

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6621813号
(P6621813)

(45) 発行日 令和1年12月18日 (2019. 12. 18)

(24) 登録日 令和1年11月29日 (2019. 11. 29)

(51) Int. Cl. F I
G 0 9 C 1 / 0 0 (2006. 01) G 0 9 C 1 / 0 0 6 5 0 A

請求項の数 18 (全 32 頁)

(21) 出願番号	特願2017-516656 (P2017-516656)	(73) 特許権者	590000248
(86) (22) 出願日	平成27年9月30日 (2015. 9. 30)		コーニンクレッカ フィリップス エヌ ヴェ
(65) 公表番号	特表2017-533458 (P2017-533458A)		KONINKLIJKE PHILIPS N. V.
(43) 公表日	平成29年11月9日 (2017. 11. 9)		オランダ国 5656 アーエー アイン ドーフエン ハイテック キャンパス 5
(86) 国際出願番号	PCT/EP2015/072635		High Tech Campus 5, NL-5656 AE Eindhoven
(87) 国際公開番号	W02016/050884		
(87) 国際公開日	平成28年4月7日 (2016. 4. 7)	(74) 代理人	100107766
審査請求日	平成30年9月26日 (2018. 9. 26)		弁理士 伊東 忠重
(31) 優先権主張番号	14186951.1	(74) 代理人	100070150
(32) 優先日	平成26年9月30日 (2014. 9. 30)		弁理士 伊東 忠彦
(33) 優先権主張国・地域又は機関	欧州特許庁 (EP)		

最終頁に続く

(54) 【発明の名称】 難読化された算術を実行するための電子計算装置

(57) 【特許請求の範囲】

【請求項 1】

可換環において難読化された算術を実行するための電子計算装置であって、前記環は有限個の環要素をもち、環要素に対して環加算および環乗算が定義されており、当該計算装置は、環要素をエンコードする整数リストに対して作用し、前記整数リストは少なくとも二つ整数を有し、整数リストは、環要素を、該環要素が一つまたは複数の基本環要素の冪乗の線形結合に等しくなるようエンコードし、前記冪乗は該整数リストによって決定される指数をもち、当該計算装置は、

- ・増分環要素について定義された増分テーブルを記憶するよう構成された記憶部を有しており、

- ・前記増分テーブルは、出力環要素が増分環要素が入力環要素に環加算されたものに等しくなるよう入力環要素を出力環要素をエンコードする出力整数リストにマッピングし、当該計算装置はさらに、環加算ユニットを有しており、前記環加算ユニットは、

- ・第一の加算入力環要素をエンコードする第一の加算入力整数リストおよび第二の加算入力環要素をエンコードする第二の加算入力整数リストを受領する段階であって、前記増分環要素は前記第一および第二の加算入力環要素から独立である、段階と、

- ・前記第一および第二の加算入力整数リストから決定される環要素に前記増分テーブルを適用することによって加算出力環要素をエンコードする加算出力整数リストを決定する段階であって、前記加算出力環要素は前記第一の加算入力環要素および前記第二の加算入力環要素の環加算に等しい、段階とを有するよう構成されており、

10

20

前記加算出力整数リストを決定することが、
・一つまたは複数の基本元の冪乗の線形結合である環要素への前記増分テーブルの第一の適用によって、中間加算環要素をエンコードする中間加算整数リストを決定する段階であって、前記冪乗は前記第一および第二の加算入力整数リストから決定される、段階と、
・前記加算出力整数リストを決定する段階であって、前記中間加算整数リストから決定されるおよび前記第二の加算入力整数リストから決定される環要素への前記増分テーブルの第二の適用を含む、段階とを含む、
計算装置。

【請求項 2】

環乗算ユニットをさらに有しており、前記環乗算ユニットは、
・第一の乗算入力環要素をエンコードする第一の乗算入力整数リストおよび第二の乗算入力環要素をエンコードする第二の乗算入力整数リストを受領する段階と、
・前記第一および第二の乗算入力整数リストから決定される環要素に前記増分テーブルを適用することによって乗算出力環要素をエンコードする乗算出力整数リストを決定する段階であって、前記乗算出力環要素は前記第一の乗算入力環要素および前記第二の乗算入力環要素の環乗算に等しい、段階とを実行するよう構成されている、
請求項 1 記載の計算装置。

【請求項 3】

前記整数リストが環要素をエンコードする仕方は、
・前記環要素が、基本元を前記整数リストの第一の整数によって決定される冪乗にしたものから、前記基本元を前記整数リストの第二の整数によって決定される冪乗にしたものを引いたものに任意的に定数を乗算したものに等しくなるようにする、または
・前記環要素が、前記基本元を前記整数リストの第一の整数によって決定される冪乗にしたものに、前記基本元を前記整数リストの第二の整数によって決定される冪乗にしたものを足したものに任意的に定数を乗算したものに等しくなるようにする、または
・前記環要素が、前記基本元を前記整数リストの第一の整数によって決定される冪乗にしたもの、かける、前記基本元を前記整数リストの第二の整数によって決定される冪乗にしたものから、前記基本元を前記整数リストの第二の整数のマイナスによって決定される冪乗にしたものを引いたものに任意的に定数を乗算したものに等しくなるようにする、または
・前記環要素が、前記基本元を前記整数リストの第一の整数および第二の整数の第一の線形結合である冪乗にしたもの、足すまたは引く、前記基本元を前記第一の整数および前記第二の整数の第二の線形結合である冪乗にしたものに任意的に定数を乗算したものに等しくなるようにするものである、
請求項 1 または 2 記載の計算装置。

【請求項 4】

前記中間加算整数リストを決定することがさらに、前記第一および第二の加算入力整数リストから決定された整数を、第一の適用から帰結する整数リスト中の整数に加算することを含む、請求項 1 記載の計算装置。

【請求項 5】

・前記増分テーブルは、前記一つまたは複数の基本環要素を前記第一の整数リストの第一の整数から前記第二の整数リストの第一の整数を引いたもの乗したもの、足すまたは引く、ある基本環要素を前記第一の整数リストの第二の整数から前記第二の整数リストの第一の整数を引いたもの乗したもの、によって形成される環要素に適用される、および／または
・前記増分テーブルは、前記一つまたは複数の基本環要素を前記第一の整数リストの前記第一の整数から前記第一の整数リストの第二の整数を引いたもの乗したもの、足すまたは引く、ある基本環要素を前記第二の整数リストの第一の整数から前記第一の整数リストの前記第二の整数を引いたもの乗したもの、によって形成される環要素に適用される、および／または

・前記増分テーブルは、前記一つまたは複数の基本環要素を前記第一の整数リストの第二の整数から前記第一の整数リストの第一の整数を引いたもの乗したもの、足すまたは引く、ある基本環要素を前記第二の整数リストの第一の整数から前記第一の整数リストの前記第一の整数を引いたもの乗したもの、によって形成される環要素に適用される、請求項 1 ないし 4 のうちいずれか一項記載の計算装置。

【請求項 6】

・前記増分テーブルの前記第二の適用の前に、前記中間加算整数リストによって表わされる前記中間加算環要素が否定にされる、請求項 1 ないし 4 のうちいずれか一項記載の計算装置。

【請求項 7】

・整数リストによって表わされる環要素が、該整数リストを並べ替えることによって否定にされる、または
 ・整数リストによって表わされる環要素が、該整数リストの各整数に定数を加えるることによって否定にされる、または
 ・整数リストによって表わされる環要素が、該整数リストを並べ替え、該整数リストの一つまたは複数の整数に定数を乗算することによって否定にされる、請求項 1 ないし 6 のうちいずれか一項記載の計算装置。

【請求項 8】

前記増分テーブルが、前記入力環要素を表わす入力整数リストを入力として取る、請求項 1 ないし 7 のうちいずれか一項記載の計算装置。

【請求項 9】

乗算出力整数リストを決定することが、
 ・前記第一および第二の乗算入力整数リストから、それぞれ第一および第二の中間乗算要素をエンコードする第一の中間乗算整数リストおよび第二の中間乗算整数リストを決定し、
 ・前記第一および第二の中間乗算整数リストを前記環加算ユニットを通じて加算することを含む、請求項 2 記載の計算装置。

【請求項 10】

・前記第一の中間乗算整数リストの第一の整数は、前記第一の乗算入力整数リストの第一の整数、足す、前記第二の乗算入力整数リストの第一の整数であり、
 ・前記第一の中間乗算整数リストの第二の整数は、前記第一の乗算入力整数リストの第一の整数、足す、前記第二の乗算入力整数リストの第二の整数であり、
 ・前記第二の中間乗算整数リストの第一の整数は、前記第一の乗算入力整数リストの第二の整数、足す、前記第二の乗算入力整数リストの第二の整数であり、
 ・前記第二の中間乗算整数リストの第二の整数は、前記第一の乗算入力整数リストの第二の整数、足す、前記第二の乗算入力整数リストの第一の整数である、請求項 9 記載の計算装置。

【請求項 11】

・前記可換環が、ある整数を法とする整数によって形成される環である、または
 ・前記可換環が、ある整数多項式を法とする整数多項式によって形成される環である、請求項 1 ないし 10 のうちいずれか一項記載の計算装置。

【請求項 12】

請求項 1 ないし 11 のうちいずれか一項記載の計算装置と、環エンコード装置とを有するシステムであって、前記環エンコード装置は、
 ・一つまたは複数の基本環要素について定義されたエンコード・テーブルを記憶するよう構成された記憶部を有しており、前記エンコード・テーブルは、環要素を、環要素が前記一つまたは複数の基本環要素の冪乗の線形結合に等しいように整数リストにマッピングするものであり、前記冪乗は前記整数リストによって決定される指数をもち、前記環エンコード装置は、可換環の環要素を整数リストとしてエンコードするよう構成さ

10

20

30

40

50

れており、

前記計算装置は、コンピュータ・ネットワークを通じて、エンコードされた情報を受領するよう構成されている、

システム。

【請求項 1 3】

請求項 1 ないし 1 1 のうちいずれか一項記載の計算装置と、環デコード装置とを有するシステムであって、前記環デコード装置は、

- ・コンピュータ・ネットワークを通じて、前記計算装置から受領された整数リストを可換環の環要素にデコードし、

- ・一つまたは複数の基本環要素について、前記環要素を、前記環要素が一つまたは複数の基本環要素の冪乗の線形結合に等しいように決定するよう構成されており、前記冪乗は前記整数リストによって決定される指数をもつ、

システム。

【請求項 1 4】

可換環において難読化された算術を実行するための計算装置において使うための増分テーブルを計算するテーブル計算装置であって、前記環は有限個の環要素をもち、環要素に対して環加算および環乗算が定義されており、前記計算装置は、環要素をエンコードする整数リストに対して作用し、前記整数リストは少なくとも二つ整数を有し、当該テーブル計算装置は、

- ・増分テーブルを構築するよう構成されたテーブル生成ユニットを有し、前記テーブル生成ユニットは、

- ・入力環要素を反復的に選択し、

- ・固定した増分環要素が前記入力環要素に環加算されたものに等しい出力環要素を決定し、

- ・前記出力環要素についてエンコードする出力整数リストを決定し、

- ・前記入力環要素を前記出力整数リストにマッピングするエントリーを増分テーブルに追加するよう構成されており、

当該テーブル計算装置は、構築された増分テーブルを前記計算装置内に記憶するよう構成されている、

テーブル計算装置。

【請求項 1 5】

可換環において難読化された算術を実行するための、計算装置によって実行される電子的な計算方法であって、前記環は有限個の環要素をもち、環要素に対して環加算および環乗算が定義されており、当該計算方法は、環要素をエンコードする整数リストに対して作用し、前記整数リストは少なくとも二つ整数を有し、整数リストは、環要素を、該環要素が一つまたは複数の基本環要素の冪乗の線形結合に等しくなるようエンコードし、前記冪乗は該整数リストによって決定される指数をもち、

前記計算装置の記憶部は、増分環要素について定義された増分テーブルを記憶しており、前記増分テーブルは、出力環要素が固定した増分環要素が入力環要素に環加算されたものに等しくなるよう入力環要素を出力環要素をエンコードする出力整数リストにマッピングするものであり、

当該計算方法は、前記計算装置の環加算ユニットによって環加算する段階を含み、前記環加算は、

- ・前記環加算ユニットが、第一の加算入力環要素をエンコードする第一の加算入力整数リストおよび第二の加算入力環要素をエンコードする第二の加算入力整数リストを受領し、

- ・前記環加算ユニットが、前記第一および第二の加算入力整数リストから決定される環要素に前記記憶部に記憶されている前記増分テーブルを適用することによって加算出力環要素をエンコードする加算出力整数リストを決定することを含み、前記加算出力環要素は前記第一の加算入力環要素および前記第二の加算入力環要素の環加算に等しく、

前記加算出力整数リストを決定することが、
・一つまたは複数の基本元の冪乗の線形結合である環要素への前記増分テーブルの第一の適用によって、中間加算環要素をエンコードする中間加算整数リストを決定する段階であって、前記冪乗は前記第一および第二の加算入力整数リストから決定される、段階と、
・前記加算出力整数リストを決定する段階であって、前記中間加算整数リストから決定されるおよび前記第二の加算入力整数リストから決定される環要素への前記増分テーブルの第二の適用を含む、段階とを含む、
電子的な計算方法。

