(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7405157号
(P7405157)

(45)発行日　令和5年12月26日(2023.12.26)

(24)登録日　令和5年12月18日(2023.12.18)

(51)国際特許分類
G09C　1/00　(2006.01)

FI
G09C　1/00　650Z

請求項の数　12　（全13頁）

(21)出願番号　特願2021-572125(P2021-572125)
(86)(22)出願日　令和2年1月20日(2020.1.20)
(86)国際出願番号　PCT/JP2020/001680
(87)国際公開番号　WO2021/149103
(87)国際公開日　令和3年7月29日(2021.7.29)
審査請求日　令和4年6月23日(2022.6.23)

(73)特許権者　000004226
日本電信電話株式会社
東京都千代田区大手町一丁目５番１号
(74)代理人　100121706
弁理士　中尾 直樹
(74)代理人　100128705
弁理士　中村 幸雄
(74)代理人　100147773
弁理士　義村 宗洋
(72)発明者　五十嵐 大
東京都千代田区大手町一丁目５番１号
日本電信電話株式会社内
審査官　青木 重徳

最終頁に続く

(54)【発明の名称】　秘密計算装置、秘密計算方法、およびプログラム

(57)【特許請求の範囲】

【請求項１】

xが実数であり、［μ］がμの秘密分散値であり、nが1以上の整数であり、t=0,…,n-1であり、u=1,…,n-1であり、$f_t(x)$が前記実数xに対する関数であり、$f'_t(x)$は前記関数$f_t(x)$の近似関数であり、近似関数$f'_0(x)$の秘密分散値［$f'_0(x)$］が$[f'_0(x)]=c_{0,0}+c_{0,1}[x]$であり、近似関数$f'_u(x)$の秘密分散値［$f'_u(x)$］が$[f'_u(x)]=c_{u,0}+c_{u,1}[x]+c_{u,2}[f_0(x)]+\cdots+c_{u,u+1}[f_{u-1}(x)]$であり、$c_{t,0}$は公開値であり、$c_{t,1},\cdots,c_{t,n+1}$は係数であり、

前記実数xの秘密分散値［x］を用いた秘密計算によって$f_t(x)-f'_t(x)$の秘密分散値［$f_t(x)-f'_t(x)$］を得る第１秘密計算部と、

前記秘密分散値［$f_t(x)-f'_t(x)$］を用いた秘密計算によって$f_t(x)-f'_t(x)$を所定ビット数だけ右シフトした$(f_t(x)-f'_t(x))_r$の秘密分散値［$f_t(x)-f'_t(x)$］$_r$を得る第２秘密計算部と、

を有する秘密計算装置。

【請求項２】

請求項１の秘密計算装置であって、

前記秘密分散値［$f_t(x)-f'_t(x)$］$_r$と前記秘密分散値［$f'_t(x)$］を用いた秘密計算によって前記関数$f_t(x)$の秘密分散値［$f_t(x)$］を得る第３秘密計算部をさらに有する秘密計算装置。

