quad i=0,1,2,\ldots;\ M_k=0,1,2,\ldots,A;\ k<A)$$

上記式（４）のM_kの下付き添え字ｋは、式（４）の各一次式（一次式の数はＡ個）を選択する範囲毎の変数を示し、ｋは０（M_0＝０）から傾きＡ（M_{k+1}＝Ａ）までを表している。例えば、Ｐ＝１（拡大無し）とすれば、傾きＡ＝２では式（１）、傾きＡ＝５では式（３）が得られる。

【００３２】

図７（Ａ）に、整数演算化したベルヌーイシフト写像の式（４）をＣ言語で記述したプログラムのソースコードを示し、図７（Ｂ）に、合同算術の式（２）をＣ言語で記述したプログラムのソースコードを示す。図７のプラグラムを実行した演算結果、を図８に示す。図８（Ａ）がベルヌーイシフト写像式（４）を係数Ａ＝５、最大区間Ｐ＝１１、反復回数ｒｅｔ＝１１に設定して、X_{10}まで反復を行った場合の結果であるX_iの値を示す。図８（Ｂ）は、合同算術式（２）を係数Ａ＝５、法Ｐ＝１１として左辺のべき乗のＰ－１を“０，１，２，…，１０”（ｒ＝１１に設定）まで変更して、それぞれの５のべき乗を法１１で割った余りを示している。この図８によれば、図８（Ａ）に示されるベルヌーイシフト写像の反復ｉの値X_iと、冪乗の値（5^i（ｍｏｄ１１））が対応して、同じ解となっていることが確認できる。

【００３３】

なお、図７（Ａ）のベルヌーイシフト写像式（４）のプログラムでは、各区間の境界“$P\times M_k$／Ａ”や“$P\times M_{k+1}$／Ａ”が割り切れるよう区間（０，Ｐ）をさらにＡ倍することで、最大区間（０，Ａ×Ｐ）とし各区間の境界を整数のみで扱えるようにしている。具

体例には、係数Ａ＝５の場合、区間の境界値は式（４）から“０／５，１１／５，２２／５，３３／５，４４／５，５５／５”となるが、割り切れないため５倍して“０，１１，２２，３３，４４，５５”の値を境界値として求めておき、メモリ（配列ｉｎｔｅｒｖａｌ［Ａ］）に保存している。X_iの値に対応する区間の検索処理（図７のプログラムでは降順に全件探索し見つかった時点で検索終了）により引き算する値“$P \times M_k$”を選定している。このため、プログラム中の演算結果をモニターに出力する“ｐｒｉｎｔｆ”文ではX_iはＡで割ってから出力し、関数の返り値もX_iは、Ａで割った値としている。

【００３４】

合同算術の式（２）の合同算術演算はＣ言語で実装した図７（Ｂ）に示すソースコードのように、法１１の余りに対して、Ａ＝５を掛け算して法１１の余りを取りＡ＝５を掛け算する繰り返しで余りの算術を行う手法を使っている。これに対してベルヌーイシフト写像の式（４）では区間の境界値を保存しておくメモリ領域と区間を検索する処理が発生するが演算が１回の引き算のみで良く、割り算を行い余りをとる除算剰余と比較して演算コストの低減が期待できる。

【００３５】

実際に図７のソースコードの演算の繰り返し部分を、１００万回反復（関数の入力値にＡ＝５，Ｐ＝１１，図７（Ａ）：ｒｅｔ=1000000，図７（Ｂ）：ｒ=1000000に設定して“ｐｒｉｎｔｆ”部分はコメントアウト）を行いオペレーションシステムＬｉｎｕｘ（登録商標）で提供されているコマンドツール“ｔｉｍｅ”を使って処理時間を調べると、ベルヌーイシフト写像による演算は合同算術の約半分の処理時間になることが確認された。式（４）のベルヌーイシフト写像は、傾きＡが大きな値となることで区間の分割数が増加し検索に時間がかかると推測されるが、区間の検索処理は二部探索法を利用するなど検索コストを削減することがより望ましい。

【００３６】

以上の考察と実験結果から、合同算術によるべき乗とベルヌーイシフト写像の反復回数は同期して、合同算術によるべき乗の余りとベルヌーイシフト写像のX_iは同じ値となる関係が得られることが推定される。よって、ＲＳＡ暗号で用いられているフェルマーの小定理を素数以外の数にも適応できるよう拡張されたオイラーの定理についてもベルヌーイシフト写像と同期することが推定される。

【００３７】

○オイラーの定理について

オイラーの定理は、Ｎが正の整数でＡをＮと互いに素な正の整数としたとき、以下の式（５）が成り立つ、という定理である。

【数６】

$$A^{\phi(N)} \equiv 1 (\mathrm{mod}\, N) \qquad (5)$$

上記式（５）におけるφ（Ｎ）は、オイラーのトーシェント関数と呼ばれ、φ（Ｎ）はＮより小さい正の整数のうちＮと互いに素な数の個数を表す。例えばＮ＝１０のときを考えると１０と互いに素な１０より小さい数をピックアップすると｛１，３，７，９｝の４個となるため、φ（１０）＝４となる。

【００３８】

また、Ｎが素数Ｐの場合、Ｐと互いに素な数は｛１，２，…，Ｐ－２，Ｐ－１｝のＰ－１個となるため、φ（Ｐ）＝Ｐ－１となり、式（５）はフェルマーの小定理の式（２）になることが判る。このため、オイラーの定理はフェルマーの小定理を素数以外にも適応できるようにした拡張版とされる。特に、素数Ｐと素数Ｑを用意して“Ｎ＝Ｐ×Ｑ”としたときには、“φ（Ｎ）＝（Ｐ－１）×（Ｑ－１）”が成り立つことが知られており、ＲＳＡ暗号では法（ｍｏｄｕｌｏ）の値としてこの２つの素数を掛け算したＮを公開鍵の一部として使用する。注意点として、ＡとＮは互いに素な整数であることが条件のため、Ａは

素数Ｐもしくは素数Ｑを使用しないことである。
【００３９】
○ＲＳＡ（Ｒｉｖｅｓｔ－Ｓｈａｍｉｒ－Ａｄｌｅｍａｎ）暗号について
ＲＳＡ暗号は、合同算術による冪乗の余りの周期性に着目したオイラーの定理に基づく公開鍵暗号方式である。公開鍵暗号方式は、第三者に公開する公開鍵により暗号化を行い、秘密鍵を持った本人のみが復号を行えるといった暗号化と復号に別々の鍵を用いる暗号方式である。送信者と受信者が同じ鍵を共有する共通鍵暗号方式は、鍵の配送の問題があるため共通鍵の交換や認証にも広く利用されている。公開鍵はＰＫ（Ｐｕｂｌｉｃ　Ｋｅｙ）と法Ｎとし、秘密鍵はＳＫ（Ｓｅｃｒｅｔ　Ｋｅｙ）と法Ｎとなる。これらの鍵はベルヌーイシフト写像よる暗号化と復号の演算でＲＳＡ暗号と等しく利用できるため、本実施形態では、これを用いるものであり、以下に鍵の生成方法について説明する。
【００４０】
ＲＳＡ暗号では、平文をＰＬ（Ｐｌａｉｎ）とし公開鍵ＰＫと公開鍵Ｎとすると、暗号文Ｃ（Ｃｉｐｈｅｒ）は２つの公開鍵を利用して以下の合同算術式（６）で得る。
【数７】

$$PL^{PK} \equiv C \pmod{N} \qquad (6)$$

暗号文Ｃから復号鍵となる秘密鍵ＳＫを用意し、以下の合同算術の式（７）で復号を行い平文に復元する。
【数８】

$$C^{SK} \equiv PL \pmod{N} \qquad (7)$$

ここで式（６）に式（７）を代入し暗号文Ｃを消去すると、
【数９】

$$(PL^{PK})^{SK} \equiv PL \pmod{N} \qquad (8)$$

と復号を行うことができる。
【００４１】
ＲＳＡ暗号では、公開鍵ＰＫと公開鍵Ｎで式（６）の暗号文Ｃを生成し、復号鍵（秘密鍵）ＳＫと復号鍵Ｎを持つ本人だけが暗号文Ｃから正しい平文ＰＬを得られる仕組みとなっており、式（８）が成り立つようなＰＫとＳＫを見つけることで公開鍵暗号として成立する。式（８）を成立させることを考えると、式（５）のオイラーの定理を利用することができる。式（５）のＡに平文ＰＬを代入して正の整数“ｙ”を導入し、ｙ乗すると、以下の式（８Ａ）が得られる。
【数１０】

$$PL^{\phi(N)\cdot y} \equiv 1 \pmod{N} \qquad (8A)$$

【００４２】
上記式（８Ａ）において、冪乗のφ（Ｎ）をｙで整数倍して法をとった値、周期長φ（Ｎ）で“１”を繰り返すことを示している。前述したように“Ｎ＝Ｐ×Ｑ”としたときには、“φ（Ｎ）＝（Ｐ－１）×（Ｑ－１）”が成り立ち、Ｎは素数Ｐと素数Ｑを掛け算したもの（素数同士の掛け算はＲＳＡ暗号の安全性根拠になる）を採用する。更に、式（８Ａ）の両辺に“ＰＬ”を掛け算すると、以下の式（９）となる。