【請求項 16】

当該計算方法がさらに、前記計算装置の環乗算ユニットによって環乗算する段階を含み、前記環乗算は、

・前記環乗算ユニットが、第一の乗算入力環要素をエンコードする第一の乗算入力整数リストおよび第二の乗算入力環要素をエンコードする第二の乗算入力整数リストを受領し、

・前記環乗算ユニットが、前記第一および第二の乗算入力整数リストから決定される環要素に前記記憶部に記憶されている前記増分テーブルを適用することによって乗算出力環要素をエンコードする乗算出力整数リストを決定することを含み、前記乗算出力環要素は前記第一の乗算入力環要素および前記第二の乗算入力環要素の環乗算に等しい、
請求項 15 記載の電子的な計算方法。

【請求項 17】

プログラム可能な装置で実行されたときに請求項 15 または 16 記載の方法を実行するよう構成されているコンピュータ・プログラム命令を有するコンピュータ・プログラム。

【請求項 18】

・請求項 17 記載のコンピュータ・プログラムを有する、
コンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は電子計算装置、環エンコード装置、環デコード装置、テーブル計算装置、電子計算方法、コンピュータ・プログラムおよびコンピュータ可読媒体に関する。

【背景技術】

【0002】

ホワイトボックス暗号法およびより一般にはソフトウェア難読化において、計算はしばしば、そのままの値ではなくエンコードされた値に対して実行される。計算が平文値自身ではなくエンコードされた値に対して実行される場合、難読化されたソフトウェアのリバースエンジニアリングはより難しくなる。

【0003】

エンコード後は、加算または乗算のような通常の演算はもはや、コンピュータの組み込みプリミティブを使って実行することはできない。エンコードされた値のストレートな加算は通常は、値の和のエンコードにはならない。同じことは乗算についてもいえる。公式でいうと、たいていの x および y について、 $E(x) + E(y) \neq E(x+y)$ となる。ここで、 E はエンコード関数を表わす。

【0004】

この問題への解決策は、加算 (A) および乗算 (M) テーブルを導入することである。それらのテーブルは二つのエンコードされた値を入力として取り、加算演算または乗算演算のエンコードに対応するエンコードされた値を出力として生成する。それらのテーブルは： $A(E(x), E(y)) = E(x+y)$ ； $M(E(x), E(y)) = E(xy)$ として定義されてもよい。これらのテーブルは、エンコードされた値に対して直接、算術が実行されることを許容する。

【発明の概要】

【発明が解決しようとする課題】

10

20

30

40

50

【0005】

テーブルを使った難読化された加算および乗算には少なくとも二つの欠点がある。第一に、テーブルがきわめて大きなものとなることがある。 x および y が $l_{\text{セル}}$ ビットとして表わされる場合、各テーブルは $2^{2l} \cdot l$ ビットを必要とする。

【0006】

第二に、そのようなテーブルは簡単にソフトウェアにおいて見出されうる。より悪いことに、それらのテーブルは、エンコードされていたとしても、加算または乗算演算として識別されうる。これはたとえばエンコードにおいて保存されるこれらの関数の特性を通じてである。たとえば、乗算テーブルは $M(E(0), E(x)) = E(0)$ を満たす。攻撃者はこのことおよび同様の特性を使ってテーブルがどの演算を表わしているかを推測しうる。

10

【課題を解決するための手段】

【0007】

難読化された算術を実行するための改良された仕方をもつことが有利であろう。請求項において定義されるような計算装置が提供される。

【0008】

本発明者らは、場合によっては、エンコードされた値に対する乗算および加算が、複数の値を単一のエンコードされた値にエンコードする必要なく、単一のテーブルを使って実行されうることを見出した。加算および乗算について同じテーブルが使用されるので、リバースエンジニアリングの際に加算または乗算のどちらが実行されるのかを見ることは困難になる。加算および乗算は外部から見たときは同じ演算に見えるので、本発明者らはこの方法を「同質難読化 (homogenous obfuscation)」と名付けた。たとえ攻撃者が使用されているテーブルを見つけることができたとしても、またどうにかしてその機能を増分テーブルとして割り出すことができたとしても、いまだ、加算または乗算演算のどちらが実行されるかを知ることができない。テーブルが整数リストの要素に対して作用する仕方は、加算と乗算とで異なるが、これは、コード難読化、ホワイトボックス実装などといった伝統的な難読化を使って容易に隠すことができる。

20

【0009】

加えて、使用される単一テーブルは背景技術において論じたものより小さい。約 $2^l \cdot l$ ビットが必要とされる。たとえ加算のみが使用されるとしても、難読化された加算のために必要とされるテーブルは背景技術において提案されているテーブルより小さい。

30

【0010】

本発明は、多くの異なる可換環 R に適用される。ただし、あらゆる環が整数リストとしてのエンコードを許容するわけではない。可換環は、多くの異なるおなじみの数学的構造、たとえばある数を法とする整数 (Z_n) またはある数および多項式を法とする多項式 ($Z_n[x]/f(x)$) を含む数学的概念である。体は可換環の特別な場合である。本稿で記述されるように、当業者は所与の環が上記難読化を許容するかどうかを検証できる。

【0011】

たとえば、環要素〔環の元〕は二つの整数 (a, b) としてエンコードされうる。算術は、エンコードされた環要素を該エンコードされた環要素に増分値を加えたものにマップする増分テーブルを使って、該エンコードに対して直接実行されることができる。たとえば、テーブルは、 $u^c - u^d = u^a - u^b + 1$ である場合に (a, b) を (c, d) にマップしてもよい。加算および乗算はいずれも、該増分テーブルの反復適用によって実行される。

40

【0012】

本稿でより十全に論じられるように、他の多くの可能性および変形がある。典型的には、任意の所与の実装において多くの変形のうちのどれが選ばれたかは攻撃者には未知である。

【0013】

計算装置は電子装置であり、モバイル電子装置、たとえば携帯電話、セットトップボックス、コンピュータ、スマートカードなどであってもよい。

【0014】

50

本稿に記載される難読化された算術は、幅広い範囲の実際的なアプリケーションにおいて適用されうる。そのような実際的なアプリケーションは、リバースエンジニアリングが防止されるべき、プライベートなハードウェアで走る安全なアプリケーション、たとえばバンキング・アプリケーションなどを含む。他のアプリケーションは、不用意なデータ漏洩が防止されるべきアプリケーションを含む。プログラムがプライベートなデータをリリースするようだまされる場合、漏れたデータがエンコードされていれば、懸念の度合いは低下する。難読化された算術は、アプリケーションを走らせているサーバーにも適用されうる。ユーザーがエンコードされた形でデータを送受信すれば、プライバシーが高まる。

【0015】

本発明に基づく方法は、コンピュータ上でコンピュータ実装される方法として、あるいは専用ハードウェアにおいて、あるいは両者の組み合わせにおいて実装されうる。本発明に基づく方法のための実行可能コードまたはその一部はコンピュータ・プログラム・プロダクトに記憶されうる。コンピュータ・プログラム・プロダクトの例はメモリ・デバイス、光記憶デバイス、集積回路、サーバー、オンライン・ソフトウェアなどを含む。好ましくは、コンピュータ・プログラム・プロダクトは、前記プログラム・プロダクトがコンピュータ上で実行されるときに本発明に基づく方法を実行するためのコンピュータ可読媒体上に記憶された非一時的なプログラム・コード手段を有する。

【0016】

ある好ましい実施形態では、コンピュータ・プログラムは、該コンピュータ・プログラムがコンピュータ上で実行されるときに本発明に基づく方法のすべての段階を実行するよう適応されたコンピュータ・プログラム・コード手段を有する。好ましくは、コンピュータ・プログラムはコンピュータ可読媒体上で具現される。

【先行技術文献】

【非特許文献】

【0017】

【非特許文献1】Chow et al, "Whitebox cryptography and an AES implementation"

【図面の簡単な説明】

【0018】

本発明のこれらおよび他の側面は、以下に記載される実施形態から明白となり、これを参照することで明快にされるであろう。

【図1a】計算装置100のある実施形態の例を概略的に示す図である。

【図1b】環加算ユニット130のある実施形態の例を概略的に示す図である。

【図1c】環乗算ユニット140のある実施形態の例を概略的に示す図である。

【図2】計算装置101のある実施形態の例を概略的に示す図である。

【図3】計算装置において使うための増分テーブルを計算するためのテーブル計算装置200のある実施形態の例を概略的に示す図である。

【図4】難読化された算術を実行するための計算方法300のある実施形態の例を概略的に示す図である。

【図5】加算方法400のある実施形態の例を概略的に示す図である。

【図6】乗算方法500のある実施形態の例を概略的に示す図である。

【図7】aは、ある実施形態に基づくコンピュータ・プログラムを有する書き込み可能部分をもつコンピュータ可読媒体を示しており、bは、ある実施形態に基づくプロセッサ・システムの概略図である。異なる図面において同じ参照符号をもつ項目は同じ構造的特徴および同じ機能をもつまたは同じ信号である。そのような項目の機能および／または構造が説明された場合、詳細な記述におけるその反復説明は必要ない。

【発明を実施するための形態】

【0019】

本発明は多くの異なる形の実施形態が可能であるが、本開示は、本発明の原理を例解するものと考えられ、本発明を図示および記述される個別的な実施形態に限定することは意

10

20

30

40

50

図されていないとの理解のもとに、一つまたは複数の個別の実施形態について図面に示し、本稿で詳細に記述する。

【0020】

以下では、理解のために、実施形態の諸要素が動作において記述される。しかしながら、それぞれの要素がそれらによって実行されると記述されている機能を実行するよう構成されていることは明白であろう。

【0021】

電子計算装置は、驚くほど小さいテーブルを使って効率的な算術を実行する。さらに、難読化された算術の技術分野では、演算がテーブルを通じて実行できればそれは利点であると考えられる。そのような演算は、たとえば伝統的なホワイトボックス技法（たとえば非特許文献1参照）を使って、容易にさらに難読化されうるからである。よって、算術演算をテーブルを使って表現する必要がある。実施形態は、従来技術よりも小さなテーブルを使って加算を実装する。たとえホワイトボックス暗号法のような追加的な難読化がなくても、電子計算装置は難読化に寄与する。本稿で示されるように、エンコードおよび増分テーブルが実装されうる多くの仕方がある。何らかの特定の実施形態においてどのエンコードが使われているかは攻撃者には未知であり、よって観察される計算は解釈するのがより困難になる。

【0022】

実施形態は、乗算および加算演算が同じテーブルを使って実行されることを許容する。これはさらに、難読化に対する上乗せとなる。もはや、増分テーブルが使われるという事実から、どの演算が実行されるかを判別することはできないからである。下記では、まず、計算装置の実施形態のいくつかの可能なアーキテクチャーが論じられる。次に、難読化された算術を実行するためのいくつかの代替が論じられる。

【0023】

図1は、計算装置100のある実施形態の例を概略的に示している。計算装置100は、有限可換環において難読化された算術を実行するための電子装置である。可換環の多くの例が知られている。下記では二つのそのような環：ある数を法とする整数 (Z_n) およびある数および多項式を法とする多項式 ($Z_n[x]/f(x)$) について例が与えられる。他の実施形態は他の可換環を使ってもよい。

【0024】

環の要素は環要素〔環の元〕と称される。環要素に対して、加算と乗算が定義される。これらは環加算および環乗算と称される。

【0025】

環要素は、必要に応じていかなる好適な形で表現されてもよい。たとえば、 Z_n の要素は整数として表現されてもよく、 $Z_n[x]/f(x)$ の要素は多項式として表現されてもよい。しかしながら、計算装置100においては、環要素は整数リストとして表現される。たとえば、環要素 a は計算装置100において、リスト (a_1, a_2) として表現される。これは非整数環、たとえば多項式環についてさえも成り立つ。整数リストは環要素を、環要素と整数リストとの間の何らかのマッピングに従ってエンコードする。任意の環要素が与えられると、その環要素を表わす少なくとも一つの整数リストがあり、任意の整数リストを与えられると、それが表わすちょうど一つの環要素がある。諸実施形態において、任意の環要素が整数リストとして表現されうる。

【0026】

整数リストは少なくとも二つの要素をもつ。実のところ、加算および乗算演算は、整数リストが短いほど必要とするステップが少なくなる。よって、ある実施形態では、整数リストは常に二つの要素をもつ。メインの記述では、整数リストは整数対であると想定するが、三つ以上の要素をもつ整数リストの例も与えられる。一例として、 (a_1, a_2) は環要素

【0027】

【数 1】

$$(u^{a_1} - u^{a_2})$$

にマップしてもよい。ここで、uは、基本環要素〔基本の環の元〕(base ring element)と称される特別な環要素である。複数の基本元を使うものを含め多くの変形が下記で論じられる。しかしながら、メインの議論では、「例示的エンコード」として、所与の整数リスト(a_1, a_2)が環要素

【0 0 2 8】

【数 2】

$$(u^{a_1} - u^{a_2})$$

10

にマップすると想定する。

【0 0 2 9】

ある実施形態では、整数リスト中の整数は非負である。これは計算を単純化するが、必要ではない。さらに、ある実施形態では、整数リスト中の整数は、基本元の位数を法とされる。基本元uの位数は、 $u^k=1$ となる最小の整数kである。たとえばkを法とする演算を実行することによって整数リスト中の値を範囲 $[0, k-1]$ 内に保つことが便利である。