【請求項３】

請求項１または２の秘密計算装置であって、

前記第１秘密計算部は、前記秘密分散値［x］を用いた積和演算の秘密計算によって前記秘密分散値［$f_t(x)-f'_t(x)$］を得る秘密計算装置。

【請求項４】
請求項２の秘密計算装置であって、
nが2以上の整数であり、
t=0,…,n-2について、前記第１秘密計算部と前記第２秘密計算部と前記第３秘密計算部の処理を実行するたびに、t+1を新たなtとして、前記第１秘密計算部と前記第２秘密計算部と前記第３秘密計算部の処理を再び実行し、秘密分散値［$f_{n-1}(x)$］を得る秘密計算装置。
【請求項５】
請求項２の秘密計算装置であって、
n=3であり、
a, b, c, d, f, g, h, i, j, k, s, m, n, o, p, q, α，β，γ，δ，ζが実数であり、
$f_0(x)=y=\delta x^2+ax$であり、
$f_1(x)=z=y(\zeta y+b)+cx$であり、
$f_2(x)=w=\gamma(z(\alpha z+d)+y(\beta x+f)+gx)$であり、
$f'_0(x)=ix+j$であり、
$f'_1(x)=ky+sx+m$であり、
$f'_2(x)=nz+oy+px+q$である、秘密計算装置。
【請求項６】
請求項５の秘密計算装置であって、
前記第１秘密計算部は、前記秘密分散値［x］を用いた積和演算の秘密計算によって秘密分散値［$f_0(x)-f'_0(x)$］=[y’]=[x(δx+a-i)-j]を得、
前記第２秘密計算部は、前記秘密分散値[y’]を用いた秘密計算によってy’を所定ビット数だけ右シフトしたy’の秘密分散値［$y'\rangle_r$］を得、
前記第３秘密計算部は、前記秘密分散値［$y'\rangle_r$］と前記秘密分散値［$f'_0(x)$］=［ix+j］を用いた秘密計算によって秘密分散値[y]=［y’+(ix+j)］を得、
前記第１秘密計算部は、前記秘密分散値［x］および前記秘密分散値[y]を用いた積和演算の秘密計算によって秘密分散値［$f_1(x)-f'_1(x)$］=[z’]=[y(ζy+b-k)+(c-s)x-m]を得、
前記第２秘密計算部は、前記秘密分散値[z’]を用いた秘密計算によってz’を所定ビット数だけ右シフトしたz’の秘密分散値［$z'\rangle_r$］を得、
前記第３秘密計算部は、前記秘密分散値［$z'\rangle_r$］と前記秘密分散値［$f'_1(x)$］=[ky+sx+m]を用いた秘密計算によって秘密分散値[y]=［z’+(ky+sx+m)］を得、
前記第１秘密計算部は、前記秘密分散値［x］、前記秘密分散値［y］、および前記秘密分散値[z]を用いた積和演算の秘密計算によって秘密分散値［w’/γ］=[z(αz+d-n/γ)+(βx+f-o/γ)y+(g-p)x+(h-q)/γ]を得、
前記第２秘密計算部は、前記秘密分散値［w’/γ］を用いた秘密計算によってw’/γにγを乗算して得られるw’を所定ビット数だけ右シフトしたw’の秘密分散値［$w'\rangle_r$］を得、
前記第３秘密計算部は、前記秘密分散値［$w'\rangle_r$］と前記秘密分散値［$f'_2(x)$］=[nz+oy+px+q]を用いた秘密計算によって秘密分散値[w]=［w’+(nz+oy+px+q)］を得る秘密計算装置。
【請求項７】
請求項６の秘密計算装置であって、
σが正整数であり、
前記第２秘密計算部は、公開値$2^\sigma/\gamma$を得、前記公開値$2^\sigma/\gamma$と前記秘密分散値［w’/γ］を用いた公開値除算の秘密計算[w’/γ]/($2^\sigma/\gamma$)によって前記秘密分散値［$w'\rangle_r$］を得る秘密計算装置。
【請求項８】
請求項２の秘密計算装置であって、
n=2であり、
a, b, c，γ，δ，i, j, k, s, mが実数であり、
$f_0(x)=y=\delta x^2+ax$であり、