【数１１】

$$PL^{\phi(N)\cdot y+1} \equiv PL(\mathrm{mod}\ N) \qquad (9)$$

【００４３】
式（８）と式（９）の指数を見比べると、
【数１２】

$$PK \cdot SK = \phi(N) \cdot y + 1 \quad (9A)$$

が成り立つようなＰＫとＳＫを見つけることで、式（８）は成り立つため暗号文Ｃは復号できる。ここで、上記式（９Ａ）の右辺の“φ（Ｎ）・ｙ”を左辺に移項すると以下の式（１０）が得られる。
【数１３】

$$PK \cdot SK - \phi(N) \cdot y = 1 \quad (10)$$

式（１０）からベズーの等式としても知られる一次不定方程式の整数解を利用することで、ＰＫとＳＫが導出できる。以下に、一次不定方程式の解法により具体的な例を提示しＰＫとＳＫの導出例を示す。
【００４４】
＜定理＞ 一次不定方程式の整数解
“ａ”と“ｂ”を互いに素な整数としたとき、以下の二元一次方程式を満たす整数解“ｘ”と“ｙ”が存在する。
【数１４】

$$a \cdot x + b \cdot y = 1 \qquad (11)$$

式（１０）と式（１１）を見比べると、ａ＝ＰＫ，ｘ＝ＳＫ，φ（Ｎ）＝ｂに相当することが判る。“ｙ”は“－ｙ”に置き換える。
前述から、“φ（Ｎ）＝（Ｐ－１）×（Ｑ－１）”のため、例として素数Ｐ＝１１と素数Ｑ＝１３を用意してＮとφ（Ｎ）を求めると、
Ｎ＝１１×１３＝１４３，φ（Ｎ）＝１０×１２＝１２０
となる。次にＰＫを導出するが式（１１）は定理より、“ａ”と“ｂ”を互いに素（最大公約数が１）な整数とするため、φ（Ｎ）＝１２０と素な整数を公開鍵ＰＫ（＝ａ）としてランダムに選択できる。
【００４５】
なお、式（５）は、φ（Ｎ）は“Ｐ－１”と“Ｑ－１”の最小公倍数ＬＣＭ（Ｌｅａｓｔ Ｃｏｍｍｏｎ Ｍｕｌｔｉｐｌｅ）をとっても成り立つため、最小公倍数をとる関数をＬＣＭ（Ｐ－１，Ｑ－１）として“φ（Ｎ）”を置き換え、“Ｎ＝Ｐ×Ｑ”のため、これも置き換えて
【数１５】

$$A^{LCM(P-1,Q-1)} \equiv 1\ (\mathrm{mod}\ (P \times Q))$$

が成り立つ。上記から１０と１２で共に割り切れる最小の値（最小公倍数）は６０となる

ため、“ＬＣＭ（１０，１２）＝６０”であり、上記の式の素数Ｐか素数Ｑ以外の適当な整数Ａの指数に“６０”を当て嵌めると“６０，１２０，１８０，…”と６０刻みに周期的に“Ｐ×Ｑ”で法をとった値は１になる。

【００４６】

公開鍵ＰＫ（＝ａ）の選択は、文献によってφ（Ｎ）＝ＬＣＭ（Ｐ－１，Ｑ－１）となる最小公倍数を当て、それより小さい素な値の

“０＜ＰＫ＜ＬＣＭ（Ｐ－１，Ｑ－１）”を採用することが紹介されているが、ここでも最小公倍数“φ（Ｎ）＝６０”を使う。

公開鍵ＰＫ（＝ａ）は、例として適当に素数７（６０と素な値）を当てることとする。

これで式（１１）の“ａ”と“ｂ”は以下のように決まった。

【数１６】

$$7 \cdot x + 60 \cdot y = 1$$

一次不定方程式の整数解「拡張されたユークリッドの互除法」によって秘密鍵ＳＫとなる“ｘ”と“ｙ”を求める。“ｘ”と“ｙ”を求める例を図９に示す。

【数１７】

$$7 \cdot (-17) + 60 \cdot 2 = 1$$

【００４７】

上記式のように“ｘ＝－１７，ｙ＝２”が得られたが、“ｘ”はマイナス、“ｙ”はプラスになっており式（１０）と見比べると、”ｙ”はマイナスつまり“ｙ＝－ｙ”となるようにしたいため“ｘ”がプラスとなるように、以下のテクニックを利用する。式（１１）を以下の式（１２）に変形する。

【数１８】

$$a \cdot (x + b) + b \cdot (y - a) = 1 \qquad (12)$$

式（１２）の“ａ×ｂ”の部分は差し引きで消去できるため式（１１）と等しい。

求めた値は“ｘ＝－１７，ｂ＝６０”、また“ｙ＝２，ａ＝７”であるため、これを式（１２）に代入すると、

【数１９】

$$7 \cdot (43) + 60 \cdot (-5) = 1$$

と“ｘ＝４３，ｙ＝－５”になり、式（１０）の形式が得られこれで残りの秘密鍵ＳＫ＝４３が得られた。

以上、ＲＳＡ暗号は平文を底の数値として、公開鍵ＰＫで冪乗を行い“Ｎ＝素数Ｐ×素数Ｑ”で法をとり暗号文を生成し、暗号文を底の数値として秘密鍵ＳＫで冪乗を行い、“Ｎ＝素数Ｐ×素数Ｑ”で法をとり、復号を行い平文に戻すことを示した。

【００４８】

図１に示したように、受信装置２００の公開鍵生成手段２０３が公開鍵を生成し、秘密鍵生成手段２０４が秘密鍵を生成する。生成した公開鍵であるＰＫとＮを、第三者としての送信装置１００のユーザに公開する。Ｎを素因数分解して素数Ｐと素数Ｑを求めて式（１０）にＰＫとφ（Ｎ）を当て嵌めれば、秘密鍵ＳＫが容易にわかってしまうが、大きな桁の素数同士を掛け算した合成数の素因数分解は、莫大な計算量が必要となり素数Ｐと素数Ｑが割り出せないことを安全性の根拠としている。２０１１年１月に、米国商務省国立

標準技術研究所（ＮＩＳＴ）は、ＲＳＡ暗号に使用する法Ｎを２０４８ビット以上にすることを公表（ＮＩＳＴ　ＳＰ８００－１３１）している。

【００４９】

なお、他に秘密鍵ＳＫの導出方法については、文献によって、

【数２０】

$$PK \cdot SK \equiv 1 \pmod{\phi(N)}$$

が成り立つ”ＳＫ”を見つけることであると、紹介されている。しかしながら、これは被除数ＰＫ×ＳＫを除数φ（Ｎ）で割った商ｙの剰余が１ということであり、以下の式が成り立つ。

【数２１】

$$\phi(N) \cdot y + 1 = PK \cdot SK$$

この式は、式（１０）と同じであり、本実施形態では一次不定方程式の整数解により秘密鍵ＳＫを導出することとした。

【００５０】

＜第１の実施形態＞

以上のようにして、求められた公開鍵と秘密鍵を用いて、図１に示した第１の実施形態に係る公開鍵暗号システムは、図１０に示すフローチャートに基づく動作を行い、ベルヌーイシフト写像よる暗号化と復号を実行する。なお、図１０では素数Ｐと素数Ｑから公開鍵ＰＫと秘密鍵ＳＫと法Ｎは受信装置２００において生成済みであり、公開鍵ＰＫと法Ｎは送信装置１００へ渡されているものとする。