【0 0 3 0】

計算装置 1 0 0 はオペランド・ストア 1 5 0 を有していてもよい。オペランドは、オペランド・ストア 1 5 0 に整数リストとして記憶される。算術は、オペランド・ストア 1 5 0 に記憶されているオペランドに対して実行されてもよい。算術の結果はオペランド・ストア 1 5 0 に記憶されてもよい。ここで、算術の結果は新たな演算において使われてもよく、あるいは異なる装置に出力されるなどしてもよい。

20

【0 0 3 1】

計算装置 1 0 0 は、ある増分環要素 (increment ring element) について定義された増分テーブルTを記憶するよう構成されている記憶部 1 1 0 を有する。増分テーブルは入力環要素を、出力環要素をエンコードする出力整数リストにマップし、出力環要素は、増分環要素が入力環要素に環加算されたものに等しくなる。ある実施形態では、入力環要素は整数リストとして表現される。このようにして、テーブルTは整数リストを整数リストにマップする。いずれも同じエンコード、たとえば同じマッピングに基づいている。しかしながら、入力環要素が代替的なエンコードでの整数リストとして表現されている実施形態がある。いずれにせよ、入力環要素はデジタル形式で表現され、テーブルが入力環要素を出力環要素にマップすることを許容する。

30

【0 0 3 2】

テーブルは、何らかのフォーマットの入力環要素を、関連付けられた出力整数リストと一緒にリストしてもよい。テーブルは、入力環を省略して、出力整数リストのみをリストすることによって、記憶部において表現されてもよい。たとえば、これは、入力環が正準フォーマットで表現される場合になされてもよい。

【0 0 3 3】

たとえば、上記例示的エンコードを想定すると、入力環要素

40

【0 0 3 4】

【数 3】

$$k = u^{k_1} - u^{k_2}$$

はテーブルTによって出力整数リストにマッピングされてもよい。この場合、入力環要素は整数リストとして表現されてもよく、 $((k_1, k_2)) = (l_1, l_2)$ となりうる。この右辺は出力環要素

【0 0 3 5】

【数 4】

$$l = u^{l_1} - u^{l_2}$$

50

をエンコードする。出力環要素は増分環要素が入力環要素に環加算されたものに等しい。たとえば、増分環要素が1であれば、 $l=k+1$ である。ある実施形態では、増分要素は1であってもよいが、これは必須ではない。たとえば、上記例示的エンコードを使うと、増分要素は、 t の何らかの値、たとえば $0 \leq t < \text{位数}(u)$ となる任意の値についての u^t と選んでもよい。

【0036】

増分テーブルは、背景技術において述べたテーブルよりずっと小さい。背景技術のテーブルは二つの入力、たとえば二つのエンコードされた数を受けて、エンコードされた出力を生じる。しかしながら、テーブル T は、一つのエンコードされた出力を生じるために一つのエンコードされた入力を取るだけであり、増分環要素は固定されている。両者のエンコードが同様の量のスペースを取るとすると、 T の入力スペースはほぼ平方根に低減される。これはかなりのサイズ改善である。

10

【0037】

計算装置100は環加算ユニット130および環乗算ユニット140を有する。計算装置100は環否定ユニット120をも有していてもよい。ある実施形態では、環乗算ユニット140は、加算を実行するために加算ユニット130を使ってもよい；加算ユニット130は否定ユニット120を使ってもよい。これは、図1では、ユニット120、130、140の間の線によって示されている。しかしながら、ユニットは二重にあってもよく、たとえば、加算ユニット130が自分自身の否定を行なってもよく、乗算140が自分自身の加算を行なってもよい。否定(negation)は「符号の変化」とも称される。

20

【0038】

否定ユニット120は、否定入力環要素(negation-input ring element) a をエンコードする否定入力整数リスト(negation-input integer list) (a_1, a_2) を受領してもよい。否定ユニット120は、否定出力環要素(negation-output ring element) b をエンコードする否定出力整数リスト (b_1, b_2) を決定するよう構成されている。否定出力環要素は、否定入力環要素の符号を変えたものである。たとえば、否定出力環要素は、加法についての中立環要素(0)から否定入力環要素を引いたものに等しい。よって、 $b = -a$ である。

【0039】

ある実施形態では、否定ユニットは、否定入力整数リストを並べ替えることによって出力整数リストを計算してもよい。上記例示的エンコード

30

【0040】

【数5】

$$a = u^{a_1} - u^{a_2}$$

を使うと、出力整数リストは (a_2, a_1) であってもよい。並べ替えによる否定は、要素の読み出し元のアドレスを変えることによってコードにおいて効率的に実装でき、必ずしもメモリにおける実際の順序を変えるものではない。

【0041】

ある実施形態では、否定ユニットは、整数リストの各整数に定数を加算することによって出力整数リストを計算してもよい。たとえば、上記例示的エンコードにおいて $u^m = -1$ となるような整数 m を使うと、たとえば、出力整数リストは $(a_1 + m, a_2 + m)$ となりうる。

40

【0042】

環加算ユニット130は、第一の加算入力環要素をエンコードする第一の加算入力整数リスト (a_1, a_2) と第二の加算入力環要素をエンコードする第二の加算入力整数リスト (b_1, b_2) とを受領するよう構成される。たとえば、環加算ユニット130は、これら二つのオペランドをオペランド・ストア150から受領してもよい。環加算ユニット130は、第一および第二の加算入力整数リストから決定される環要素に増分テーブルを適用することによって、加算出力環要素をエンコードする加算出力整数リストを決定するよう構成される。加算出力環要素は、第一の加算入力環要素および第二の加算入力環要素の環加算に等しい。

50

【0043】

ある実施形態では、整数リストの特定の環要素へのマッピングは複数のサブマッピングを含み、各サブマッピングは整数リストの整数と関連付けられ、サブマッピングは整数を環要素にマッピングする。マッピングは、関連付けられた整数に適用されるサブマッピングの線形結合、例えば和である。サブマッピングは、基本元を、関連付けられた整数によって決定される冪乗にすることであってもよい。たとえば、上記の例示的エンコードにおいて、 (a_1, a_2) は、サブマッピング

【0044】

【数6】

$$u^{a_1} \text{ および } -u^{a_2}$$

10

の和であると言われてもよい。

【0045】

図1bは、加算ユニット130のある実施形態を示している。加算ユニット130は第一の加算入力整数リスト131および第二の加算入力整数リスト132を受領する。加算ユニット130は、第二の加算入力整数リスト132の整数から得られた環要素を、第一の加算入力環要素に逐次反復的に加算するよう構成された中間加算ユニット134を有する。たとえば、中間加算ユニット134は、第一の整数リスト要素に初期化される中間和133に対して加算してもよい。加算は、記憶部110からの増分テーブルの適用に関わる。

20

【0046】

環乗算ユニット140は、第一の乗算入力環要素をエンコードしている第一の乗算入力整数リスト (r_1, r_2) および第二の乗算入力環要素をエンコードしている第二の乗算入力整数リスト (s_1, s_2) を受領する。たとえば、乗算ユニット140は、第一および第二の乗算入力整数リストから決定される環要素に増分テーブルを適用することによって乗算出力環要素をエンコードしている乗算出力整数リストを決定するよう構成される。乗算出力環要素は第一の乗算入力環要素および第二の乗算入力環要素の環乗算に等しい。

【0047】

図1cは、乗算ユニット140のある可能な実施形態を示している。乗算ユニット140は第一の乗算入力整数リスト141および第二の乗算入力整数リスト142を受領する。乗算ユニット140は、第一および第二の乗算入力整数リスト141、142から、それぞれ第一および第二の中間乗算還要素をエンコードする第一の中間乗算整数リスト145 (t_1, t_2) および第二の中間乗算整数リスト146 (u_1, u_2) を決定するよう構成された中間乗算ユニット144を有する。乗算ユニット140は、第一145および第二の中間乗算整数リスト146を環加算ユニット130を通じて加算するよう構成される。中間整数リストを決定することは、整数リスト内の整数に対する算術演算に関わってもよいが、増分テーブルを要求しない。

30

【0048】

計算装置100は任意的には、可換環の環要素を整数リストとしてエンコードするための環エンコード・ユニット170と、整数リスト (a, b) を可換環の環要素 (x) にデコードするための環デコード・ユニット160とを有する。エンコード・ユニット170および／またはデコード・ユニット160は、たとえば計算装置100がエンコードされた入力を受け取るおよび／またはエンコードされた出力を報告するときは、不在であってもよい。エンコード・ユニット170および／またはデコード・ユニット160は、スタンドアローンのユニットとして、たとえばエンコード装置および／またはデコード装置160として実装されてもよい。

40

【0049】

環エンコード・ユニット170は、一つまたは複数の基本環要素 (u) について定義されたエンコード・テーブルを記憶するよう構成された記憶部172を有していてもよい。該エンコード・テーブルは、環要素 (x) を整数リスト $((a, b))$ にマッピングして、環要

50

素が前記一つまたは複数の基本環要素の冪乗の組み合わせに等しくなるようにする ($x=u^{a-u^b}$)。ここで、冪乗は、前記整数リストによって決定された指数をもつ。エンコード・ユニット 170 は、エンコードされた環要素をオペレーター・ストア 150 において記憶してもよい。エンコード・ユニット 170 はシステムが平文情報とともに機能することを許容する。

【0050】

環デコード・ユニット 160 は、一つまたは複数の基本環要素 (u) について、環要素 (x) を決定し、前記環要素が一つまたは複数の基本環要素の冪乗の線形結合に等しくなるようにする ($x=u^a-u^b$)。ここで、冪乗は、整数リストによって決定された指数をもつ。たとえば、デコード・ユニット 160 は、整数リストを環要素にマッピングするデコード・テーブルを記憶しているストアを有していてもよい。たとえば、デコード・ユニット 160 は、前記冪乗およびその線形結合を計算する計算ユニットを有していてもよい。

10

【0051】

多くの興味深い実施形態は、エンコードおよびデコード・ユニット 160 および 170 の一方または両方を省略する。たとえば、計算装置 100 は、エンコードされた情報を、コンピュータ・ネットワーク、たとえばインターネットを通じて受領するよう構成されてもよい。難読化された計算装置 100 が走るシステム、たとえば難読化された計算ソフトウェアを実行するコンピュータの権利者は、入力情報のためや、システム 100 によって出力される、たとえばコンピュータ・ネットワークを通じて送信し返される情報のために使われたエンコードを知らなくてもよい。よって、たとえ計算がクラウドで実行されるとしても、情報の所有者は、自分の情報が安全であるといういくらかの安心をもつ。エンコードされた形での情報に対する演算は典型的には暗号法、たとえば暗号化を使っては可能ではない。たとえ背景技術において概説されたようにテーブル・システムが使われるとしても、これは二つのテーブルを必要とする。

20

【0052】

典型的には、計算装置 100 は、装置 100 において記憶された適切なソフトウェアを実行するマイクロプロセッサを有する。たとえば、そのソフトウェアは、対応するメモリ、たとえば RAM のような揮発性メモリまたはフラッシュ (図示せず) のような不揮発性メモリ (図示せず) においてダウンロードされているおよび／または記憶されているのでもよい。あるいはまた、装置 100 は、全体的にまたは部分的に、プログラム可能な論理において、たとえばフィールドプログラマブルゲートアレイ (FPGA) として実装されてもよい。装置 100 は全体的または部分的に、いわゆる特定用途向け集積回路 (ASIC)、すなわち特定の用途のためにカスタマイズされる集積回路 (IC) として実装されてもよい。

30

【0053】

ある実施形態では、電子計算装置は、対応するユニットの機能を実行するよう構成された環加算回路および環乗算回路を有する。計算装置は、否定回路をも有していてもよい。該回路は、たとえば Verilog および VHDL のようなハードウェア記述言語において機能を記述することによって得られる、CMOS のような集積回路であってもよい。それらの回路は、プロセッサ回路および記憶回路であってもよく、プロセッサ回路が記憶回路に電子的に表現されている命令を実行してもよい。それらの回路も FPGA、ASIC などでありうる。

40

【0054】

テーブル・ストア 110 およびオペランド・ストア 150 は電子記憶、たとえばメモリとして実装されてもよい。両者は同じメモリの一部であってもよいが、相異なるメモリであってもよい。テーブル・ストア 110 は不揮発性の非書き込み可能なもの、たとえば ROM または一度書き込み複数読み出し (WORM) メモリであってもよい。オペランド・ストア 150 は揮発性または不揮発性の書き込み可能なメモリ、たとえばフラッシュまたは RAM であってもよい。

【0055】

図 2 は、計算装置 101 のある実施形態の例を概略的に示している。計算装置 101 は計算装置 100 の洗練である。ある実施形態では、計算装置 101 は複数環加算ユニット

50

、複数の環乗算ユニットおよび任意的には複数の否定ユニットを有する。たとえば、図2は、三つの乗算ユニット140. 1、140. 2および140. 3および二つの加算ユニット130. 1および130. 2を示している。これらのユニットは、それぞれユニット140および130と同じ設計であってもよい。乗算および加算ユニットは比較的小さなスペースを取る。たとえば、ソフトウェアで実装されるとき、これらのユニットは高々数百の低レベルのコンピュータ命令しか必要としない。特に、コンピュータ・プログラムにおいて必要とされるそれぞれの乗算または加算について、上記加算および／または乗算ユニットのコピーが使用されてもよい。このことは、伝統的な難読化技法を許容する。例として、図2は、難読化された算術を使って、多項式 ax^2+bx+c がどのようにして計算されるかを示している。