$f_1(x)=z=\gamma(y(\delta y+b)+cx)$であり、
$f'_0(x)=ix+j$であり、
$f'_1(x)=ky+sx+m$である、秘密計算装置。
【請求項９】
請求項８の秘密計算装置であって、
前記第１秘密計算部は、前記秘密分散値［x］を用いた積和演算の秘密計算によって秘密分散値［$f_0(x)-f'_0(x)$］=[y’]=[x(δx+a-i)-j]を得、
前記第２秘密計算部は、前記秘密分散値[y’]を用いた秘密計算によってy’を所定ビット数だけ右シフトしたy’の秘密分散値［y’］$_r$を得、
前記第３秘密計算部は、前記秘密分散値［y’］$_r$と前記秘密分散値［$f'_0(x)$］=［ix+j］を用いた秘密計算によって秘密分散値[y]=［y’+(ix+j)］を得、
前記第１秘密計算部は、前記秘密分散値［x］と前記秘密分散値[y]を用いた積和演算の秘密計算によって秘密分散値[z’/γ]=[y(ζy+b-k/γ)+(c-s/γ)x-m/γ]を得、
前記第２秘密計算部は、前記秘密分散値［z’/γ］を用いた秘密計算によってz’/γにγを乗算して得られるz’を所定ビット数だけ右シフトしたz’の秘密分散値［z’］$_r$を得、
前記第３秘密計算部は、前記秘密分散値［z’］$_r$と前記秘密分散値［$f'_1(x)$］=[ky+sx+m]を用いた秘密計算によって秘密分散値[z]=［z’+(ky+sx+m)］を得る、秘密計算装置。
【請求項１０】
請求項９の秘密計算装置であって、
σが正整数であり、
前記第２秘密計算部は、公開値$2^\sigma/\gamma$を得、前記公開値$2^\sigma/\gamma$と前記秘密分散値［z’/γ］を用いた公開値除算の秘密計算[z’/γ]/($2^\sigma/\gamma$)によって前記秘密分散値［z’］$_r$を得る秘密計算装置。
【請求項１１】
xが実数であり、［a］がaの秘密分散値であり、nが1以上の整数であり、t=0,…,n-1であり、u=1,…,n-1であり、$F_t(x)$が前記実数xに対する関数であり、$f'_t(x)$は前記関数$f_t(x)$の近似関数であり、近似関数$f'_0(x)$の秘密分散値［$f'_0(x)$］が$[f'_0(x)]=c_{0,0}+c_{0,1}[x]$であり、近似関数$f'_u(x)$の秘密分散値［$f'_u(x)$］が$[f'_u(x)]=c_{u,0}+c_{u,1}[x]+c_{u,2}[f_0(x)]+\cdots+c_{u,u+1}[f_{u-1}(x)]$であり、$c_{t,0}$は公開値であり、$c_{t,1},\cdots,c_{t,n+1}$は係数であり、
第１秘密計算部で、前記実数xの秘密分散値［x］を用いた秘密計算によって$f_t(x)-f'_t(x)$の秘密分散値［$f_t(x)-f'_t(x)$］を得る第１秘密計算ステップと、
第２秘密計算部で、前記秘密分散値［$f_t(x)-f'_t(x)$］を用いた秘密計算によって$f_t(x)-f'_t(x)$を所定ビット数だけ右シフトした$(f_t(x)-f'_t(x))_r$の秘密分散値［$f_t(x)-f'_t(x)$］$_r$を得る第２秘密計算ステップと、
を有する秘密計算方法。
【請求項１２】
請求項１から１０の何れかの秘密計算装置としてコンピュータを機能させるためのプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、秘密計算に関する。
【背景技術】
【０００２】
近年、秘密計算による高度な統計や機械学習の研究が盛んになってきている。しかしこれらの演算のほとんどは秘密計算で得意な加減乗算を超える、逆数、平方根、指数、対数などの初等関数群の計算を含んでいる。これらは秘密計算の応用研究を花開かせる観点で極めて大きな障害である。これに対し、非特許文献１では、逆数、除数秘匿除算、平方根とその逆数,指数などの計算方法を提示している。
【先行技術文献】