【００５１】

即ち、送信装置１００は、公開鍵ＰＫと公開鍵Ｎ（法Ｎ）を入手し（Ｓ１１）、平文を入手し（Ｓ１２）、公開鍵ＰＫをベルヌーイシフト写像の反復回数にセットし、公開鍵Ｎをベルヌーイシフト写像の最大区間にセットし、平文をベルヌーイシフト写像の傾き係数にセットし（Ｓ１３）、準備が完了となる。

【００５２】

次に、ベルヌーイシフト写像により暗号化を行う（Ｓ１４）。このステップＳ１４の処理は、図７（Ａ）に示したプログラムの実行による処理となる。ステップＳ１４の処理が行われ、反復回数ＰＫ繰り返された結果の値X_{PK}が暗号文として生成される（Ｓ１５）。この暗号文X_{PK}は受信装置２００へ伝送路３００を介して送信される。

【００５３】

一方受信装置２００は、秘密鍵ＳＫと秘密鍵Ｎ（法Ｎ）を入手し（Ｓ２１）、暗号文を受信する（Ｓ２２）。次に、秘密鍵ＳＫをベルヌーイシフト写像の反復回数にセットし、秘密鍵Ｎをベルヌーイシフト写像の最大区間にセットし、暗号文をベルヌーイシフト写像の傾き係数にセットし（Ｓ２３）、準備が完了となる。

【００５４】

次に、ベルヌーイシフト写像により復号を行う（Ｓ２４）。このステップＳ２４の処理は、図７（Ａ）に示したプログラムの実行による処理となる。ステップＳ２４の処理が行われ、反復回数ＳＫ繰り返された結果の値X_{SK}が復号文（平文）として生成される（Ｓ２５）。

【００５５】

上記公開鍵暗号システムの処理を、具体例により説明する。例として、３つの平文（Ａ＝５，Ａ＝１９，Ａ＝１０１）を用意し、公開鍵による暗号化と秘密鍵による復号したときのそれぞれの反復X_iの値を図１１に示す。前述したが、平文Ａには素数Ｐ＝１１もしくは素数Ｑ＝１３と同じ値を入れない。このため、実用では平文は素数Ｐと素数Ｑ以下の値にするなど制約を与える。各平文は公開鍵の値となる７回分を式（４）で反復演算を行

なった図１１（Ａ）の“ｉ＝７”の行に示す“X_7”が暗号文となる。暗号文に秘密鍵の値となる４３回を式（４）で反復演算を行ったX_{43}が、元の平文に復号されていることが図１１（Ｂ）の“ｉ＝４３”の行において確認できる。なお、初期値X_0は暗号化と復号で共に“１”に設定している。

【００５６】

この暗号化と復号のイメージを、ベルヌーイシフト写像のマップとして図１２に示す。この図１２によって、平文を５としたときの式（４）における傾きがＡ＝５になり、復号鍵分（ＰＫ＝７）の回数の反復を行って暗号化することで暗号文X_7＝４７が得られることが判る。復号時はX_7＝４７を式（４）の傾きＡ＝４７に設定して、秘密鍵分の回数（ＳＫ＝４３）の反復写像を行うことによってX_{43}＝５の平文５が復元される。

【００５７】

演算負荷について考察する。図７のソースコードに示すように、傾きＡの値で区間の分割数が決まるため、例えば送信者がパスワード８文字のような６４ｂｉｔのデータを送信する場合、８ｂｉｔずつ小分けにして平文（傾きＡ）を小さくして暗号化を行い８回分送信する。これにより、１回（８ｂｉｔ）の暗号処理における区間の最大分割数（最大２５５）が少なくなり、区間の検索回数を少なくできる。送信する暗号文X_i（復号に使用する傾きＡ）は法Ｎの値により大きな値になるため、送信側は処理能力の低い携帯端末で暗号化を行い、受信者側は処理能力の高いサーバで復号を行うといったパスコード認証のような形態において好適となる。

【００５８】

従来の合同算術による計算では、図７（Ｂ）のソースコードのステップ数“ｒ”が公開鍵もしくは秘密鍵の値に相当するが、実際には大きな値を扱うため計算量が莫大になり現実的ではなくなる。このため、“冪乗法”を利用することで計算量が少なくなる手法を用いている。

【００５９】

例として図１１の暗号文４７を復号文５に戻す場合は式（７）より以下の合同算術式が成り立っている。

【数２２】

$$47^{43} \equiv 5 \pmod{143}$$

指数部分の秘密鍵ＳＫ＝４３から図７では４３ステップの計算が必要となる。冪乗法ではこの計算に補助項

【数２３】

$$47^2 \equiv 64 \pmod{143},\ 47^4 \equiv 92 \pmod{143},\ 47^8 \equiv 27 \pmod{143},\ 47^{16} \equiv 14 \pmod{143},$$
$$47^{32} \equiv 53 \pmod{143}$$

を計算する。この手法はその前の項の２乗

【数２４】

$$（例えば、\ 47^8 \equiv 92^2 \equiv 27 \pmod{143}\ ）$$

から得られるため計算ステップは５回となる。

【数２５】

$$47^{43} \equiv 47^{32} \cdot 47^8 \cdot 47^2 \cdot 47^1 \equiv 53 \cdot 27 \cdot 64 \cdot 47 \equiv 5 \pmod{143} \quad （１３）$$

となり、計算ステップが３回となるため、合計８回のステップで計算が可能となり演算負荷を軽減できる。

【００６０】
ここまでの計算では法Ｎ＝１４３であるため、最大桁は“１４３^2＝２０４４９”までを扱うことが考えられるが、最大桁を“１４３×２＝２８６”までに小さくする手法がある。たとえば上記の
【数２６】

$$53 \cdot 27 \equiv 1 \pmod{143}$$

の箇所ついて“５３×２７＝１４３１”となるが、２７を２進展開すると
“$2^4+2^3+2^1+2^0$”となり、補助項を求めると、
【数２７】

$$2^0 \cdot 53 \equiv 53 \pmod{143},\ 2^1 \cdot 53 \equiv 106 \pmod{143},\ 2^2 \cdot 53 \equiv 2 \cdot 106 \equiv 69 \pmod{143},$$
$$2^3 \cdot 53 \equiv 2 \cdot 69 \equiv 138 \pmod{143},\ 2^4 \cdot 53 \equiv 2 \cdot 138 \equiv 133 \pmod{143}$$

となり、
【数２８】

$$53 \cdot 27 \equiv 53 \cdot 2^4 + 53 \cdot 2^3 + 53 \cdot 2^1 + 53 \cdot 2^0 \equiv 133 + 138 + 106 + 53 \equiv 1 \pmod{143}$$

となり、足し算の計算は
【数２９】

$$133 + 138 \equiv 128 \pmod{143},\ 128 + 106 \equiv 91 \pmod{143},\ 91 + 53 \equiv 1 \pmod{143}$$

となり、ここまでのすべての途中の計算は最大桁を、“１４３×２＝２８６”以下にできている。
【００６１】
また、
【数３０】

$$1 \cdot 64 \equiv 64 \pmod{143},\quad 64 \cdot 47 \equiv 5 \pmod{143},$$

であるが、同様に４７を２進展開すると“$2^5+2^3+2^2+2^1+2^0$”となり、補助項を求めると
【数３１】

$$2^0 \cdot 64 \equiv 64 \pmod{143},\ 2^1 \cdot 64 \equiv 128 \pmod{143},\ 2^2 \cdot 64 \equiv 2 \cdot 128 \equiv 113 \pmod{143},$$
$$2^3 \cdot 64 \equiv 2 \cdot 113 \equiv 83 \pmod{143},\ 2^4 \cdot 64 \equiv 2 \cdot 83 \equiv 23 \pmod{143},$$
$$2^5 \cdot 64 \equiv 2 \cdot 23 \equiv 46 \pmod{143}$$

となり、
【数３２】

$$64 \cdot 47 \equiv 64 \cdot 2^5 + 64 \cdot 2^3 + 64 \cdot 2^2 + 64 \cdot 2^1 + 64 \cdot 2^0 \equiv 46 + 83 + 113 + 128 + 64$$
$$\equiv 5 \pmod{143}$$