10

【0056】

複数の算術ユニット、たとえば加算、乗算、否定、の演算は、そのデータ依存性によって許容されるいかなる順序で順序付けられてもよい。たとえば、演算140. 3は順序140. 1、140. 2、130. 1および130. 2において130. 1より前のどの点に挿入されてもよい。さらに、その後の乗算または加算の順序は逆にされてもよい。こうして、図2のような図は、多くの仕方、ソフトウェア・プログラムのための線形な順序に変換されてもよい。これらのユニットが厳密に分離されることは必要ではない。第一のユニットのための命令は別のユニットのための命令の間に散りばめられてもよい。

【0057】

図3は、計算装置において使うための増分テーブルを計算するためのテーブル計算装置200のある実施形態の例を概略的に示している。増分テーブルは、計算装置100のような装置において使われてもよい。増分テーブルは、非一時的な記憶装置、たとえばハードディスク、不揮発性メモリ・チップなどに記憶されてもよい。

20

【0058】

テーブル計算装置200は、増分テーブルを構築するよう構成されたテーブル生成ユニット210を有する。たとえば、テーブル生成ユニットは下記を実行するよう構成されていてもよい。

- ・入力環要素、たとえば x を反復的に選択する。
- ・増分環要素が入力環要素に環加算されたものに等しい出力環要素を決定する。たとえば、増分値が1であれば $y=x+1$ 。
- ・出力環要素についてエンコードする出力整数リストを決定する。たとえば、テーブル計算装置200は、エンコード・ユニット170のようなエンコード・ユニットを有していてもよい。
- ・入力環要素を出力整数リストにマッピングする増分テーブルにエントリーを追加する。

30

【0059】

これらの段階は、すべての環の要素が整数リストにマッピングされ終わるまで実行されてもよい。いくつかの実施形態では、諸要素がスキップされて、部分的な増分テーブルを構築してもよい。たとえば、ある種の環要素が生起しないことがコンテキストから知られていてもよい。

【0060】

環 R 、潜在的な基本環要素 u 、エンコード、たとえば上記の例示的エンコードおよび整数リスト長、たとえば2を与えられて、デコード・テーブルが下記で与えるように生成されてもよい。 k は u の位数であるとする。

40

- ・あらゆる整数リストを生成する。これはたとえば、前記整数リスト長のすべての整数リストを生成し、リスト中の各位置について0から k の手前までのすべての整数を許容することによる。たとえば、 $(0,0), (0,1), (1,0), (1,1), (0,2), (1,2), (2,2), (2,0), (2,1), (0,3), \dots$ などを生成する。
- ・各生成された整数リストについて、該整数リストによってエンコードされる環要素を計算し、その整数リストをそのデコードに関連付けるエントリーをデコード・テーブルに追加する。

50

【0061】

デコードは、デコード・テーブルを使っても使わなくてもよいが、そのようなテーブルも有用である。デコード・テーブルから、たとえば該テーブルを環要素についてソートすることによって、エンコード・テーブルが生成されうるからである。環要素が複数のエンコードをもつことがありうる。たとえば、環要素0（加算についての中立元）は上記の例示的エンコードでは、任意の a について (a, a) と表現されうる。そのような複数のエンコードは、テーブルから除去されてもよい。これはたとえば、所与の環要素について、複数のエントリーのうちの一つを除いてすべてを削除することによる。あるいは複数のエンコードをテーブルに残して複数のエントリーのうちのランダムなものにエンコードするようエンコード・テーブルを使うことによる。

10

【0062】

デコードまたはエンコード・テーブルの構築は、環要素 u が基本環要素であるかどうかを見出すために使われてもよい。いくつかの環要素がエンコードをもたないことが判明したためエンコード・テーブルの構築が失敗したら、 u は基本環要素ではない。

【0063】

下記では、エンコード、増分テーブル、環加算方法および環乗算方法のいくつかの実施形態が呈示される。計算装置100の前記否定、加算および乗算ユニットはこれらの実施形態の任意のもののために構成されうる。すべての例は任意の可換環、特に Z_n および $Z_n[x]/f(x)$ に当てはまる。ここでは n は正の整数である。さらに、可換環の任意の元が、選ばれたエンコードにおいて表現されうるが大いに好ましい。すべての可換環が、所与のエンコードにおいて、たとえば所与の型の整数リスト表現としてすべての元が表現されることを許容するわけではない。可換環 R を与えられたとき、 R における任意の元が所与のエンコード型を使って整数リストとして表現されうる場合、完全な同質難読化を許容することにする。当業者は、エンコードが与えられたとき、所与の可換環が完全な同質難読化を許容するかどうかの検証を、たとえばあらゆる許容されるエンコードを生成し、それらが一緒になって所与の環のすべての元を表わすことを検証することによって、実行できる。いくつかの用途については、エンコードがいくつかのギャップをもつことが許容されてもよい。これは、結果として、それらのギャップでは、少なくともその難読化された整数リスト・エンコードを使ってでは、算術が実行できないことになりうる。特定の型のエンコードを許容する可換環の具体例はのちに呈示する。

20

30

【0064】

下記では、まず、例示的エンコードの記述が与えられる。多くの型のエンコードがあるが、環要素が整数のリストとして表現されうることは共通している。これらの整数は環要素ではない。たとえば、環が整数環でなく、たとえば多項式環であっても元は整数リストとして表現されうる。使用されるエンコード、どのようにに所与の整数リストが環要素にマッピングされるかは、エンコードと称される。典型的には、整数リストは常に同じ長さであるが、これは必須ではない。一般に、エンコードがより多くの型の整数リスト、たとえばより長いリストを許容するほど、所与の環要素が異なる仕方で整数リストとしてエンコードされうる可能性が高くなる。

【0065】

40

上記例示的エンコードをもつ可換環 R を与えられたとき、 R の任意の元 a が何らかの整数 a_1 および a_2 について $u^{a_1} - u^{a_2}$ と書ける特別な環要素 u がある。そのような特別な環要素を基本環要素と称することにする。すべての可換環がこのやり方でエンコードされうるのではないが、十分に多くの可換環について該エンコードは有用である。整数 a_1 および a_2 はそれ自身としては環 R の環の元ではない；これらは基本元の位数を法として演算対象となる整数である。環要素 a は基本元 u の冪乗、つまり u^{a_1} および u^{a_2} の線形結合に等しいことを注意しておく。この場合、線形結合はそれらの冪乗に $+1$ または -1 を乗算して足すことによって、より具体的には、第一の冪乗から第一の冪乗を引くことによって、得られる。計算装置は、上記の仕方でエンコードされた環要素に対して作用する。加算、否定および乗算ユニットはこのエンコードにおいて環要素に作用できる。

50

【0066】

増分テーブルTは加算および乗算演算両方において中心的な役割を演じる。増分テーブルは入力環要素をマッピングする。この場合、入力環要素は整数リストとして表現されていてもよい。たとえば、入力環要素 $k=u^{k_1}-u^{k_2}$ を表現する入力整数リスト (k_1, k_2) を与えられて、テーブルTはこれを、出力環要素 $l=u^{l_1}-u^{l_2}$ をエンコードする出力整数リストにマッピングする。たとえば、 $T((k_1, k_2))=(l_1, l_2)$ 。出力環要素は増分環要素が入力環要素に環加算されたものに等しい。この例では、増分要素は1、すなわち環乗算についての恒等元である環要素に取られてもよい。この場合、 $l=k+1$ である。テーブルが、同じエンコードを使う諸環要素に直接適用されうることが便利である。よって上記整数リスト表現をもつ諸環要素に適用されう。それにもかかわらず、テーブルが代替的なエンコード

10

【0067】

下記では、演算、否定、加算および乗算が記述される。

【0068】

否定。

否定入力環要素 $a=u^{a_1}-u^{a_2}$ を表わす否定入力整数リスト (a_1, a_2) を与えられて、該整数リストを並べ替えることによって、この場合は順序を逆にすることによって、否定出力整数リストが得られてもよい。否定出力整数リストは (a_2, a_1) であってもよい。多くの環Rについてそうであるように $u^m=-1$ となる m が存在するとすると、否定は代替的には、整数リストの各整数に定数、たとえば m を加算することによって得られてもよい。この場合、否定出力整数リストは (a_1+m, a_2+m) であってもよい。 $-a=u^{a_2}-u^{a_1}=u^{a_1+m}-u^{a_2+m}$ なので、これは成立する。整数リストにおける算術は好ましくは、基本元の位数を法として行なわれる。ここで、整数リストの整数は基本元の指数に対応する。よって、基本元の位数を法として同じである整数は同じ環要素をエンコードする。

20

【0069】

加算。

第一の加算入力環要素 $a=u^{a_1}-u^{a_2}$ をエンコードする受領された第一の加算入力整数リスト (a_1, a_2) および第二の加算入力環要素 $b=u^{b_1}-u^{b_2}$ をエンコードする第二の加算入力整数リスト (b_1, b_2) を加算するには、まず、中間加算環要素 c をエンコードする中間加算整数リスト $((c_1, c_2))$ が決定される。

30

【0070】

環要素 c は、第一の加算入力環要素 a に、基本元 u の、第二の加算入力整数リスト、特に第二の加算入力整数リストの最初の整数から決定される冪乗を加えたものであってもよい。この例では、 $c=u^{a_1}-u^{a_2}+u^{b_1}$ としてもよい。これを計算するには、 $c=u^{a_1}-u^{a_2}+u^{b_1}=(u^{a_1-b_1}-u^{a_2-b_1}+1)u^{b_1}$ であることを観察する。括弧内の項は、増分テーブルを使うエンコードにおいて書き直されてもよい。環要素 $u^{a_1-b_1}-u^{a_2-b_1}$ への増分テーブルの第一の適用を通じて、要素 $u^{d_1}-u^{d_2}=u^{a_1-b_1}-u^{a_2-b_1}+1$ が得られる。たとえば、 $T((a_1-b_1, a_2-b_1))=(d_1, d_2)$ による。すると、 $c_1=d_1+b_1$ および $c_2=d_2+b_1$ が得られる。このように、中間加算整数リスト $((c_1, c_2))$ を決定することはさらに、第二の加算入力整数リストから決定された整数を第一の適用から帰結する整数リストにおける整数に加算することを含んでいてもよい。 u^{b_1} を整数リスト表現におけるある環要素に、この場合は a に加えることは、時に、正の還元 (positive reduction) ステップと称される。

40

【0071】

このように、加算ユニットは、中間加算環要素 $c=u^{a_1}-u^{a_2}+u^{b_1}=u^{c_1}-u^{c_2}$ を整数リスト (c_1, c_2) として得た。中間加算環要素は、このように、一つまたは複数の基本元の冪乗の線形結合である。ここで、それらの冪乗は、第一および第二の加算入力整数リストから決定される。この場合、増分テーブルが適用されるのは、前記一つまたは複数の基本環要素 (u) を第一の整数リストの最初の整数 (a_1) から第二の整数リストの最初の整数 (b_1) を引いたもの乗したものから、前記基本環要素 (u) を第一の整数リストの第二の整数

50

(a_2) から第二の整数リストの最初の整数 (b_1) を引いたもの乗したものを引いたものによって形成される環要素 $u^{a_1-b_1}-u^{a_2-b_1}$ である。

【0072】

この例では、加算出力整数リストは、中間加算整数リストおよび第二の加算入力整数リストから決定された環要素への増分テーブルの第二の適用を通じて決定されてもよい。これは、中間加算環要素 c と、基本元を第二の加算入力整数リスト、たとえば第二の加算入力整数リストの第二の整数 b_2 から決定される冪乗にしたもののマイナスとの和を計算することを含んでいてもよい： $c-u^{b_2}=u^{c_1}-u^{c_2}-u^{b_2}$ 。これは、増分テーブルの第二の適用前に、中間加算整数リストによって表わされる中間加算環要素を否定にすることによって達成されてもよい。 c の否定は、上記のようになされてもよい。例として並べ替えを使うが、同じ動作は、指数に定数を加えることによって実行されてもよい。否定後は、前記和は、基本元を第二の加算入力整数リストから決定される冪乗にしたもののプラス（マイナスではなく）を使ってもよい： $-c+u^{b_2}=u^{c_2}-u^{c_1}+u^{b_2}$ 。この演算は上記と同じ型であり、 u^{b_1} を加えるのと同じ仕方でテーブル適用を通じて実行できる。この後は、結果が再び否定にされる。完全な加算は、二回の否定と同じ増分テーブル T の二回のテーブル適用を使ってもよい。

10

【0073】

整数リスト表現における環要素から、この場合 c から $-u^{b_2}$ を引くことは、時に負の還元 (negative reduction) ステップと称される。負の還元ステップは否定にして、上記正の還元ステップを実行し、再び否定にすることによって実行されてもよい。