【非特許文献】
【０００３】
【文献】五十嵐大，“秘密計算AIの実装に向けた秘密実数演算群の設計と実装-O(¦p¦)ビット通信量O(1)ラウンドの実数向け右シフト，”In CSS2019, 2019.
【発明の概要】
【発明が解決しようとする課題】
【０００４】
しかしながら、秘密計算によって右シフトや公開値による除算を行う場合に、オーバーフローによって正しく計算ができなくなってしまう場合がある。一方、オーバーフローを防ぐために右シフトを行って小数領域へのビット割り当てを減らして整数領域へのビット割り当てを増やしたのでは精度が低下する。
【０００５】
本発明はこのような点に鑑みてなされたものであり、高い精度を保ちつつ、オーバーフローを抑制する秘密計算技術を提供する。
【課題を解決するための手段】
【０００６】
xが実数であり、［μ］がμの秘密分散値であり、nが1以上の整数であり、$t=0,\cdots,n-1$であり、$u=1,\cdots,n-1$であり、$f_t(x)$が前記実数xに対する関数であり、$f'_t(x)$は関数$f_t(x)$の近似関数であり、近似関数$f'_0(x)$の秘密分散値［$f'_0(x)$］が$[f'_0(x)]=c_{0,0}+c_{0,1}[x]$であり、近似関数$f'_u(x)$の秘密分散値［$f'_u(x)$］が$[f'_u(x)]=c_{u,0}+c_{u,1}[x]+c_{u,2}[f_0(x)]+\cdots+c_{u,u+1}[f_{u-1}(x)]$であり、$c_{t,0}$は公開値であり、$c_{t,1},\cdots,c_{t,n+1}$は係数であるとする。本発明では、実数$x$の秘密分散値［$x$］を用いた秘密計算によって$f_t(x)-f'_t(x)$の秘密分散値［$f_t(x)-f'_t(x)$］を得、秘密分散値［$f_t(x)-f'_t(x)$］を用いた秘密計算によって$f_t(x)-f'_t(x)$を所定ビット数だけ右シフトした$(f_t(x)-f'_t(x))_r$の秘密分散値［$f_t(x)-f'_t(x)$］$_r$を得る。
【発明の効果】
【０００７】
本発明では、高い精度を保ちつつ、オーバーフローを抑制することができる。
【図面の簡単な説明】
【０００８】
【図１】図１は実施形態の秘密計算装置を例示したブロックである。
【図２】図２は第１実施形態の処理を説明するためのフロー図である。
【図３】図３は第２実施形態の処理を説明するためのフロー図である。
【図４】図４は第３実施形態の処理を説明するためのフロー図である。
【図５】図５は各初等関数に関する計算済みのパラメータを例示した表である。
【図６】図６はハードウェア構成を説明するためのブロック図である。
【発明を実施するための形態】
【０００９】
以下、図面を参照して本発明の実施の形態を説明する。
近年、秘密計算による高度な統計や機械学習の研究が盛んになってきている。しかしこれらの演算のほとんどは秘密計算の得意な加減乗算を超える、逆数、平方根、指数、対数などの初等関数計算を含んでいる。初等関数等の基礎的な関数の関数近似法にはTaylor展開などがある。Taylor展開などは多項式であり、任意の関数を多項式で近似することで、秘密計算の得意な加減乗算を用いて当該関数の近似計算を行うことができる。
【００１０】
以下の実施形態では、任意の関数を多項式関数$f_t(x)$で近似し、さらに右シフト前の関数$f_t(x)$と当該関数$f_t(x)$の近似関数$f'_u(x)$との差分$f_t(x)-f'_t(x)$の秘密分散値［$f_t(x)-f'_t(x)$］を計算し、$f_t(x)-f'_t(x)$を右シフトした$(f_t(x)-f'_t(x))_r$の秘密分散値［$f_t(x)-f'_t(x)$］$_r$を得、秘密分散値［$f_t(x)-f'_t(x)$］$_r$と秘密分散値［$f'_t(x)$］の秘密計算によって$f_t(x)-f'_t(x)$に$f'_t(x)$を加算した関数$f_t(x)$の秘密分散値［$f_t(x)$］を得る。ただし、xが実数であり、［μ］がμの秘密分散値であり、nが1以上の整数（例えば、nは2以上の整数）であり、$t=0,\cdots,n-1$

であり、u=1,…,n-1であり、$f_t(x)$が実数xに対する関数であり、$f'_t(x)$は関数$f_t(x)$の近似関数であり、近似関数$f'_0(x)$の秘密分散値［$f'_0(x)$］が$[f'_0(x)]=c_{0,0}+c_{0,1}[x]$であり、近似関数$f'_u(x)$の秘密分散値［$f'_u(x)$］が$[f'_u(x)]=c_{u,0}+c_{u,1}[x]+c_{u,2}[f_0(x)]+\cdots+[f_{u-1}(x)]$であり、$c_{t,0}$は公開値であり、$c_{t,1},\cdots,c_{t,n+1}$は係数である。ただし、$c_{t,1},\cdots,c_{t,n+1}$は有効ビット数の小さな値であり、$c_{t,1},\cdots,c_{t,n+1}$が乗じられても桁あふれによってシフトが必要になるようなことがない値である。$f_t(x)-f'_t(x)$は正である。また環上の整数に公開の小数点位置を定めることで固定小数点の実数と見なすことができる。実施形態ではこのようにして環上で表した固定小数点の実数を単に実数と表記する。秘密分散方式に限定はなく、例えば、加法的秘密分散方式やシャミア秘密分散方式などを例示できる。［μ］の一例は剰余環上の要素μを線形秘密分散した秘密分散値（シェア）である。

【００１１】

ここで$f_t(x)-f'_t(x)$の大きさは$f_t(x)$の大きさよりも小さいため、秘密分散値［$f_t(x)-f'_t(x)$］のオーバーフローを抑制することができる。また右シフト前の関数$f_t(x)$と当該関数$f_t(x)$の近似関数$f'_u(x)$との差分$f_t(x)-f'_t(x)$の秘密分散値［$f_t(x)-f'_t(x)$］を計算するため、高い精度を保つことができる。オーバーフローは秘密計算を実装したプロセッサの性能に基づく問題であり、本方式はこのハードウェア上の制約に基づく問題を解決するための手法を提供する。このように、本方式は純粋数学上の問題を解決するものではなく、ハードウェア実装上の問題を解決するものであって技術的特徴を有するものである。例えば、秘密分散値［$f_t(x)$］を計算するとオーバーフローしてしまうが秘密分散値［$f_t(x)-f'_t(x)$］の計算ではオーバーフローしないプロセッサではその技術的特徴は顕著である。