が最終的に導かれ、足し算部分は上記のように一回の足し算の余りの結果を次の値と足し算して余りを順に求めて行く。
【００６２】
式（１３）と比較すると、計算ステップ数は増えているがすべての途中の計算は最大桁を“１４３×２＝２８６”以下にでき、桁数を抑えることができる。こうした演算に必要な最大桁を小さくするテクニックを取り込み、図７に示すように合同算術部分をベルヌーイシフト写像の演算に置き換えることで、傾きＡの桁数が小さくなり区間の検索回数を減

らすことができるため、演算負荷を小さくすることが期待できる。

【００６３】

図１Ａには、第２の実施形態に係る公開鍵暗号システムが示されている。第２の実施形態に係る第２の実施形態に係る公開鍵暗号システムでは、送信装置１００Ａがベルヌーイシフト写像の初期値を暗号化の度に変更する送信側初期値変更制御手段１０５を備えており、受信装置２００Ａが、暗号文を復号する度にベルヌーイシフト写像の初期値を、上記送信側初期値変更制御手段１０５によって変更された初期値と同じ値に変更する受信側初期値変更制御手段２０５と、上記受信側初期値変更制御手段２０５により変更された初期値を用いて前記暗号文の逆元を算出する逆元算出手段２０６を備えている。暗号化手段１０２は、送信側初期値変更制御手段１０５により変更された初期値を用いてベルヌーイシフト写像を実行して暗号化を行う。復号手段２０２は、受信側初期値変更制御手段２０５により変更された初期値を用いて得られた逆元の暗号文に対してベルヌーイシフト写像を実行して復号を行う。この逆元については、次の第３の実施形態で詳しく述べる。本実施形態によれば、暗号文の暗号化の度に初期値が変更されるので、決り切った平文として“Ｙｅｓ”か“Ｎｏ”のみにより構成される場合、同じ平文を同じ公開鍵で暗号化するために行い続けると暗号文が限られるため平文が見破られる危険性を低減させる。

【００６４】

図１Ｂには、第３の実施形態に係る公開鍵暗号システムが示されている。この実施形態では、ベルヌーイシフト写像を実行する場合の初期値となる乱数種を生成する乱数種生成手段４００を有する。乱数種生成手段４００は、送信装置１００Ｂと受信装置２００Ｂ以外の装置に設けられ、生成された乱数種は、秘匿した状態で送信装置１００Ｂと受信装置２００Ｂに与えられる。ここでは、伝送路３００を用いた配信の場合に暗号化して送信し、送信装置１００Ｂと受信装置２００Ｂにおいて復号して用いるようにすることができる。勿論、乱数種生成手段４００は、送信装置１００Ｂと受信装置２００Ｂの少なくとも一方に設けられても良い。

【００６５】

前記暗号化手段１０２Ｂは、上記乱数種生成手段４００により生成され送信側初期値変更制御手段１０５を介して与えられた乱数種を初期値とし、この初期値と上記公開鍵を用いてベルヌーイシフト写像を実行して平文を暗号文へ変換する。また、受信装置２００Ｂには、上記暗号文を上記乱数種生成手段４００により生成され受信側初期値変更制御手段２０５を介して与えられた乱数種を用いて上記暗号文の逆元を算出する逆元算出手段２０６が備えられる。復号手段２０２Ｂは、上記逆元算出手段２０６により算出された逆元とされた暗号文に対し、ベルヌーイシフト写像を用いた復号を実行して暗号文を平文へ変換する。

【００６６】

この第３の実施形態による処理のフローチャートを図１３に示す。このフローチャートの処理は図１０のものを変更したもので、送信装置１００Ｂと受信装置２００Ｂにおいては、乱数種の取得ステップ（Ｓ１６，Ｓ２６）が加わっている。そして、送信側では、ステップＳ１７において、公開鍵ＰＫをベルヌーイシフト写像の反復回数にセットし、公開鍵Ｎをベルヌーイシフト写像の最大区間にセットし、平文をベルヌーイシフト写像の傾き係数にセットし、乱数種を初期値X_0にセットし（Ｓ１７）、準備が完了となる。また、受信装置２００Ｂにおいては、暗号文の受信後に乱数種を用いて逆元を求める処理を行い（Ｓ２７）、この逆元算出した暗号文を復号する暗号文にセットする（Ｓ２８）。これ以外の処理は、第１の実施形態による処理と同様である。

【００６７】

上記公開鍵暗号システムの処理を、具体例により説明する。図１３における送信装置１００Ｂによる暗号化のフローチャートにて、平文を５とし初期値$X_0=1$の場合に追加して、例として$X_0=3$と$X_0=67$の２つ初期値X_0を用意した。この２つの$X_0=3$と$X_0=67$を初期値に設定し、復号鍵（ＰＫ＝７）の回数を反復演算して、取得した“X_i”を示す表が図１４である。初期値$X_0=3$に設定した場合は、この暗号文をC_3と表現して

C_3＝１４１が得られている。初期値X_0＝１では暗号文をC_1とすると暗号文C_1＝４７であるのに対し、初期値X_0＝３に変えたことで異なる暗号文が得られることが確認できる。ここで、初期値X_0から初めてX_0，X_1，X_2，・・・と変化する各X_iの値に着目すると、X_0＝３，X_1＝１５，X_2＝７５となっておいる。これらをX_0＝３で各々割ると、X_0＝１，X_1＝５，X_2＝２５となり初期値X_0＝１に設定した場合と同じ値になることが判る。このように、初期値を変更したときに得られる暗号文は、初期値X_0＝１のときの暗号文が異なってしまうので、初期値を変更したことにより得られる暗号文から変更した初期値を用いて、初期値X_0＝１のときの暗号文を求まる処理を、「逆元を算出する処理」という。そして、逆元算出手段が「逆元を算出する処理」を実行することになる。

【００６８】

初期値X_0＝３から始めた場合に暗号文C_3＝１４１となるまでの各暗号文の値について、３で合同算術の逆数をとれば、初期値X_0＝１のときの各々の“X_i”が取得でき、X_7となる暗号文C_1＝４７を取得できる。この場合では、

【数３３】

$$3 \cdot C_1 \equiv 141 \pmod{143}$$

が成り立つ暗号文C_1を求めればよいため、“１４１／３”を計算した結果のC_1＝４７が該当することが判る。ただし、注意点がある。初期値X_0＝１の場合と同等になる暗号文C_1は乱数で決めた初期値X_0で逆数をとることで求められるが、合同算術において乗法逆元は上記合同式を満たすC_1＝４７のみでなく複数の解が得られることである。

【００６９】

上記式は被除数３×C_1を除数１４３で割った商ｙの剰余が１４１ということであり、以下の式が成り立つ

【数３４】

$$143 \cdot y + 141 = 3 \cdot C_1$$

変形すると

【数３５】

$$C_1 = \frac{143 \cdot y}{3} + \frac{141}{3} = \frac{143 \cdot y}{3} + 47$$

となり、“ｙ＝０，３，６，…”とすることで、“C_1＝４７，１９０，３３３，…”と際限なく複数の解が得られて元の暗号文が１つに識別できなくなる。そこで、「初期値X_0とする乱数種と平文の数値は、法Ｎ以下の値とする」と言うルールを設けることで１つに絞る。この例では、C_1＝４７のみが取得されることになる。

【００７０】

次に、初期値X_0＝６７に設定変更した場合は暗号文をC_{67}として（X_7＝）C_{67}＝３が得られており、初期値X_0＝１のときと異なる暗号文が得られることが確認できる。この場合

【数３６】

$$67 \cdot C_1 \equiv 3 \pmod{143}$$

が成り立つ暗号文C_1を求めればよい。以下の式の解が整数になるC_1を求めると、

【数３７】

$$C_1 = \frac{143 \cdot y + 3}{67}$$

“ｙ＝２２，８９，１５６，…”とすることで、“C_1＝４７，１９０，３３３，…”と複数の解が得られる。そこで、前述のルールに従って法Ｎ（ここでは１４３）以下の値とするためC_1＝４７のみを取得することができる。

【００７１】

以上から乱数種を初期値X_0に設定したときに生成された暗号文C_{x0}から、初期値X_0＝１を設定したときに生成される暗号文C_1に戻すとき、以下の式（１４）が成立するC_1を求めればよい。