20

【0074】

乗算。

第一の乗算入力環要素 $r=u^{r_1}-u^{r_2}$ をエンコードする受領された第一の乗算入力整数リスト (r_1, r_2) と第二の乗算入力環要素 $s=u^{s_1}-u^{s_2}$ をエンコードする第二の乗算入力整数リスト ((s_1, s_2)) とを乗算するために、第一の中間乗算整数リスト (t_1, t_2) および第二の中間乗算整数リスト (u_1, u_2) が決定される。乗算出力環要素をエンコードする乗算出力整数リストは、第一および第二の中間要素から決定される。他の実施形態では、二つより多くの中間乗算整数リストがあってもよい。 $r \cdot s = (u^{r_1}-u^{r_2})(u^{s_1}-u^{s_2}) = u^{r_1+s_1}-u^{r_1+s_2}+u^{r_2+s_2}-u^{r_2+s_1} = (u^{r_1+s_1}-u^{r_1+s_2}) + (u^{r_2+s_2}-u^{r_2+s_1}) = t+u$ となる。展開された積の項を二つの項 t および u に分割することは、異なる仕方で、たとえば $(u^{r_1+s_1}-u^{r_2+s_1}) + (u^{r_2+s_2}-u^{r_1+s_2})$ のようになされてもよい。

30

【0075】

このように、整数リストとして表現されている二つの環要素を乗算するために、それらの環要素は二つの新たな整数リストに変換されてもよく、該新たな変数リストを加算して乗算への答えを得ることができる。加算は上記のようになされてもよい。たとえば、乗算ユニットは中間整数リストを計算して、乗算ユニットに送ってもよい。

【0076】

たとえば、第一の中間乗算整数リストの第一の整数 t_1 は、第一の乗算入力整数リストの第一の整数 r_1 に第二の乗算入力整数リストの第一の整数 s_1 を加えたものであってもよく、第一の中間乗算整数リストの第二の整数 t_2 は、第一の乗算入力整数リストの第一の整数 r_1 に第二の乗算入力整数リストの第二の整数 s_2 を加えたものであってもよい： $t_1=r_1+s_1$ 、 $t_2=r_1+s_2$ ；第二の中間乗算整数リストの第一の整数 u_1 は、第一の乗算入力整数リストの第二の整数 r_2 に第二の乗算入力整数リストの第二の整数 s_2 を加えたものであってもよく、第二の中間乗算整数リストの第二の整数 u_2 は、第一の乗算入力整数リストの第二の整数 r_2 に第二の乗算入力整数リストの第一の整数 s_1 を加えたものであってもよい： $u_1=r_2+s_2$ 、 $u_2=r_2+s_1$ 。

40

【0077】

ある実施形態では、たとえばたった今開示した例では、算術は整数リストに対して実行され、環要素は何らかの自然な表現での環要素として計算される必要はない。ここで変形のいくつかを論じる。変形の多くは独立である。たとえば、変形エンコードが加算を実行

50

するための変形と組み合わせられてもよい。

【0078】

たとえば $u^{a^1-b^1}$ に対応する整数リストにおいて計算が実行されるときに難読化された算術を通じて、値は u の位数を法として約されてもよい。たとえば u の位数が30であれば、すべての計算は $\text{mod } 30$ で実行されてもよい。

【0079】

増分値。

増分値は1である必要はない。異なる増分値を使うには少なくとも二つの仕方がある。第一に、式 $c=u^{a^1}-u^{a^2}+u^{b^1}=(u^{a^1-b^1}-u^{a^2-b^1}+1)u^{b^1}$ は、 $c=u^{a^1}-u^{a^2}+u^{b^1}=(u^{a^1-b^1+t}-u^{a^2-b^1+t}+u^t)u^{b^1-t}$ に修正されてもよい。これは、値 u^t を加える増分テーブルが構築されうることを意味する。この増分テーブルは、整数 t が加えられるほかは同じ整数リストに適用される。増分テーブルの第一の適用後、 b_1 の代わりに数 b_1-t が加えられる。

10

【0080】

増分値を変更するもう一つの仕方は、環における g の反復加算が p を与えるよう R の二つの元 g および p を取るというものである。たとえば、 $h \cdot g = g + \dots + g = p$ となる整数 h がある。増分値 p 、たとえば $p=1$ または $p=u^t$ をもつ増分テーブル T_p があるとすると、 g を増分値として増分テーブル T_g が構築されうる。直接 T_p を適用するのと同じ効果を得るために、テーブル T_g が h 回適用されてもよい。異なる増分値をもつ異なる増分テーブルを使うことは、たとえば難読化を高めるために、単一の実施形態において組み合わせられてもよい。この構築は、後続の加算計算を変えることなく複数の増分値が組み合わせられうるという利点をもつ。

20

【0081】

増分テーブルの構築も変更されてもよい。たとえば、中間加算環要素についての式に戻るが、 $c=u^{a^1}-u^{a^2}+u^{b^1}=(u^{a^1-b^1}-u^{a^2-b^1}+1)u^{b^1}$ という因子分解の代わりに、 $c=u^{a^1}-u^{a^2}+u^{b^1}=(u^{a^1-a^2}+u^{b^1-a^2}-1)u^{a^2}$ ということが観察される。この公式を使って、増分値 -1 について増分テーブルが構築されることができる。この型の増分値は環要素 $u^{a^1-a^2}+u^{b^1-a^2}$ に適用される。この環要素は上記の例示的エンコードをもたない。にもかかわらず、この環要素は整数リスト、たとえば (a_1-a_2, b_1-a_2) として表現されることができる。よって、この増分テーブルは整数リストを入力として取り、整数リストを出力として生じる。しかしながら、先の例とは異なり、入力整数リストは出力エンコードとは異なるエンコードをもつ。さらに、加算ユニットへの入力において使われるエンコードはギャップをもたない、すなわち任意の環要素がこのエンコードにおいて表現されることがずっと好ましいものの、この増分テーブルのこの代替的な入力エンコードがギャップをもたない必要性はない。テーブル入力として表現される必要のあるすべての要素は、構築によって表現できる。

30

【0082】

増分テーブルをたとえば整数リスト (a_1-a_2, b_1-a_2) として表現されている環要素 $u^{a^1-a^2}+u^{b^1-a^2}$ に適用したのち、整数 a_2 が増分テーブルの出力の両方の要素に加算される。結果は、上記で定義した中間値 c である。第二のテーブル適用を実行するために、否定にし、この代替的な増分テーブルを使って u^{b^2} を加算し、再び否定にするという上記と同じ構築が使われてもよい。上記で示される構築を使って、増分値は -1 から他の値に変えられてもよい。

40

【0083】

増分テーブルを環要素 $u^{a^1-a^2}+u^{b^1-a^2}$ に適用することは、著しい利点をもつ。式は対称的であり、よって整数リスト表現を入力値として使うと $T(a_1-a_2, b_1-a_2)=T(b_1-a_2, a_1-a_2)$ となる。このことは、増分テーブルを圧縮された形で記憶することを許容する。テーブルの約半分が記憶されればよいのである。たとえば、 $T(x, y)$ を記憶するのは $x \leq y$ の場合のみとすることができる。この方法のわずかな潜在的な不都合は、中間整数リストが異なるエンコードを使うということである。

【0084】

50

さらなる変形として、増分テーブルは $u^{a^2-a^1}+u^{b^1-a^1}$ に適用されてもよい。

【0085】

上記の例示的エンコードについて示した原理はいくつかの代替的なエンコードに適用されうる。第一の代替的なエンコードは、エンコード $a=u^{a^1}+u^{a^2}$ を使って環要素 a を整数リスト (a_1, a_2) としてエンコードするというものである。任意の環要素がこの仕方ではエンコードされうるような基本環要素 u をもつ環は、正の難読化された算術を許容すると言われる。上記の例示的エンコードは、負の難読化された算術と称される。基本環要素 u をもつ正の難読化された算術を許容する任意の環について、 $u^m=-1$ となるような整数 m が存在することが数学的に証明されうる。さらに、負の難読化された算術を許容する環は、そのような値 m が存在する場合にかつその場合にのみ正の難読化された算術を許容する。正の難読化された算術を許容する任意の環は負の難読化された算術も許容するが、逆は真ではない。

10

【0086】

正の難読化された算術は、上記で概説した負の難読化された算術についてとほぼ同じ線に沿って行なわれる。手短かには、整数リストの符号の変化が、整数リスト中のすべての整数に値 m を加えることによって行なわれてもよい。加算入力 $a=u^{a^1}+u^{a^2}$ および $b=u^{b^1}+u^{b^2}$ を与えられると、加算は、中間的な $c=u^{a^1}+u^{a^2}+u^{b^1}$ を計算することによって、たとえば $(u^{a^1-b^1}+u^{a^2-b^1}+1)u^{b^1}$ を通じて、実行されてもよい。増分テーブルは、増分値 1 をもって $u^{a^1-b^1}+u^{a^2-b^1}$ に適用される。正の還元が二度、 u^{b^1} および u^{b^2} の両方について適用されてもよい。負の還元は必要ない。これは加算を単純化する。増分テーブルの構築は、上記のように、 u の異なる冪乗を使うことによって変えられてもよい。増分値は上記のように変えられてもよい。正の難読化された算術は、増分テーブルが常に対称的であり、圧縮された形で記憶されうるという利点をもつ。正の難読化の難点は、この型のエンコードを許容する環がより少ないということである。

20

【0087】

これまでに与えたエンコードは任意的に、何らかの v についての一定の環要素 $w=u^v$ を乗算されてもよい。こうして、整数リスト (a_1, a_2) は環要素 $a=w(u^{a^1}-u^{a^2})$ を表現してもよい。否定にするステップは変更ない。正の還元ステップは $c=wu^{a^1}-wu^{a^2}+wu^{b^1}=(wu^{a^1-b^1}-wu^{a^2-b^1}+w)u^{b^1}$ となる。増分テーブルは増分値として w を使ってもよく、同じエンコード型をもつ $wu^{a^1-b^1}-wu^{a^2-b^1}$ に適用される。乗算は、整数リスト (r_1, r_2) および (s_1, s_2) として表現された $r=wu^{r^1}-wu^{r^2}$ および $s=wu^{s^1}-wu^{s^2}$ を、 $r \cdot s=(wu^{r^1}-wu^{r^2})(wu^{s^1}-wu^{s^2})=(wu^{r^1+s^1+v}-wu^{r^1+s^2+v})+(wu^{r^2+s^2+v}-wu^{r^2+s^1+v})=t+u$ を使って乗算しうる。

30

【0088】

さらなる代替的なエンコードは $a=u^{a^1}(u^{a^2}-u^{-a^2})$ によって、または定数 $1/2$ をかけて $u^{a^1}((u^{a^2}-u^{-a^2})/2)$ によって与えられる。奇数の位数をもつ基本環要素 u をもつ負の難読化された算術を許容する環について、任意の環要素 x が $u^{a^1}((u^{a^2}-u^{-a^2})/2)$ と書けることを証明できる。これはエンコードを、たとえば整数リストから環要素へのマップを変更する。環が負の難読化をもつ場合も、基本環要素が奇数の位数をもつ限り、この表現ができる。

【0089】

加算および乗算ステップは、種々のエンコードについて対応して適応されることができ。たとえば、エンコードされた形の数 $a=u^{a^1}(u^{a^2}-u^{-a^2})$ を与えられたとき、 $a=u^{a^1-1}-u^{a^1-2}$ となるような a'_1 および a'_2 を a_1 および a_2 を使って計算してもよい。これはたとえば、 u の位数を法とする a_1+a_2 および u の位数を法とする a_1-a_2 を計算することによる。これらの整数を使って、上記の加算および乗算が使用されうる。

40

【0090】

双曲表現を得るためにやったことは、任意の種類の線形変換に一般化できる。変換が可逆であれば、新たな表現はもとの表現と等価である。

【0091】

表現 $a=u^{a^1}-u^{a^2}$ および行列形式で書かれた関係

50

【0092】

【数7】

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} m & n \\ o & p \end{pmatrix} \begin{pmatrix} a_3 \\ a_4 \end{pmatrix}$$

があるとする。

【0093】

変換が環 Z_k において単位元である行列式 $mp-no$ をもつ場合には、 a_3 および a_4 での表現は他方と等価である。 k は環 R における u の位数である。これは $\gcd(mp-no, k) = 1$ である場合にかつその場合にのみ成り立つ。双曲表現は例であり（ $1/2$ の乗算を含め）、 k が奇数であることを要求する。その場合には変換の行列式が2（または-2）になるからである。

10

【0094】

上記方法を別の例を用いて説明する。環 Z_{79} を考え、 $u=8$ とする。この元は位数 $k=13$ をもち、 Z_{79} におけるすべての元は、何らかの指数について差 $u^{a^1} - u^{a^2}$ として書けることがわかっている。変換

【0095】

【数8】

$$M = \begin{pmatrix} 10 & 8 \\ 6 & 4 \end{pmatrix}$$

を考える。行列式は $5 \pmod{13}$ であり、よって行列は逆をもつ。逆行列は

【0096】

20

【数9】

$$M^{-1} = \begin{pmatrix} 6 & 1 \\ 4 & 2 \end{pmatrix}$$

である。

【0097】

Z_{79} におけるすべての x について、 $x = 8^{\alpha} - 8^{\beta}$ となるような α および β をみつけられることがわかっている。だが、この変換を使うと、 $x = 8^{10\gamma + 8\delta} - 8^{6\gamma + 4\delta}$ となるような値 γ および δ をみつけることができることをただちに演繹できる。