【００１２】

以下に各実施形態を説明する。

［第１実施形態］

図１に例示するように、第１実施形態の秘密計算装置１は、秘密計算部１１，１２，１３、および制御部１９を有する。本実施形態の秘密計算装置１は、実数xの秘密分散値［x］∈［Ｌ，Ｒ）を入力とし、秘密計算を行って目的の関数$f_{n-1}(x)$の秘密分散値［$f_{n-1}(x)$］を出力する。なお、Ｌ，ＲはＬ＜Ｒを満たす実数であり、［Ｌ，Ｒ）はＬ以上Ｒ未満の左閉右開区間を表す。関数$f_{n-1}(x)$の例は初等関数を近似する多項式である。$f_{n-1}(x)$を得る過程で表れる関数を$f_0(x),\cdots,f_{n-2}(x)$と表記する。以下、図２を用いて詳細に説明する。

【００１３】

図２に例示するように、まず秘密計算装置１の秘密計算部１１に秘密分散値［x］が入力される（ステップＳ１０）。次に制御部１９はt=0に初期化する（ステップＳ１９ａ）。

【００１４】

秘密計算部１１は、少なくとも秘密分散値［x］を用い、積和の秘密計算によって関数$f_t(x)$と当該関数$f_t(x)$の近似関数$f'_u(x)$との差分$f_t(x)-f'_t(x)$の秘密分散値［$f_t(x)-f'_t(x)$］を得て出力する。ただし、$[f'_0(x)]=c_{0,0}+c_{0,1}[x]$であり、u=1,…,n-1について$[f'_u(x)]=c_{u,0}+c_{u,1}[x]+c_{u,2}[f_0(x)]+\cdots+[f_{u-1}(x)]$である。例えば、t=0のときには、秘密計算部１１は秘密分散値［x］と関数$f_0(x)$と$c_{0,0},c_{0,1}$を用いて秘密分散値［$f_0(x)-f'_0(x)$］を得る。t=1,…,n-1のときには、秘密計算部１１は秘密分散値［x］と［$f_0(x)$］,…,［$f_t(x)$］と$c_{0,0},c_{0,1},\cdots,c_{0,t+1}$と用いて秘密分散値［$f_t(x)-f'_t(x)$］を得る（ステップＳ１１）。

【００１５】

秘密分散値［$f_t(x)-f'_t(x)$］は秘密計算部１２に入力される。秘密計算部１２は、秘密分散値［$f_t(x)-f'_t(x)$］を用いた秘密計算によって$f_t(x)-f'_t(x)$を所定ビット数だけ右シフトした$(f_t(x)-f'_t(x))_r$の秘密分散値［$f_t(x)-f'_t(x)$］$_r$を得て出力する。右シフトの秘密計算は除算の秘密演算によって実現でできる。これによって$f_t(x)-f'_t(x)$の小数点位置を所定の桁まで下げる。この小数点位置は予め定められている（ステップＳ１２）。

【００１６】

秘密分散値［$f_t(x)-f'_t(x)$］$_r$は秘密計算部１３に入力される。秘密計算部１３は、秘密分散値［$f_t(x)-f'_t(x)$］$_r$と秘密分散値［$f'_t(x)$］を用いた秘密計算によって関数$f_t(x)$の秘密分散値［$f_t(x)$］を得て出力する。すなわち、秘密計算部１３は、秘密分散値［$f_t(x)-f'_t(x)$

]$_r$と秘密分散値［f'$_t$(x)］を用いた加算の秘密計算によって、$f_t(x)-f'_t(x)+f'_t(x)=f_t(x)$の秘密分散値［$f_t(x)$］を得る（ステップＳ１３）。

【００１７】

制御部１９はt=n-1であるかを判定する（ステップＳ１９ｂ）。t=n-1でなければ、制御部１９はt+1を新たなtとして処理をステップＳ１１に戻す（ステップＳ１９ｃ）。一方、t=n-1であれば、秘密計算部１３は秘密分散値［$f_{n-1}(x)$］を出力する（ステップＳ１９ｄ）。すなわち、秘密計算装置１は、t=0,…,n-2について、秘密計算部１１～１３のステップＳ１１～Ｓ１３の処理を実行するたびに、t+1を新たなtとして、ステップＳ１１～Ｓ１３の処理を再び実行し、秘密分散値［$f_{n-1}(x)$］を得る。