【数３８】

$$X_0 \cdot C_1 \equiv C_{X0} \pmod{N} \qquad (14)$$

【００７２】

乱数種の初期値X_0は、送信装置（送信者）と受信装置（受信者）でお互い秘匿することが必要である。このための構成を備える第４の実施形態に係る公開鍵暗号システムを、図１Ｃに示す。乱数種生成手段４００は、送信装置１００Ｃに備えられる。この送信装置１００Ｃに備えられている暗号化手段１０２Ｃが、乱数種生成手段４００により生成された乱数種を上記公開鍵を用いて暗号化手段１０２Ｃにおいてベルヌーイシフト写像を実行して暗号化する。更に、暗号化した乱数種を送信手段１０１を介して受信装置２００Ｃへ送る。受信装置２００Ｃでは、受信した暗号化された乱数種を復号手段２０２Ｃでベルヌーイシフト写像を実行して復号して用いる。この第４の実施形態では第３の実施形態と同様に、逆元算出手段２０６が、変更された初期値（乱数種）を用いて逆元の算出を行う。

【００７３】

第４の実施形態における、乱数種の処理と、平文を暗号化して送信する処理及び暗号化された暗号文を復号化する処理を図１３Ａのフローチャートに示す。受信装置２００Ｃにおいては、素数Ｐと素数Ｑを生成し、ＰとＱから公開鍵Ｎ＝ＰｘＱと公開鍵ＰＫを作成し（Ｓ４１）、公開鍵ＰＫと公開鍵Ｎを送信者に送付する（Ｓ４２）。更に、公開鍵Ｎと公開鍵ＰＫから秘密鍵ＳＫを生成する（Ｓ４３）。送信装置１００Ｃでは、乱数種を生成し上記で受信装置２００Ｃから到来した公開鍵を用いてベルヌーイシフト写像による乱数種の暗号化を行う（Ｓ３１）。次に暗号化した乱数種を受信装置２００Ｃへ送る（Ｓ３２）。これに対し、受信装置２００Ｃでは、受信した暗号化した乱数種を秘密鍵を用いてベルヌーイシフト写像による復号を行い乱数種を復元する（Ｓ４４）。以上で、送信装置１００Ｃと受信装置２００Ｃには同じ乱数種が暗号文の生成と復号の度に更新されて揃うことになる。

【００７４】

次に、送信装置１００Ｃが、公開鍵を用いて乱数種を初期値X_0に設定したベルヌーイシフト写像による平文の暗号化を行い（Ｓ３３）、暗号文を受信装置２００Ｃへ送信する（Ｓ３４）。受信装置２００Ｃでは、送信装置１００Ｃからの暗号文を受け取り、乱数種を逆数として受信した暗号文の逆元を計算する（Ｓ４５）。これで、乱数種を初期値として暗号化された暗号文が初期値が１である暗号文へと変換される（逆元の暗号文が求められる）。受信装置２００Ｃでは、この逆元の暗号文に秘密鍵を用いてベルヌーイシフト写像により平文に復号する（Ｓ４６）。而して、送信装置１００Ｃにおいて暗号化される前の平文が受信装置２００Ｃにおいて安全に復元される。

【００７５】

図１Ｄに第５の実施形態に係る公開鍵暗号システムの構成を示す。本実施形態では、送信装置１００Ｄに設けられた乱数種生成手段４００Ｄは、暗号文の生成に際して乱数種の元となる元乱数種を１つ生成する。この元乱数種は、送信手段１０１を介して受信装置２００Ｄへ与える。乱数種生成手段４００Ｄは、平文の暗号化の毎に上記元乱数種に基づき所定のアルゴリズムで乱数種を生成し送信側初期値変更制御手段１０５を介して暗号化手段１０２Ｄへ与え、暗号化の初期値として用いさせる。

【００７６】

受信装置２００Ｄでは、受信手段２０１Ｄにより受信された暗号化されている元乱数種は復号手段２０２Ｄへ送られて復号されて元乱数種とされる。上記受信装置２００Ｄには、受信側乱数種生成手段２０８が備えられる。受信側乱数種生成手段２０８は、暗号文の復号毎に元乱数種に基づき乱数種生成手段４００Ｄと同じアルゴリズムで乱数種を生成する。受信装置２００Ｄでは、受信側乱数種生成手段２０８により生成された乱数種を用いて復号を行う。即ち、上記で生成された乱数種が逆元算出手段２０６へ与えられ、これを初期値として用いて上記暗号文の逆元を算出する。逆元の算出された暗号文は、復号手段２０２Ｄにより他の実施形態と同様にして復号され平文が得られる。

【００７７】

本第５の実施形態のポイントは、送信装置（送信者）から受信装置（受信者）へ送られる初期値X_0を秘匿とすることにある。図１Ｂに示した第３の実施形態では送信装置が暗号文を受信装置に送付する度に乱数種（初期値X_0）を生成して、それを公開鍵で暗号化して送信し、受信装置側は秘密鍵で復号を行い、乱数種（初期値X_0）を入手することでお互いに秘匿の情報を共有するものであった。しかしながら、この実施形態によれば、通信の度に乱数を生成して暗号化と復号を公開鍵暗号方式で行うことは比較的時間がかかってしまう問題が生じる。

【００７８】

このため、この第５の実施形態では、最初の１回目だけお互い秘匿とする乱数である元乱数種を送受信し、２回目以降はその秘匿の乱数（元乱数種）を種にして秘密情報である初期値X_0を生成すれば、毎回乱数種を送信装置が暗号化して送信する手間と、受信装置が暗号化された乱数種を受信して復号する手間が、削減できる。

【００７９】

本実施形態のフローチャートを図１５に示す。受信装置２００Ｄにおいては、素数Ｐと素数Ｑを生成し、ＰとＱから公開鍵Ｎ＝ＰｘＱと公開鍵ＰＫを作成し（Ｓ４１）、公開鍵ＰＫと公開鍵Ｎを送信者に送付する（Ｓ４２）。更に、公開鍵Ｎと公開鍵ＰＫから秘密鍵ＳＫを生成する（Ｓ４３）。送信装置１００Ｄでは元乱数種を生成して（Ｓ３１Ｄ）、受信装置２００Ｄに送信する（Ｓ３２Ｄ）。送信装置１００Ｄと受信装置２００Ｄは、予め秘匿の乱数種（１つ）を共有している。秘匿で共有するまでの手法に制限はないが、例えば、図１Ｃの第４の実施形態が暗号化毎の乱数種を共有した手法と同じ手法により、受信装置２００Ｄへ送信を行っておくことができる。送信装置１００Ｄと受信装置２００Ｄは、同じアルゴリズムで乱数生成を行う。秘匿の乱数種と送信装置１００Ｄで都度生成した乱数種を受信装置２００Ｄと送信装置１００Ｄで共有する同一の乱数生成アルゴリズムにて乱数種を変換（生成）する（Ｓ３５、Ｓ４７）。

【００８０】

変換した乱数種はベルヌーイシフト写像の初期値X_0に設定して送信装置１００Ｄは図１Ｄの暗号化手段１０２Ｄにより暗号化を行って（Ｓ３３）、暗号文を受信装置２００Ｄへ送信する（Ｓ３４）。受信装置２００Ｄでは、送信装置１００Ｄから受信した元乱数種に基づき受信側乱数種生成手段２０８により乱数種を変換して（Ｓ４７）、受信した暗号文を変換した乱数種で逆元を算出する（Ｓ４５）。逆元演算の結果取得した暗号文は秘密鍵ＳＫを用いて復号を行い平文を得る（Ｓ４６）。このような構成を採ることで、通信の度に生成する乱数種を送信装置１００Ｄで暗号化して受信装置２００Ｄで復号するといった比較的計算コストの高い公開鍵暗号の計算を省くことができる。

【００８１】

受信装置２００Ｄと送信装置１００Ｄで共有する乱数生成アルゴリズムは、例えばベルヌーイシフト写像を用いる。この場合、秘匿の乱数を傾きの値にして送信装置１００Ｄで生成した乱数種は初期値X_0に設定し、所定回数（受信装置２００Ｄと送信装置１００Ｄで秘匿している乱数種が望ましい）を反復演算して変換した乱数種を求めることが考えられる。