【0098】

このことは、大きなクラスの諸表現が等価であることを示している。線形変換は、二つの加法的定数 r, s を含めて

30

【0099】

【数10】

$$\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} m & n \\ o & p \end{pmatrix} \begin{pmatrix} a_3 \\ a_4 \end{pmatrix} + \begin{pmatrix} r \\ s \end{pmatrix}$$

とすればアフィン変換に拡張されうる。

【0100】

この変換は、線形変換 M が可逆であれば、可逆である。

【0101】

整数リストにおける整数の数。

40

これまで論じた例では、整数リストにおける要素の数は常に2であった。この数には利点がある。すなわち、計算ステップ数を削減する。他方、整数リスト中により多くの要素を許容することは、難読化を許容する環の数を拡張する。下記の例は、リスト当たり三つの整数を考えるが、より多数が可能であり、同様に機能する。

【0102】

それぞれ要素 $u^{a^1} + u^{a^2} + u^{a^3}$ および $u^{b^1} + u^{b^2} + u^{b^3}$ をエンコードしている第一の整数リスト (a_1, a_2, a_3) および第二の整数リスト (b_1, b_2, b_3) を考える。否定にすることは、リスト中の整数に定数 m を加算することによってできる。加算は、第二の整数リストにおける各整数についての、この場合三回の増分テーブルの適用によってできる。第一の中間加算整数リストは $u^{a^1} + u^{a^2} + u^{a^3} + u^{b^3} = (u^{a^1 - b^3} + u^{a^2 - b^3} + u^{a^3 - b^3} + 1)u^{b^3}$ から計算されうる。この

50

場合、増分値は1であり、増分テーブルは $u^{a^1-b^3}+u^{a^2-b^3}+u^{a^3-b^3}$ に適用される。乗算するには、第二の整数リスト中と同数の中間乗算整数リスト、たとえば $(a_1+b_1, a_2+b_1, a_3+b_1)$ 、 $(a_1+b_2, a_2+b_2, a_3+b_2)$ 、 $(a_1+b_3, a_2+b_3, a_3+b_3)$ が作られる。

【0103】

複数の異なる基本環要素。

$u^s=v$ および $v^t=u$ となるような指数をもつ二つの基本元 u および v を考える。整数リスト (a_1, a_2) は環要素 $a=u^{a^1}-v^{a^2}$ をエンコードし、 (b_1, b_2) についても同様。否定は、 (a_1, a_2) を (sa_2, ta_1) にマッピングすることによって得られる。正の還元ステップ $u^{a^1}-v^{a^2}+u^{b^1}=(u^{a^1-b^1}-v^{a^2-tb^1}+1)u^{b^1}$ 。増分値は1であり、テーブルは整数リスト (a_1-b_1, a_2-tb_1) に適用される。負の還元は否定を使って正の還元に着せられる。乗算は加算に着せられる。

10

【0104】

下記では、負および／または正の難読化を許容する環についての例が与えられる。

【0105】

環 R は法 n についての整数環 Z_n であってもよい。

【0106】

たとえば、 n は13で、基本環要素 $u=4$ であってもよい。この要素は位数6をもつ。下記では、すべての環要素0～6が上記の例示的エンコードを使って整数リストとしてエンコードされる。ここで、すべての要素が複数のエンコードをもつことを注意しておく。挙げられている第一のエンコードについて、整数リストを与えられたときにどのように対応する環要素が見出されるかを例証するマッピングの例を与えた。環要素7～12は、環要素1～6を否定にすることによって見出される。

20

【0107】

【表1】

環要素	整数リスト	マッピング例
0	$(x, x) \quad 0 \leq x < 6$ は任意	$4^x - 4^x$
1	$(1, 2), (5, 4)$	$4^1 - 4^2$
2	$(0, 3), (2, 0), (3, 5)$	$4^0 - 4^3$
3	$(1, 0), (3, 4)$	$4^1 - 4^0$
4	$(0, 5), (2, 3)$	$4^0 - 4^5$
5	$(0, 4), (1, 3), (4, 1)$	$4^0 - 4^4$
6	$(2, 5), (4, 2), (5, 1)$	$4^2 - 4^5$

30

この例は、この環では $4^3 = -1$ なので、正の難読化も許容する。

【0108】

負の難読化を許容する n および u についての他の値は： $n=151, u=2$ ； $u=87, u=20; n=79, u=8$ などである。

40

【0109】

本発明者らは、負および／または正のエンコードを許容する環の多数の例を発見した。本稿に記載された与えられた負および／または正のエンコードから、多くの変形が導出可能であることを注意しておく。

【0110】

環 R は、多項式 f および法 n についての多項式環 $Z_n[x]/f(x)$ であってもよい。多項式は既約である必要はない。 f が既約でなければ、体ではない可換環が得られる。任意の可換な多項式環 R が難読化を許容することがわかる。

【0111】

50

たとえば、いくつかの体が与えられる。

【0 1 1 2】

体 $F(2^6)$

この体は $F_2[x]/(x^6+x^4+x^3+x+1)$ と同型である。基本 $u=x^3$ の位数は21である。

【0 1 1 3】

体 $F(2^8)$

この体は $F_2[x]/(x^8+x^4+x^3+x^2+1)$ と同型である。

基本 $u=x^3$ の位数は85である。

基本 $u=x+1$ の位数は51である。

【0 1 1 4】

体 $F(2^{10})$

この体は $F_2[x]/(x^{10}+x^6+x^5+x^3+x^2+x+1)$ と同型である。

基本 $u=x^3$ の位数は341である。

基本 $u=x^7+x^6+x^4+x^3+x^2+x$ の位数は93である。

【0 1 1 5】

体 $F(2^{12})$

この体は $F_2[x]/(x^{12}+x^7+x^6+x^5+x^3+x+1)$ と同型である。

基本 $u=x^3$ の位数は1365である。

基本 $u=x^5$ の位数は819である。

基本 $u=x^7$ の位数は585である。

基本 $u=x^9$ の位数は455である。

基本 $u=x^8+x^7+x^6+x^4+x^2+x$ の位数は315である。

基本 $u=x^{10}+x^9+x^8+x^6+x^4+x^3$ の位数は273である。

基本 $u=x^{11}+x^{10}+x^7+x^5+x^3+x^2+x+1$ の位数は195である。

【0 1 1 6】

図4は、可換環（たとえば Z_n ； $Z_n[x]/f(x)$ ）において難読化された算術を実行するための計算方法300の実施形態の例を概略的に示す。環は有限個の環要素をもつ。環要素に対して環加算および環乗算が定義されている。本計算方法は、環要素（ $u^{a^1}-u^{a^2}$ ）をエンコードする整数リスト（ (a_1, a_2) ）に対して作用する。整数リストは少なくとも二つ整数を有する。本計算方法は、

- ・増分環要素（ $1; u^l$ ）について定義された増分テーブル（ T ）を記憶する段階であって、前記増分テーブルは、出力環要素が増分環要素が入力環要素に環加算されたものに等しくなるよう（ $l=k+1$ ）入力環要素（ $k=u^{k^1}-u^{k^2}$ ）を出力環要素（ $l=u^{l^1}-u^{l^2}$ ）をエンコードする出力整数リストにマッピングする（ $T((k_1, k_2))=(l_1, l_2)$ ）、段階と；

- ・環加算する段階であって、前記環加算は

- ・第一の加算入力環要素をエンコードする第一の加算入力整数リスト（ (a_1, a_2) ）および第二の加算入力環要素をエンコードする第二の加算入力整数リスト（ (b_1, b_2) ）を受領310し、

- ・前記第一および第二の加算入力整数リストから決定される環要素に前記増分テーブルを適用することによって加算出力環要素をエンコードする加算出力整数リストを決定320することを含み、前記加算出力環要素は前記第一の加算入力環要素および前記第二の加算入力環要素の環加算に等しい、段階と；

- ・環乗算する段階であって、前記環乗算は

- ・第一の乗算入力環要素をエンコードする第一の乗算入力整数リスト（ (r_1, r_2) ）および第二の乗算入力環要素をエンコードする第二の乗算入力整数リスト（ (s_1, s_2) ）を受領330し、

- ・前記第一および第二の乗算入力整数リストから決定される環要素に前記増分テーブルを適用することによって乗算出力環要素をエンコードする乗算出力整数リストを決定340することを含み、前記乗算出力環要素は前記第一の乗算入力環要素および前記第二の乗算入力環要素の環乗算に等しい、段階とを含む。

10

20

30

40

50

【0117】

図5は、装置100または方法300などにおいて使用されうる加算方法400の実施形態の例を概略的に示している。この例は、上記の例示的エンコードを使う。本方法は、他のエンコードに適応されてもよい。本稿に記載されるすべての変形が適用されうる。この例は増分値1をもち、増分テーブルは因子 u^{b_1} をくり出すことによって構築される。

【0118】

方法400は、加算オペランドを受領すること410を含む。これは、第一の加算入力整数リスト、たとえば (a_1, a_2) を受領410し、第二の加算入力整数リスト、たとえば (b_1, b_2) を受領420することを含んでいてもよい。

【0119】

方法400はさらに、中間加算整数リスト、たとえば (c_1, c_2) を決定420することを含む。たとえば、これは、第一および第二の加算入力整数リストから決定された環要素に増分テーブルを適用することを含んでいてもよい。特に、増分テーブルは整数リストに適用されてもよい。該整数中の要素は入力整数リスト中の要素から導出される。

【0120】

たとえば、決定420は、 $(a_1 - b_1, a_2 - b_1)$ に増分テーブルを適用422してたとえば (d_1, d_2) を得て；第二の加算入力整数リストから決定された整数 b_1 を第一の適用から帰結する整数リスト中の整数に加算424すること、たとえば $(c_1, c_2) = (d_1 + b_1, d_2 + b_1)$ を含んでいてもよい。

【0121】

方法400はさらに、中間加算整数リストおよび第二の加算入力整数リストから決定された環要素への増分テーブルの第二の適用を通じて加算出力整数リストを決定430することを含む。より長い整数リストについては、これは追加的な増分テーブル適用を含んでもよい。たとえば、これは中間加算整数リストを否定すること431、たとえば (c_2, c_1) に並べ替えることを含んでいてもよい。増分テーブルの適用432および加算434は、加算入力整数リスト (a_1, a_2) が中間整数リスト (c_2, c_1) によって置き換わり、 b_1 が b_2 によって置き換わっているほかは、適用422および加算424と同じである。最後に、434の結果が否定にされて453、難読化加算の結果が得られる。

【0122】

今のように負の難読化の代わりに正の難読化が使われる場合には、否定431、435は省略されてもよい。

【0123】

図6は、装置100または方法300などにおいて使用されうる乗算方法500の実施形態の例を概略的に示している。この例は、方法400と同じエンコードおよび増分テーブルを使う。

【0124】

方法500は、乗算オペランドを受領すること510を含む。これは、第一の乗算入力整数リスト、たとえば (r_1, r_2) を受領510し、第二の乗算入力整数リスト、たとえば (s_1, s_2) を受領514することを含んでいてもよい。

【0125】

方法500はさらに、第一および第二の中間乗算整数リストを決定520することを含む。たとえば、520は、第一の中間乗算整数リストを決定し522、第二の中間乗算整数リストを決定する524ことを含んでいてもよい。これらはたとえば、それぞれ $(s_1 + r_1, s_1 + r_2)$ および $(s_2 + r_2, s_2 + r_1)$ として選ばれてもよいが、他の選択肢もある。乗算はこれらの数を加算方法400において加算することによって続く。

【0126】

テーブルは適用422および適用432において使われるだけであり、方法400および500の他のどこでも使われていないことを注意しておく。加算も乗算も同じテーブルを使い、いずれもテーブルを同じ回数（二回）使う。他の演算は、たとえば基本環要素の位数を法とした、整数リスト中の整数に対する小さな算術演算を含む。

10

20

30

40

50

【0127】

上記の諸方法を実行するための多くの異なる仕方が可能である。このことは当業者には明白であろう。たとえば、段階の順序は変えられてもよく、いくつかの段階は並列に実行されてもよい。さらに、段階と段階の間に、他の方法段階が挿入されてもよい。挿入される段階は、本稿に記載されるような方法の洗練を表わしていてもよく、あるいは本方法に無関係であってもよい。さらに、所与の段階は次の段階が開始される前に完全に完了していないこともありうる。

【0128】

実施形態に基づく方法は、プロセッサ・システムに方法300、400および500のうちの任意のものを実行させるための命令を含むソフトウェアを使って実行されてもよい。ソフトウェアは、システムの特定のサブエンティティによって行なわれる段階のみを含んでいてもよい。ソフトウェアは、ハードディスク、フロッピー、メモリなどといった好適な記憶媒体に記憶されていてもよい。ソフトウェアは、ワイヤに沿ったあるいは無線の信号として、あるいはデータ・ネットワーク、たとえばインターネットを使って送られてもよい。ソフトウェアはダウンロードおよび／またはサーバー上でのリモート使用のために利用可能にされてもよい。方法は、本方法を実行するようプログラム可能な論理、たとえばフィールドプログラマブルゲートアレイ (FPGA) を構成するよう構成されたビットストリームを使って実行されてもよい。