【００１８】

［第２実施形態］

図１に例示するように、第２実施形態の秘密計算装置２は、秘密計算部２１，２２，２３、および制御部１９を有する。第２実施形態の秘密計算装置２は、実数xの秘密分散値［x］∈［Ｌ，Ｒ）を入力とし、秘密計算を行って目的の関数$f_{n-1}(x)$の秘密分散値［$f_{n-1}(x)$］を出力する。第２実施形態では、n=3であり、a, b, c, d, f, g, h, i, j, k, s, m, n, o, p, q, α，β，γ，δ，ζが実数であり、$f_0(x)=y=\delta x^2+ax$であり、$f_1(x)=z=y(\zeta y+b)+cx$であり、$f_2(x)=w=\gamma(z(\alpha z+d)+y(\beta x+f)+gx)$であり、$f'_0(x)=ix+j$であり、$f'_1(x)=ky+sx+m$であり、$f'_2(x)=nz+oy+px+q$である例を説明する。なお、近似関数$f'_0(x)=ix+j$，$f'_1(x)=ky+sx+m$，$f'_2(x)=nz+oy+px+q$の設定方法および具体例については後述する。

【００１９】

図３に例示するように、まず秘密計算装置２の秘密計算部２１に秘密分散値［x］が入力される（ステップＳ１０）。

【００２０】

秘密計算部２１は、秘密分散値［x］を用いた積和演算の秘密計算によって秘密分散値［$f_0(x)-f'_0(x)$］=[y']=[x(δx+a-i)-j]を得て出力する（ステップＳ２１ａ）。

【００２１】

秘密分散値[y']は秘密計算部２２に入力される。秘密計算部２２は、秘密分散値[y']を用いた秘密計算によってy'を所定ビット数だけ右シフトしたy'の秘密分散値［y'］$_r$を得て出力する（ステップＳ２２ａ）。

【００２２】

秘密分散値［y'］$_r$は秘密計算部２３に入力される。秘密計算部２３は、秘密分散値［y'］$_r$と秘密分散値［f'$_0$(x)］=［ix+j］を用いた秘密計算によって秘密分散値[y]=［y'+(ix+j)］を得て出力する（ステップＳ２３ａ）。

【００２３】

秘密分散値[y]は秘密計算部２１に入力される。秘密計算部２１は、秘密分散値[x]および秘密分散値[y]を用いた積和演算の秘密計算によって秘密分散値［$f_1(x)-f'_1(x)$］=[z']=[y(ζy+b-k)+(c-s)x-m]を得て出力する（ステップＳ２１ｂ）。

【００２４】

秘密分散値[z']は秘密計算部２２に入力される。秘密計算部２２は、秘密分散値[z']を用いた秘密計算によってz'を所定ビット数だけ右シフトしたz'の秘密分散値［z'］$_r$を得て出力する（ステップＳ２２ｂ）。

【００２５】

秘密分散値［z'］$_r$は秘密計算部２３に入力される。秘密計算部２３は、秘密分散値［z'］$_r$と秘密分散値［f'$_1$(x)］=[ky+sx+m]を用いた秘密計算によって秘密分散値[z]=［z'+(ky+sx+m)］を得て出力する（ステップＳ２３ｂ）。

【００２６】

秘密分散値[z]は秘密計算部２１に入力される。秘密計算部２１は、秘密分散値［x］、秘密分散値［y］、および秘密分散値[z]を用いた積和演算の秘密計算によって秘密分散値［w'/γ］=[z(αz+d-n/γ)+(βx+f-o/γ)y+(g-p)x+(h-q)/γ]を得て出力する（ステップＳ２１ｃ）。

【００２７】
秘密分散値［w’/γ］は秘密計算部２２に入力される。秘密計算部２２は、秘密分散値［w’/γ］を用いた秘密計算によってw’/γにγを乗算して得られるw’を所定ビット数だけ右シフトしたw’の秘密分散値［w’］$_r$を得て出力する（ステップＳ２２ｃ）。秘密分散値［w’］$_r$を得るための処理に限定は無いが、例えば、秘密計算部２２は、公開値$2^{\sigma}/\gamma$を得、公開値$2^{\sigma}/\gamma$と秘密分散値［w’/γ］を用いた公開値除算の秘密計算[w’/γ]/($2^{\sigma}/\gamma$)によって秘密分散値［w’］$_r$を得てもよい。ただし、σは右シフト量を表す正整数である。これによってγの乗算および右シフトの秘密計算を同時に実行できるため、処理コストを低減できる。