【００８２】

また、近年インターネットから銀行の預金口座にアクセス（ログイン）する際にパスワ

ードカードを利用するサービスが提供されている。ここで、サーバで生成される乱数とユーザが持つパスワードカードで生成される乱数は、時間を種として同じ乱数生成変換アルゴリズムから生成される。そして、常にサーバとユーザが持つパスワードカードの乱数が同期できるようお互い秘匿とする乱数として保持できる仕組みとなっており、このような乱数種を用いてもよい。

【００８３】

上記第２～５の実施形態によれば、初期値X_0を送信の度（暗号文生成の度）に毎回初期ベクターとして変更を行ってベルヌーイシフト写像による平文の暗号化を行うので、毎回同じ平文（パスコード）でも通信上は異なる暗号文となるため、第三者（攻撃者）が同一の平文を通信していることが判らなくなり、安全性向上が期待できる。

【００８４】

以上のように、お互い秘匿の乱数種を共通鍵暗号の共通鍵として捕えると公開鍵暗号方式に共通鍵暗号方式を追加した安全性強度を持たせることができる。公開鍵の法Ｎの鍵長はコンピュータの処理性能の向上と共に安全性確保のため年々大きくなる傾向があるが公開鍵の鍵長が大きくなると演算コストも大きくなるため、公開鍵の長さを抑えて秘匿の乱数種を共通鍵として与えて安全性を確保する形態をとることで安全性の強度と演算コストをトレードオフする構成をとることができる。

【００８５】

ＲＳＡ暗号では合同算術による演算を基礎とするが、本実施形態では、ベルヌーイシフト写像で演算することで除算剰余は不要となり、一回の引き算で行えるため演算コストを抑えることができる。また、式（４）の傾きＡの値によって区間の分割数が決まる。そこで本実施形態では、暗号化処理において送信装置は平文を小分けにして平文（傾きＡ）の値を小さくすることによって区間の検索回数の演算コストを少なくしても良い。このため送信側は処理能力の低い携帯端末で暗号化を行い、受信側は処理能力の高いサーバで復号を行うといったパスコード認証のような形態に好適である。

【００８６】

本実施形態では、ベルヌーイシフト写像の初期値X_0を初期ベクターとして乱数種で変更する構成を採用している。このため、同じ平文を同じ公開鍵で暗号化すると乱数種に応じた毎回異なる暗号文を出力するようにでき、同じ平文を暗号化していると第三者に知られる可能性が低下し、安全性が高くなる効果が期待できる。また、本実施形態を、初期値X_0となるお互い秘匿の乱数種を共通鍵暗号の共通鍵を用いるものであるとして捕えると、公開鍵暗号に追加して共通鍵暗号を追加した安全性強度を持たせる効果が期待できる。

【００８７】

公開鍵の法Ｎの鍵長はコンピュータの処理性能の向上と共に安全性確保のため年々大きくなる傾向がある。公開鍵の鍵長が大きくなると演算コストが高くなるため、本実施形態を、公開鍵の長さを抑えて秘匿の乱数種を共通鍵として採用するものであると見立てることで、安全性の強度と演算コストをトレードオフする構成をとることができるものである。

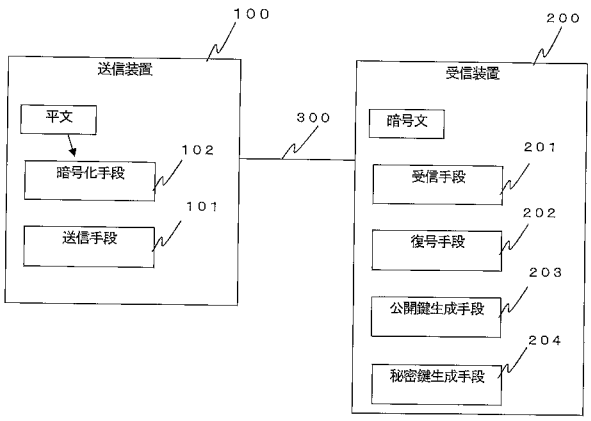
【符号の説明】

【００８８】

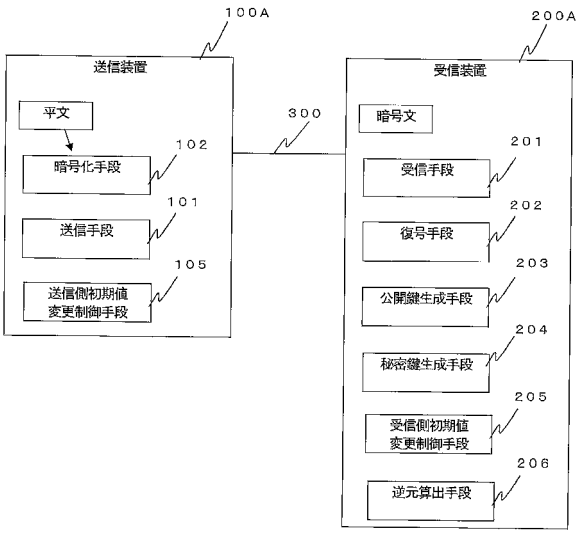
１００、１００Ａ、１００Ｂ、１００Ｃ、１００Ｄ　送信装置
１０１　送信手段
１０２、１０２Ｂ、１０２Ｃ、１０２Ｄ　暗号化手段
１０５　送信側初期値変更制御手段
２００、２００Ａ、２００Ｂ、２００Ｃ、２００Ｄ　受信装置
２０１　受信手段
２０２、２０２Ｂ、２０２Ｃ、２０２Ｄ　復号手段
２０３　公開鍵生成手段
２０４　秘密鍵生成手段
２０５　受信側初期値変更制御手段