【0129】

実施形態が、実施形態を実施するために適応されたコンピュータ・プログラム、特に搬送波上または搬送波中のコンピュータ・プログラムに拡張されることは理解されるであろう。プログラムは、ソースコード、オブジェクトコード、部分的にコンパイルされた形のようなコード中間ソースおよびオブジェクトコードの形で、あるいは実施形態に基づく方法の実装において使うために好適な他の任意の形でありうる。コンピュータ・プログラム・プロダクトに関係する実施形態は、記載される方法のうち少なくとも一つの方法の処理段階のそれぞれに対応するコンピュータ実行可能命令を含む。これらの命令は、サブルーチンに分割され、および／または静的または動的にリンクされうる一つまたは複数のファイルにおいて記憶されてもよい。コンピュータ・プログラム・プロダクトに関係する別の実施形態は、記載されるシステムおよび／またはプロダクトのうち少なくとも一方の各手段に対応するコンピュータ実行可能命令を含む。

【0130】

図7のaは、コンピュータ・プログラム1020を有する書き込み可能部分1010を有するコンピュータ可読媒体1000を示している。コンピュータ・プログラム1020は、プロセッサ・システムに、実施形態に基づく難読化算術を実行するための計算方法を実行させるための命令を含む。書き込み可能部分は、複数回の書き込みのために構成されていてもよく、あるいは一度だけの書き込みのために構成されていてもよい。コンピュータ・プログラム1020はコンピュータ可読媒体1000上で、物理的なマークとして、あるいはコンピュータ可読媒体1000の磁化によって具現されてもよい。しかしながら、他の任意の好適な実施形態も考えられる。さらに、コンピュータ可読媒体1000はここでは光ディスクとして示されているが、コンピュータ可読媒体1000はハードディスク、半導体メモリ、フラッシュメモリなどといったいかなる好適なコンピュータ可読媒体であってもよく、記録不可能でも記録可能でもよいことは理解されるであろう。コンピュータ・プログラム1020は、難読化算術を実行するための前記計算方法をプロセッサ・システムに実行させるための命令を含む。

【0131】

コンピュータ可読媒体、たとえばコンピュータ可読媒体1000は増分テーブルおよび／またはデコード・テーブルおよび／またはエンコード・テーブルを含んでいてもよい。

【0132】

図7のbは、実施形態に基づく処理システム1100の概略的な表現を示している。プロセッサ・システムは、一つまたは複数の集積回路1110を有する。一つまたは複数の

10

20

30

40

50

集積回路 1 1 1 0 のアーキテクチャは図 7 の b に概略的に示されている。回路 1 1 1 0 は、実施形態に基づく方法を実行するためおよび／またはそのモジュールもしくはユニットを実装するためにコンピュータ・プログラム・コンポーネントを走らせるための処理ユニット 1 1 2 0、たとえば CPU を有する。回路 1 1 1 0 は、プログラミング・コード、データなどを記憶するためのメモリ 1 1 2 2 を有する。メモリ 1 1 2 2 の一部は読み出し専用であってもよい。回路 1 1 1 0 は通信要素 1 1 2 6、たとえばアンテナ、コネクタまたはその両方などを有していてもよい。回路 1 1 1 0 は、本方法において定義される処理の一部または全部を実行するための専用の集積回路 1 1 2 4 を有していてもよい。プロセッサ 1 1 2 0、メモリ 1 1 2 2、専用 IC 1 1 2 4 および通信要素 1 1 2 6 は相互接続 1 1 3 0、たとえばバスを介して互いに接続されてもよい。プロセッサ・システム 1 1 1 0 は、それぞれアンテナおよび／またはコネクタを使って接触式および／または非接触式通信のために構成されていてもよい。

10

【0 1 3 3】

上述した実施形態は、本発明を限定するのではなく例解するものであり、当業者は多くの代替実施形態を設計できるであろうことを注意しておくべきである。

【0 1 3 4】

請求項では、括弧内に参照符号があったとしてもその請求項を限定するものと解釈してはならない。動詞「有する／含む」およびその活用形の使用は、請求項において述べられているもの以外の要素やステップの存在を排除しない。要素の単数形の表現はそのような要素の複数の存在を排除しない。本発明は、いくつかの相異なる要素を有するハードウェアによって、および好適にプログラムされたコンピュータによって実装されうる。いくつかの手段を列挙する装置請求項では、これらの手段のいくつかが同一のハードウェア項目によって具現されてもよい。ある種の施策が互いに異なる従属請求項において記載されているというだけの事実が、これらの施策の組み合わせが有利に使用できないことを示すものではない。

20

【0 1 3 5】

請求項では、括弧内の参照符号は実施形態の図面における参照符号または実施形態の公式を指し、それにより請求項の理解しやすさを増す。これらの参照は網羅的ではなく、請求項を限定するものと解釈してはならない。

いくつかの態様を記載しておく。

30

〔態様 1〕

可換環 $(Z_n; Z_n[x]/f(x))$ において算術を実行するための電子計算装置であって、前記環は有限個の環要素をもち、環要素に対して環加算および環乗算が定義されており、当該計算装置は、環要素 $(u^{a^1} - u^{a^2})$ をエンコードする整数リスト $((a_1, a_2))$ に対して作用し、前記整数リストは少なくとも二つ整数を有し、整数リスト $((a_1, a_2))$ は、環要素 (a) を、該環要素が一つまたは複数の基本環要素 $(u; u, v)$ の冪乗の線形結合に等しくなる $(a = u^{a^1} - u^{a^2}; a = u^{a^1} - v^{a^2})$ ようエンコードし、前記冪乗は該整数リストによって決定される指数をもち、当該計算装置は、

・増分環要素 $(1; u^t)$ について定義された増分テーブル (T) を記憶するよう構成された記憶部を有しており、

40

・前記増分テーブルは、出力環要素が増分環要素が入力環要素に環加算されたものに等しくなる $(l = k + 1)$ よう入力環要素 $(k = u^{k^1} - u^{k^2})$ を出力環要素 $(l = u^{l^1} - u^{l^2})$ をエンコードする出力整数リストにマッピングし $(T((k_1, k_2)) = (l_1, l_2))$ 、

当該計算装置はさらに、環加算ユニットを有しており、前記環加算ユニットは、

・第一の加算入力環要素をエンコードする第一の加算入力整数リスト $((a_1, a_2))$ および第二の加算入力環要素をエンコードする第二の加算入力整数リスト $((b_1, b_2))$ を受領する段階であって、前記増分環要素は前記第一および第二の加算入力環要素から独立である、段階と、

・前記第一および第二の加算入力整数リストから決定される環要素に前記増分テーブルを適用することによって加算出力環要素をエンコードする加算出力整数リストを決定す

50

る段階であって、前記加算出力環要素は前記第一の加算入力環要素および前記第二の加算入力環要素の環加算に等しい、段階とを執行するよう構成されている、
計算装置。

〔態様 2〕

環乗算ユニットをさらに有しており、前記環乗算ユニットは、

・第一の乗算入力環要素をエンコードする第一の乗算入力整数リスト $((r_1, r_2))$ および第二の乗算入力環要素をエンコードする第二の乗算入力整数リスト $((s_1, s_2))$ を受領する段階と、

・前記第一および第二の乗算入力整数リストから決定される環要素に前記増分テーブルを適用することによって乗算出力環要素をエンコードする乗算出力整数リストを決定する段階であって、前記乗算出力環要素は前記第一の乗算入力環要素および前記第二の乗算入力環要素の環乗算に等しい、段階とを執行するよう構成されている、

態様 1 記載の計算装置。

〔態様 3〕

前記整数リスト $((a_1, a_2))$ が環要素 (a) をエンコードする仕方は、

・前記環要素が、基本元を前記整数リストの第一の整数によって決定される冪乗にしたものから、前記基本元を前記整数リストの第二の整数によって決定される冪乗にしたものを引いたもの $(a = u^{a_1} - u^{a_2})$ に任意的に定数を乗算したもの $(a = t(u^{a_1} - u^{a_2}))$ に等しくなるようにする、または

・前記環要素が、前記基本元を前記整数リストの第一の整数によって決定される冪乗にしたものに、前記基本元を前記整数リストの第二の整数によって決定される冪乗にしたものを足したもの $(a = u^{a_1} + u^{a_2})$ に任意的に定数を乗算したものに等しくなるようにする、または

・前記環要素が、前記基本元を前記整数リストの第一の整数によって決定される冪乗にしたもの、かける、前記基本元を前記整数リストの第二の整数によって決定される冪乗にしたものから、前記基本元を前記整数リストの第二の整数のマイナスによって決定される冪乗にしたものを引いたもの $(a = u^{a_1}(u^{a_2} - u^{-a_2}))$ に任意的に定数を乗算したもの $(a = u^{a_1}(u^{a_2} - u^{-a_2})/2)$ に等しくなるようにする、または

・前記環要素が、前記基本元を前記整数リストの第一の整数および第二の整数の第一の線形結合である冪乗にしたもの、足すまたは引く、前記基本元を前記第一の整数および前記第二の整数の第二の線形結合である冪乗にしたもの（行列 M を与えられて

【数 1 1】

$$M \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

となるような $x = u^{b_1} + u^{b_2}$ または $x = u^{b_1} - u^{b_2}$ ）に任意的に定数を乗算したものに等しくなるようにするものである、

態様 1 または 2 記載の計算装置。

〔態様 4〕

前記加算出力整数リストを決定することが、

・一つまたは複数の基本元の冪乗の線形結合である環要素 $(u^{a_1 - b_1} - u^{a_2 - b_1})$ への前記増分テーブルの第一の適用によって、中間加算環要素をエンコードする中間加算整数リスト $((c_1, c_2))$ を決定する段階であって、前記冪乗は前記第一および第二の加算入力整数リストから決定される、段階 $(u^{c_1} - u^{c_2} = u^{a_1} - u^{a_2} + u^{b_1})$ と、

・前記加算出力整数リストを決定する段階であって、前記中間加算整数リストから決定されるおよび前記第二の加算入力整数リストから決定される環要素への前記増分テーブルの第二の適用を含む、段階とを含む、

態様 1 または 3 記載の計算装置。

〔態様 5〕

前記中間加算整数リスト $((c_1, c_2))$ を決定することがさらに、前記第一および第二の加算入力整数リストから決定された整数を、第一の適用から帰結する整数リスト中の整数

10

20

30

40

50

に加算することを含む、態様4記載の計算装置。

〔態様6〕

・前記増分テーブルは、前記一つまたは複数の基本環要素(u)を前記第一の整数リストの第一の整数(a_1)から前記第二の整数リストの第一の整数(b_1)を引いたもの乗したものの、足すまたは引く、ある基本環要素(u)を前記第一の整数リストの第二の整数(a_2)から前記第二の整数リストの第一の整数(b_1)を引いたもの乗したものの、によって形成される環要素($u^{a_1-b_1}-u^{a_2-b_1}$; $u^{a_1-b_1}+u^{a_2-b_1}$)に適用される、および/または

・前記増分テーブルは、前記一つまたは複数の基本環要素(u)を前記第一の整数リストの前記第一の整数(a_1)から前記第一の整数リストの第二の整数(a_2)を引いたもの乗したものの、足すまたは引く、ある基本環要素(u)を前記第二の整数リストの第一の整数(b_1)から前記第一の整数リストの前記第二の整数(a_2)を引いたもの乗したものの、によって形成される環要素($u^{a_1-a_2}+u^{b_1-a_2}$; $u^{a_1-a_2}+u^{b_1-a_2}$)に適用される、および/または

・前記増分テーブルは、前記一つまたは複数の基本環要素(u)を前記第一の整数リストの第二の整数(a_2)から前記第一の整数リストの第一の整数(a_1)を引いたもの乗したものの、足すまたは引く、ある基本環要素(u)を前記第二の整数リストの第一の整数(b_1)から前記第一の整数リストの前記第一の整数(a_1)を引いたもの乗したものの、によって形成される環要素($u^{b_1-a_1}-u^{a_2-a_1}$; $u^{a_2-a_1}+u^{b_1-a_1}$)に適用される、

態様4または5記載の計算装置。

〔態様7〕

・前記増分テーブルの前記第二の適用の前に、前記中間加算整数リストによって表わされる前記中間加算環要素が否定にされる、

態様4ないし6のうちいずれか一項記載の計算装置。

〔態様8〕

・整数リストによって表わされる環要素が、該整数リストを並べ替えることによって否定にされる、および/または

・整数リストによって表わされる環要素が、該整数リストの各整数に定数を加えるることによって否定にされる、および/または

・整数リスト((a_1, a_2))によって表わされる環要素が、該整数リストを並べ替え、該整数リストの一つまたは複数の整数に定数を乗算する((sa_2, ta_1))ことによって否定にされる、

態様1ないし7のうちいずれか一項記載の計算装置。

〔態様9〕

前記増分テーブルが、前記入力環要素($k=u^{k_1}-u^{k_2}$)を表わす入力整数リスト((k_1, k_2))を入力として取る、態様1ないし8のうちいずれか一項記載の計算装置。

〔態様10〕

乗算出力整数リストを決定することが、

・前記第一および第二の乗算入力整数リストから、それぞれ第一および第二の中間乗算要素をエンコードする第一の中間乗算整数リスト((t_1, t_2))および第二の中間乗算整数リスト((u_1, u_2))を決定し、