【００２８】
秘密分散値［w’］$_r$は秘密計算部２３に入力される。秘密計算部２３は、秘密分散値［w’］$_r$と秘密分散値［$f'_2(x)$］=[nz+oy+px+q]を用いた秘密計算によって秘密分散値[w]=［w’+(nz+oy+px+q)］を得て出力する。

【００２９】
＜近似関数の探索方法の例示＞
以下に右シフト前の近似関数の探索方法を例示する。
入力:区間[L,R）、関数$y=\delta x^2+ax$，$z=y(\zeta y+b)+cx$，$w=\gamma(z(\alpha z+d)+y(\beta x+f)+gx)$
設定済みパラメータ:各離散係数i, k, s, n, o, pの探索最小値i_{min}, k_{min}, s_{min}, n_{min}, o_{min}, p_{min}、各離散係数i, k, s, n, o, pの探索最大値i_{max}, k_{max}, s_{max}, n_{max}, o_{max}, p_{max}
出力: yの近似関数ix+j、y-(ix+j)の最大値M_y、zの近似関数ky+sx+m、z-(ky+sx+m)の最大値M_z、wの近似関数nz+oy+px+q、w-(nz+oy+px+q)の最大値M_w

【００３０】
1: for i=i_{min} to i_{max} do
2: y-ixの区間[L,R）における最大値と最小値の差を計算する。
3: y-ixの区間[L,R）における最大値と最小値の差が最も小さいiと、そのときの差y-ixの最小値j, 差分M_y（（y-ixの最大値）－（y-ixの最小値）、言い換えるとy-ixの関数値の動く幅）を出力する。
4: for each (k, s)∈{k_{min},…,k_{max}}×{s_{min},…,s_{max}}do
5: z-(ky+sx)の区間[L,R）における最大値と最小値の差を計算する。
6: z-(ky+sx)の区間[L,R）における最大値と最小値の差が最も小さい(k, s)とそのとき差z-(ky+sx)の最小値m, 差分M_z（（z-(ky+sx)の最大値）－（z-(ky+sx)の最小値）、言い換えるとz-(ky+sx)の関数値の動く幅）を出力する。
7: for each (n, o, p) ∈ {n_{min},…,n_{max}}×{o_{min},…,o_{max}}×{p_{min},…,p_{max}} do
8: z-(nz+oy+px)の区間[L,R）における最大値と最小値の差を計算する。
9: z-(nz+oy+px)の区間[L,R）における最大値と最小値の差が最も小さい(n, o, p)とそのとき差z-(nz+oy+px)の最小値q,差分M_w（（z-(nz+oy+px)の最大値）－（z-(nz+oy+px)の最小値）、言い換えるとz-(nz+oy+px)の関数値の動く幅）を出力する。

【００３１】
［第３実施形態］
第３実施形態に例示するように、第３実施形態の秘密計算装置３は、秘密計算部３１，３２，３３、および制御部１９を有する。第３実施形態の秘密計算装置３は、実数xの秘密分散値［x］∈［Ｌ，Ｒ）を入力とし、秘密計算を行って目的の関数$f_{n-1}(x)$の秘密分散値［$f_{n-1}(x)$］を出力する。第３実施形態では、n=2であり、a, b, c，γ，δ，i, j, k, s, mが実数であり、$f_0(x)=y=\delta x^2+ax$であり、$f_1(x)=z=\gamma(y(\delta y+b)+cx)$であり、$f'_0(x)=ix+j$であり、$f'_1(x)=ky+sx+m$である例を説明する。

【００３２】
図４に例示するように、まず秘密計算装置３の秘密計算部３１に秘密分散値［x］が入力される（ステップＳ１０）。

【００３３】
秘密計算部３１は、秘密分散値［x］を用いた積和演算の秘密計算によって秘密分散値［

$f_0(x)-f'_0(x)]$ =[y’]=[x(δx+a-i)-j]を得て出力する（ステップＳ２１ａ）。

【００３４】

秘密分散値[y’]は秘密計算部３２に入力される。秘密計算部３２は、秘密分散値[y’]を用いた秘密計算によってy’を所定ビット数だけ右シフトしたyの秘密分散値$[y']_r$を得て出力する（ステップＳ２２ａ）。