２０６　逆元算出手段
２０８　受信側乱数種生成手段
３００　伝送路
４００、４００Ｄ　乱数種生成手段

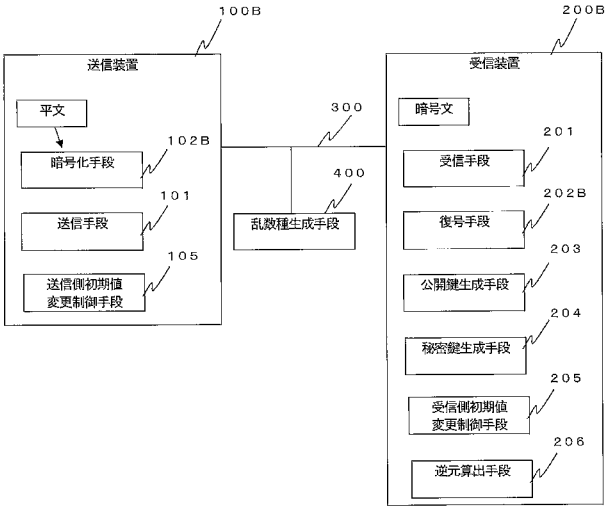
【図１】



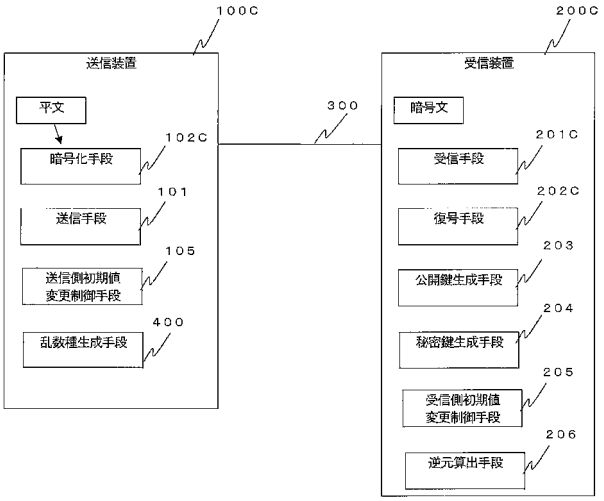
【図１Ａ】



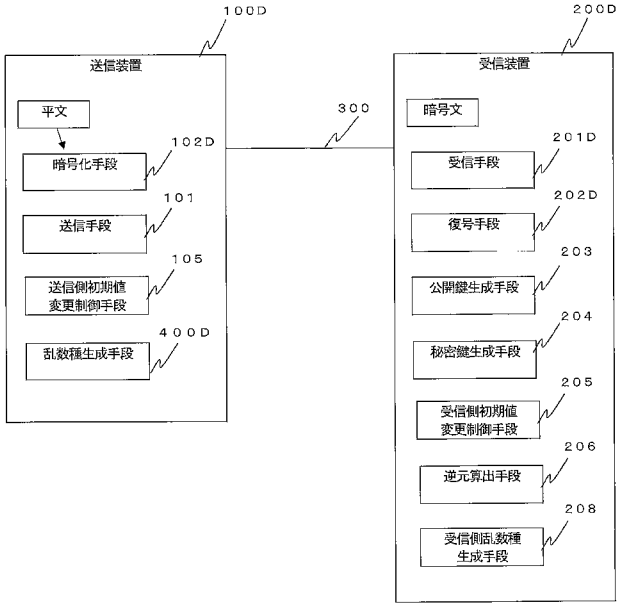
【図１Ｂ】



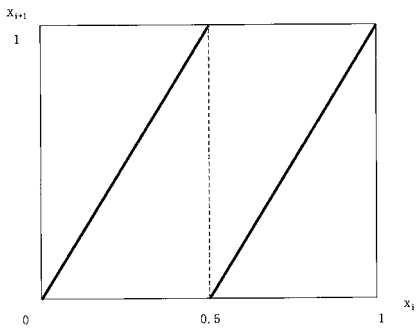
【図１Ｃ】



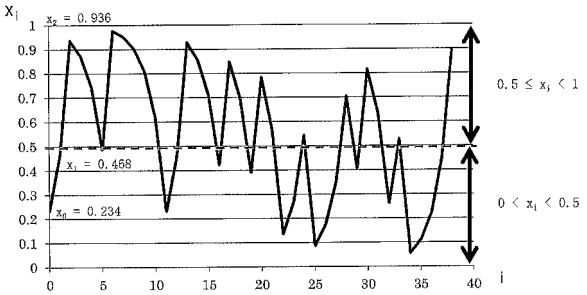
【図１Ｄ】



【図２】



【図３】



【図４】

(A)

▽ベルヌーイシフト写像

$X_{i+1} = 2X_i$ or $2X_i-1$

$X_0 = 1/11$
$X_1 = 2/11$
$X_2 = 4/11$
$X_3 = 8/11$
$X_4 = 16/11 - 11/11 = 5/11$
$X_5 = 10/11$
$X_6 = 20/11 - 11/11 = 9/11$
$X_7 = 18/11 - 11/11 = 7/11$
$X_8 = 14/11 - 11/11 = 3/11$
$X_9 = 6/11$
$X_{10} = 12/11 - 11/11 = 1/11$

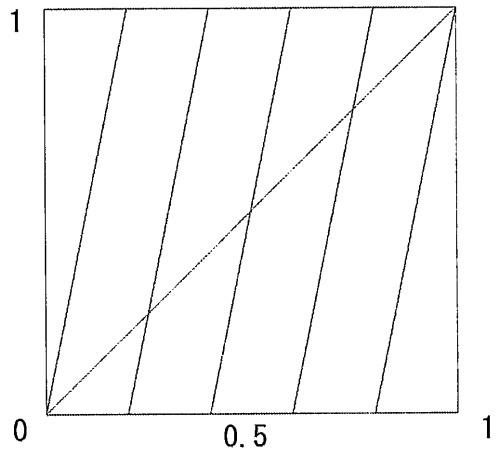
(B)

▽合同算術

$2^i \bmod 11$

$X_0 = 2^0 \equiv 1 \pmod{11}$
$X_1 = 2^1 \equiv 2 \pmod{11}$
$X_2 = 2^2 \equiv 4 \pmod{11}$
$X_3 = 2^3 \equiv 8 \pmod{11}$
$X_4 = 2^4 \equiv 2\times8 \equiv 5 \pmod{11}$
$X_5 = 2^5 \equiv 2\times5 \equiv 10 \pmod{11}$
$X_6 = 2^6 \equiv 2\times10 \equiv 9 \pmod{11}$
$X_7 = 2^7 \equiv 2\times9 \equiv 7 \pmod{11}$
$X_8 = 2^8 \equiv 2\times7 \equiv 3 \pmod{11}$
$X_9 = 2^9 \equiv 2\times3 \equiv 6 \pmod{11}$
$X_{10} = 2^9 \equiv 2\times6 \equiv 1 \pmod{11}$

【図５】



【図６】

(A)

▽ベルヌーイシフト写像

$X_{i+1} = 5X_i - M_0$

$X_0 = 1/11$
$X_1 = 5/11$
$X_2 = 25/11 - 22/11 = 3/11$
$X_3 = 15/11 - 11/11 = 4/11$
$X_4 = 20/11 - 11/11 = 9/11$
$X_5 = 45/11 - 44/11 = 1/11$

(B)

▽合同算術

$5^i \bmod 11$

$X_0 = 5^0 \equiv 1 \pmod{11}$
$X_1 = 5^1 \equiv 5 \pmod{11}$
$X_2 = 5^2 \equiv 5\times5 \equiv 3 \pmod{11}$
$X_3 = 5^3 \equiv 5\times3 \equiv 4 \pmod{11}$
$X_4 = 5^4 \equiv 5\times4 \equiv 9 \pmod{11}$
$X_5 = 5^5 \equiv 5\times9 \equiv 1 \pmod{11}$

【図７】

(A) ベルヌーイシフト写像

```
// BelnuiMap (係数、最大区間、反復回数)
int BelnuiMap( int A, int P, int ret ) {
        int interval[A];
        int i,j,x;

        // 各々の区間の境界値
        for(i=0;i<A;i++) {
                // 各区間をP倍に拡大し保存
                interval[i] = i * P;
        }

        // 写像の反復(ret回)
        x = 1 * A; // 初期値X0=1⇒A倍に拡大
        for(i=0;i<ret;i++) {
                printf("%02d¥tx-%d¥n",i,x/A);

                // 区間の検索（降順）
                j = A - 1;
                while( interval[j] > x ) {
                        j--;
                }
                // ベルヌーイシフト写像の演算
                x = x - interval[j];
                x = A * x;
        }

        return x / A;
}
```

(B) 合同算術

```
// modulo (係数、法、乗数)
int modulo(int A, int P, int r) {

        int i, remain;

        remain = 1;

        // A^0からr乗までの余り
        for(i=0;i<r;i++) {
                printf("%02d¥tmod=%d¥n",i,remain);

                // べき乗
                remain = A * remain;
                // 余り
                remain = remain % P;
        }

        return remain;
}
```

剰余算が1回の引き算で行えるようになる

【図８】

(A)

i	X_i
00	x=1
01	x=5
02	x=3
03	x=4
04	x=9
05	x=1
06	x=5
07	x=3
08	x=4
09	x=9
10	x=1

(B)

i	$5^i \pmod{11}$
00	mod=1
01	mod=5
02	mod=3
03	mod=4
04	mod=9
05	mod=1
06	mod=5
07	mod=3
08	mod=4
09	mod=9
10	mod=1

【図９】

一次元不定方程式の整数解を利用し
$7x+60y=1$
を拡張したユークリッドの互除法を使いｘとｙを求める

STEP1

$60 = 7\times8+4$
$7 = 4\times1+3$
$4 = 3\times1+1$
~~$3 = 1\times3+0$~~

余りが１となるまで
ユークリッドの互除法
を行う

STEP2

$4 = 60-7\times8$
$3 = 7-4\times1$
$1 = 4-3\times1$

余りを左辺に移項し
式を変形する

STEP3

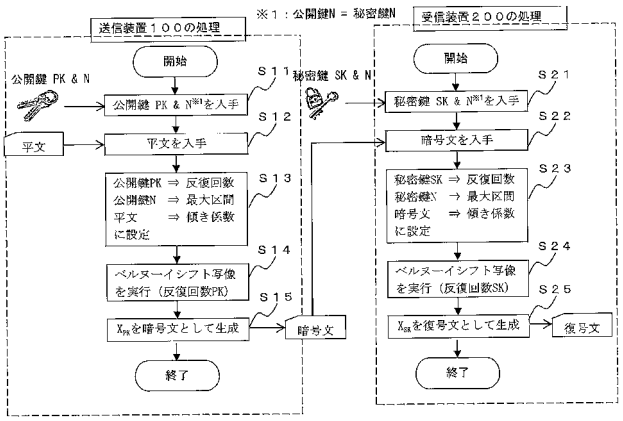
$1 = 4-3\times1$

$1 = 4-(7-4\times1)\times1$
$1 = 7\times(-1)+4\times2$ ※式を整理

$1 = 7\times(-1)+(60-7\times8)\times2$
$1 = 7\times(-17)+60\times2$ ※式を整理

STEP2の下の式から矢印で
示したように値を代入して
式を整理しながら
$1=7x+60y$
の形式に整理する
※下線の数値をくくりだすようにする

【図１０】



【図１１】

(A)