・前記第一および第二の中間乗算整数リストを前記環加算ユニットを通じて加算することを含む、

態様2記載の計算装置。

〔態様11〕

・前記第一の中間乗算整数リストの第一の整数(t_1)は、前記第一の乗算入力整数リストの第一の整数(r_1)、足す、前記第二の乗算入力整数リストの第一の整数(s_1)であり、

・前記第一の中間乗算整数リストの第二の整数(t_2)は、前記第一の乗算入力整数リストの第一の整数(r_1)、足す、前記第二の乗算入力整数リストの第二の整数(s_2)であり($t_1=r_1+s_1$; $t_2=r_1+s_2$)、

・前記第二の中間乗算整数リストの第一の整数(u_1)は、前記第一の乗算入力整数リストの第二の整数(r_2)、足す、前記第二の乗算入力整数リストの第二の整数(s_2)であり、

10

20

30

40

50

・前記第二の中間乗算整数リストの第二の整数 (u_2) は、前記第一の乗算入力整数リストの第二の整数 (r_2)、足す、前記第二の乗算入力整数リストの第一の整数 (s_1) である ($u_1 = r_2 + s_2$ 、 $u_2 = r_2 + s_1$)、
 態様 10 記載の計算装置。

〔態様 12〕

・前記可換環が、ある整数を法とする整数によって形成される環 (Z_n) である、または
 ・前記可換環が、ある整数多項式を法とする整数多項式によって形成される環 ($Z_n[x]/f(x)$) である、
 態様 11 ないし 12 のうちいずれか一項記載の計算装置。

〔態様 13〕

態様 1 記載の計算装置と一緒に使うための、可換環 ($Z_n; Z_n[x]/f(x)$) の環要素を整数リストとしてエンコードする環エンコード装置であって、当該環エンコード装置は、
 ・一つまたは複数の基本環要素 (u) について定義されたエンコード・テーブルを記憶するよう構成された記憶部を有しており、前記エンコード・テーブルは、環要素 (x) を、環要素が前記一つまたは複数の基本環要素の冪乗の線形結合に等しい ($x = u^a - u^b$) ように整数リスト ((a, b)) にマッピングするものであり、前記冪乗は前記整数リストによって決定される指数をもつ、
 環エンコード装置。

〔態様 14〕

態様 1 記載の計算装置と一緒に使うための、整数リスト ((a, b)) を可換環 ($Z_n; Z_n[x]/f(x)$) の環要素 (x) にデコードする環デコード装置であって、当該環デコード装置は、
 ・一つまたは複数の基本環要素 (u) について、前記環要素 (x) を、前記環要素が一つまたは複数の基本環要素の冪乗の線形結合に等しい ($x = u^a - u^b$) ように決定するよう構成されており、前記冪乗は前記整数リストによって決定される指数をもつ、
 環デコード装置。

〔態様 15〕

可換環 ($Z_n; Z_n[x]/f(x)$) において算術を実行するための計算装置において使うための増分テーブルを計算するテーブル計算装置であって、前記環は有限個の環要素をもち、環要素に対して環加算および環乗算が定義されており、前記計算装置は、環要素 ($u^{a1} - u^{a2}$) をエンコードする整数リスト ((a_1, a_2)) に対して作用し、前記整数リストは少なくとも二つ整数を有し、当該テーブル計算装置は、
 ・増分テーブルを構築するよう構成されたテーブル生成ユニットを有し、前記テーブル生成ユニットは、
 ・入力環要素を反復的に選択し、
 ・増分環要素が前記入力環要素に環加算されたものに等しい出力環要素を決定し、
 ・前記出力環要素についてエンコードする出力整数リストを決定し、
 ・前記入力環要素を前記出力整数リストにマッピングするエントリーを増分テーブルに追加するよう構成されている、
 テーブル計算装置。

〔態様 16〕

可換環 ($Z_n; Z_n[x]/f(x)$) において算術を実行するための電子的な計算方法であって、前記環は有限個の環要素をもち、環要素に対して環加算および環乗算が定義されており、当該計算方法は、環要素 ($u^{a1} - u^{a2}$) をエンコードする整数リスト ((a_1, a_2)) に対して作用し、前記整数リストは少なくとも二つ整数を有し、整数リスト ((a_1, a_2)) は、環要素 (a) を、該環要素が一つまたは複数の基本環要素 ($u; u, v$) の冪乗の線形結合に等しくなる ($a = u^{a1} - u^{a2}$; $a = u^{a1} - v^{a2}$) ようエンコードし、前記冪乗は該整数リストによって決定される指数をもち、当該計算方法は、
 ・増分環要素 ($1; u^t$) について定義された増分テーブル (T) を記憶する段階であって、前記増分テーブルは、出力環要素が増分環要素が入力環要素に環加算されたものに等しくなる ($1 = k+1$) よう入力環要素 ($k = u^{k1} - u^{k2}$) を出力環要素 ($1 = u^{11} - u^{12}$) をエンコ

10

20

30

40

50

ードする出力整数リストにマッピングする ($T((k_1, k_2)) = (l_1, l_2)$)、段階と、

・環加算する段階であって、前記環加算は、

・第一の加算入力環要素をエンコードする第一の加算入力整数リスト $((a_1, a_2))$ および第二の加算入力環要素をエンコードする第二の加算入力整数リスト $((b_1, b_2))$ を受領し、

・前記第一および第二の加算入力整数リストから決定される環要素に前記増分テーブルを適用することによって加算出力環要素をエンコードする加算出力整数リストを決定することを含み、前記加算出力環要素は前記第一の加算入力環要素および前記第二の加算入力環要素の環加算に等しい、

電子的な計算方法。

〔態様 17〕

環乗算する段階を含み、前記環乗算は、

・第一の乗算入力環要素をエンコードする第一の乗算入力整数リスト $((r_1, r_2))$ および第二の乗算入力環要素をエンコードする第二の乗算入力整数リスト $((s_1, s_2))$ を受領し、

・前記第一および第二の乗算入力整数リストから決定される環要素に前記増分テーブルを適用することによって乗算出力環要素をエンコードする乗算出力整数リストを決定することを含み、前記乗算出力環要素は前記第一の乗算入力環要素および前記第二の乗算入力環要素の環乗算に等しい、

態様 16 記載の電子的な計算方法。

〔態様 18〕

プログラム可能な装置で実行されたときに態様 16 または 17 記載の方法を実行するよう構成されているコンピュータ・プログラム命令を有するコンピュータ・プログラム。

〔態様 19〕

・態様 18 記載のコンピュータ・プログラム、および／または

・態様 1 記載の電子計算装置によって使用される前記増分テーブルを有する、コンピュータ可読媒体。

【符号の説明】

【0136】

- 100 計算装置
- 110 増分テーブルを記憶するよう構成された記憶部
- 120 環否定ユニット
- 130 環加算ユニット
- 140 環乗算ユニット
- 150 オペランド・ストア
- 160 デコード・ユニット
- 170 エンコード・ユニット
- 172 エンコード・テーブルを記憶するよう構成された記憶部

10

20

30

【図 1 a】

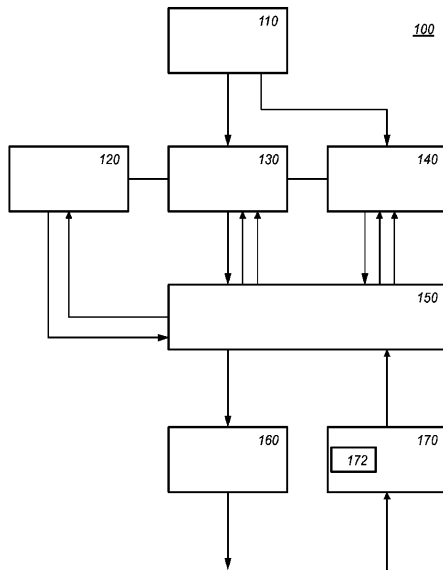


Fig. 1a

【図 1 b】

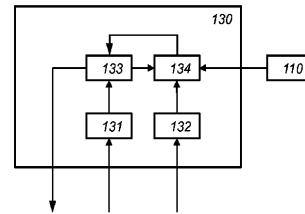


Fig. 1b

【図 1 c】

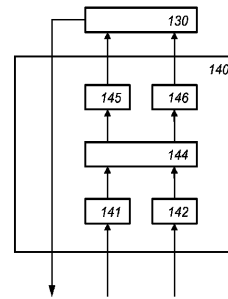


Fig. 1c

【図 2】

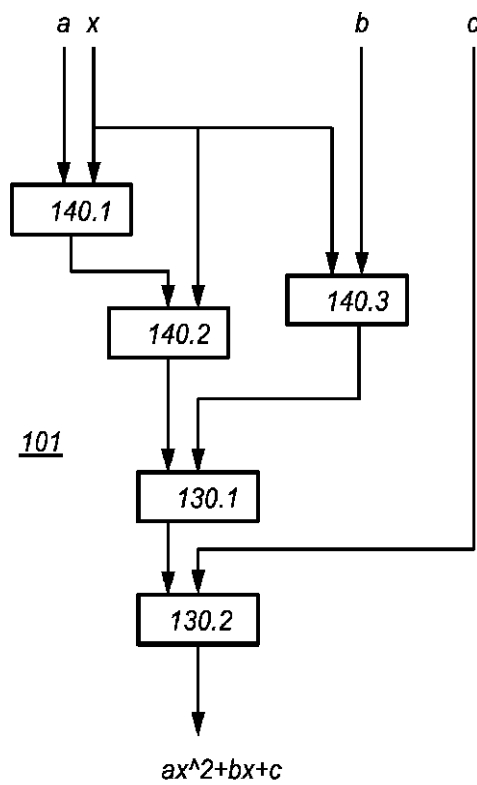


Fig. 2

【図 3】

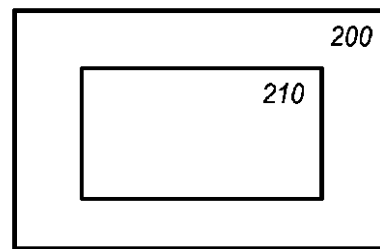


Fig. 3

【図 4】

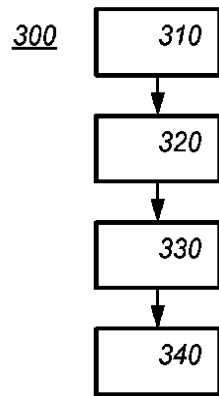


Fig. 4

【図 5】

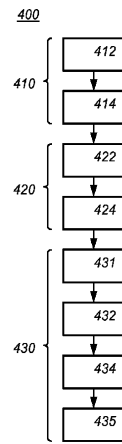


Fig. 5

【図 6】

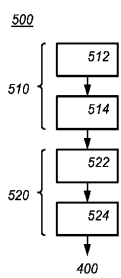


Fig. 6

【図 7 a】

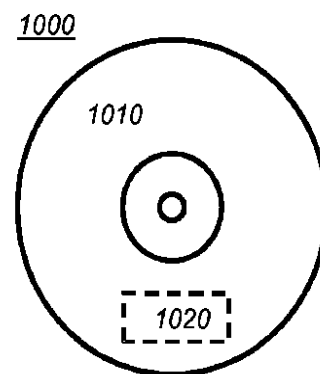
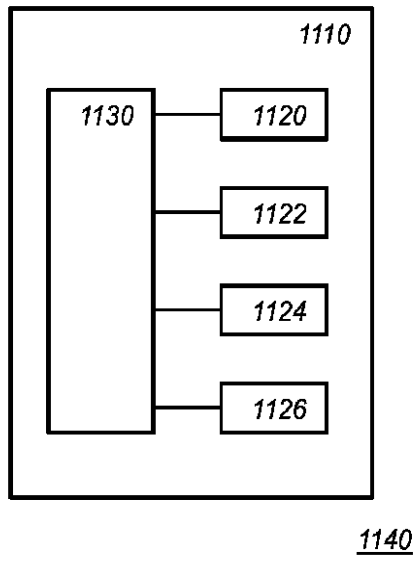


Fig. 7a

【図 7 b】

*Fig. 7b*

フロントページの続き

(74)代理人 100091214

弁理士 大貫 進介

(72)発明者 マリン, レアンドロ

オランダ国 5656 アーエー アインドーフエン ハイテック キャンパス 5

(72)発明者 ブリュケルス, アルフォンス アントニウス マリア ランベルテウス

オランダ国 5656 アーエー アインドーフエン ハイテック キャンパス 5

(72)発明者 ホリッセン, パウリウス マティアス ヒューベルテウス メヒティルデイス アントニウス

オランダ国 5656 アーエー アインドーフエン ハイテック キャンパス 5

審査官 青木 重徳

(56)参考文献 特表2008-541540 (JP, A)

国際公開第2014/095772 (WO, A1)

国際公開第2014/006605 (WO, A1)

国際公開第2007/126049 (WO, A1)

国際公開第2006/118086 (WO, A1)

国際公開第2005/098795 (WO, A1)

赤井 健一郎 ほか, White-Box AESのプログラムサイズ削減に関する一方式, コンピュータセキュリティシンポジウム2008 論文集, 日本, 社団法人情報処理学会, 2008年10月 8日, 第2008巻 第8号, p. 373-378

(58)調査した分野(Int.Cl., DB名)

G09C 1/00