	平文		
	5	19	101
i	Xi	Xi	Xi
0	1	1	1
1	5	19	101
2	25	75	48
3	125	138	129
4	53	48	16
5	122	54	43
6	38	25	53
7	47	46	62

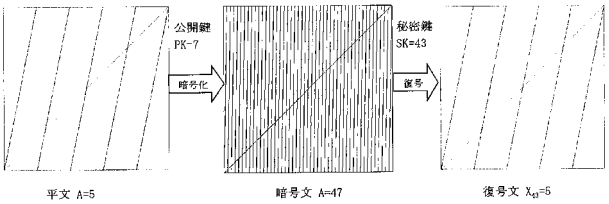
傾きA=5のベルヌーイシフト写像を
PK=7回反復することでX_7=47を得て
これを暗号文とする

傾きA=47のベルヌーイシフト写像を
SK=43回反復することでX_{43}=5を得て
これを復号文とする

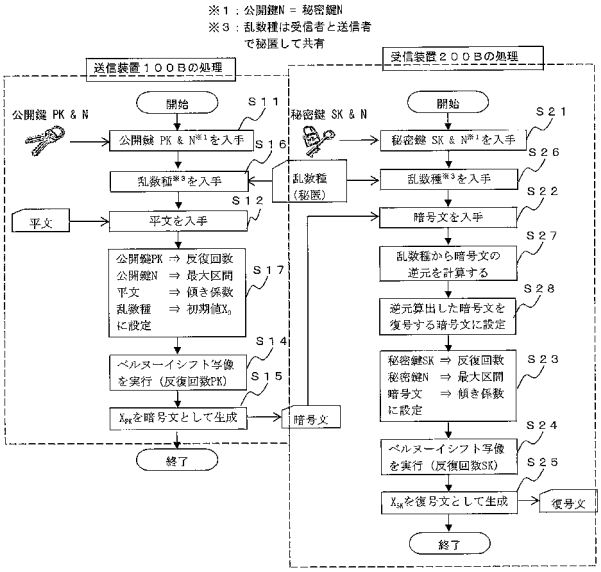
(B)

	暗号文		
	47	46	62
i	Xi	Xi	Xi
0	1	1	1
1	47	46	62
2	64	114	126
3	5	96	90
4	92	126	3
5	34	76	43
6	25	64	92
7	31	84	127
8	27	3	9
9	125	138	129
10	12	56	133
11	135	2	95
12	53	92	27
13	60	85	101
14	103	49	113
15	122	109	142
16	14	9	81
17	86	128	17
18	38	25	53
19	70	6	140
20	1	133	100
21	47	112	51
22	64	4	16
23	5	41	134
24	92	27	14
25	34	98	10
26	25	75	48
27	31	18	116
28	27	113	42
29	125	50	30
30	12	12	1
31	135	123	62
32	53	81	126
33	60	8	90
34	103	82	3
35	122	54	43
36	14	53	92
37	86	7	127
38	38	36	9
39	70	83	129
40	1	100	133
41	47	24	95
42	64	103	27
43	5	19	101

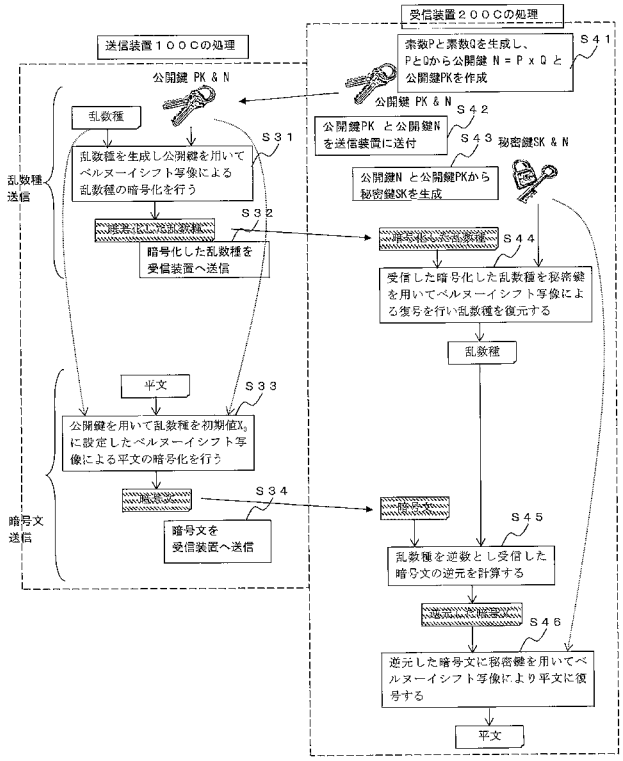
【図１２】



【図１３】



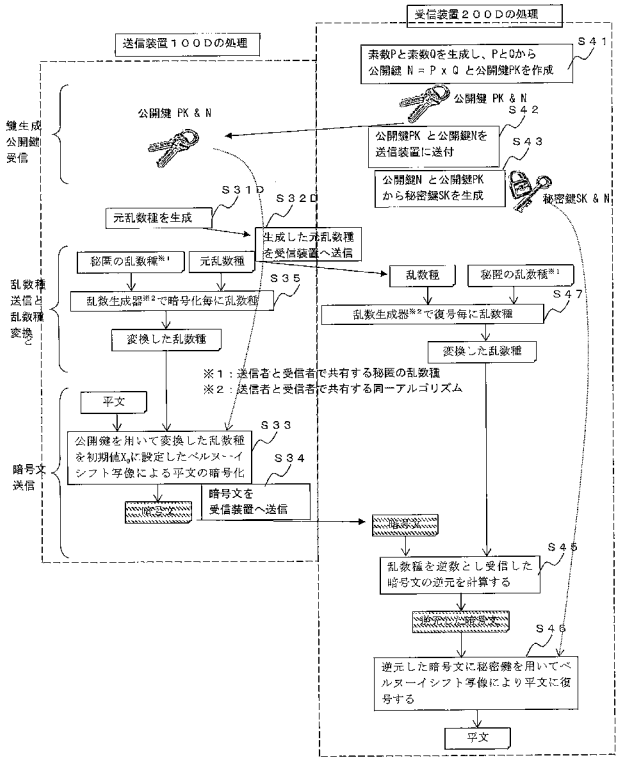
【図１３Ａ】



【図１４】

	平文		
	5	5	5
i	Xi	Xi	Xi
0	1	3	67
1	5	15	49
2	25	75	102
3	125	89	81
4	53	16	119
5	122	80	23
6	38	114	115
7	47	141	3

【図１５】


受信装置２００Ｄの処理
送信装置１００Ｄの処理
鍵生成
公開鍵
受信
乱数種
送信と
乱数種
変換
暗号文
送信
素数Pと素数Qを生成し、PとQから
公開鍵 N = P x Q と公開鍵PKを作成
S41
公開鍵 PK & N
公開鍵 PK & N
S42
公開鍵PK と公開鍵Nを
送信装置に送付
S43
公開鍵N と公開鍵PK
から秘密鍵SKを生成
秘密鍵SK & N
S31D
S32D
元乱数種を生成
生成した元乱数種
を受信装置へ送信
秘匿の乱数種※1
元乱数種
乱数種
秘匿の乱数種※1
S35
乱数生成器※2で暗号化毎に乱数種
S47
乱数生成器※2で復号毎に乱数種
変換した乱数種
変換した乱数種
※１：送信者と受信者で共有する秘匿の乱数種
※２：送信者と受信者で共有する同一アルゴリズム
平文
S33
公開鍵を用いて変換した乱数種
を初期値X0に設定したベルヌーイ
シフト写像による平文の暗号化
S34
暗号文
暗号文を
受信装置へ送信
暗号文
S45
乱数種を逆数とし受信した
暗号文の逆元を計算する
S46
逆元した暗号文に秘密鍵を用いてベ
ルヌーイシフト写像により平文に復
号する
平文