【００３５】

秘密分散値$[y']_r$は秘密計算部３３に入力される。秘密計算部３３は、秘密分散値$[y']_r$と秘密分散値$[f'_0(x)]$ = [ix+j] を用いた秘密計算によって秘密分散値[y]= [y’+(ix+j)] を得て出力する（ステップＳ２３ａ）。

【００３６】

秘密分散値[y]は秘密計算部３１に入力される。秘密計算部３１は、秘密分散値[x]および秘密分散値[y]を用いた積和演算の秘密計算によって秘密分散値[z’/γ]=[y(ζy+b-k/γ)+(c-s/γ)x-m/γ]を得て出力する（ステップＳ３１ｃ）。

【００３７】

秘密分散値[z’/γ]は秘密計算部３２に入力される。秘密計算部３２は、秘密分散値$[z'/\gamma]$を用いた秘密計算によってz’/γにγを乗算して得られるz’を所定ビット数だけ右シフトしたz’の秘密分散値$[z']_r$を得て出力する（ステップＳ３２ｂ）。秘密分散値$[z']_r$を得るための処理に限定は無いが、例えば、秘密計算部３２は、公開値$2^{\sigma}/\gamma$を得、公開値$2^{\sigma}/\gamma$と秘密分散値$[z'/\gamma]$を用いた公開値除算の秘密計算$[z'/\gamma]/(2^{\sigma}/\gamma)$によって秘密分散値$[z']_r$を得てもよい。これによってγの乗算および右シフトの秘密計算を同時に実行できるため、処理コストを低減できる。

【００３８】

秘密分散値$[z']_r$は秘密計算部３３に入力される。秘密計算部３３は、秘密分散値$[z']_r$と秘密分散値$[f'_1(x)]$ =[ky+sx+m]を用いた秘密計算によって秘密分散値[z]= [z’+(ky+sx+m)] を得て出力する（ステップＳ３３ｂ）。

【００３９】

［各初等関数に関する計算済みのパラメータの例］

図５に関数$f_{n-1}(x)$が初等関数である逆数関数、平方根関数、平方根の逆数関数、指数関数、対数関数である場合の計算済みのパラメータを例示する。なお、ex,ey,ezはそれぞれx,y,zの小数点位置を示す。また、e'x,e'y,e'zはそれぞれ右シフト前のx',y',z'の小数点位置を示す。これらの小数点位置は、下位ビットから数えた小数点位置のビット位置を表す。このビット位置を表す値は０から始まり、下位ビットから数えてｅ１ビット目が１を表すときに、小数点位置がｅ１であると表記する。

【００４０】

［ハードウェア構成］

各実施形態における秘密計算装置１，２，３は、例えば、ＣＰＵ（central processing unit）等のプロセッサ（ハードウェア・プロセッサ）やＲＡＭ（random-access memory）・ＲＯＭ（read-only memory）等のメモリ等を備える汎用または専用のコンピュータが所定のプログラムを実行することで構成される装置である。このコンピュータは１個のプロセッサやメモリを備えていてもよいし、複数個のプロセッサやメモリを備えていてもよい。このプログラムはコンピュータにインストールされてもよいし、予めＲＯＭ等に記録されていてもよい。また、ＣＰＵのようにプログラムが読み込まれることで機能構成を実現する電子回路（circuitry）ではなく、単独で処理機能を実現する電子回路を用いて一部またはすべての処理部が構成されてもよい。また、１個の装置を構成する電子回路が複数のＣＰＵを含んでいてもよい。

【００４１】

図６は、各実施形態における秘密計算装置１，２，３のハードウェア構成を例示したブロック図である。図６に例示するように、この例の秘密計算装置１，２，３は、ＣＰＵ（Central Processing Unit）１０ａ、入力部１０ｂ、出力部１０ｃ、ＲＡＭ（Random Access Memory）１０ｄ、ＲＯＭ（Read Only Memory）１０ｅ、補助記憶装置１０ｆ

及びバス１０ｇを有している。この例のＣＰＵ１０ａは、制御部１０ａａ、演算部１０ａｂ及びレジスタ１０ａｃを有し、レジスタ１０ａｃに読み込まれた各種プログラムに従って様々な演算処理を実行する。また、出力部１０ｃは、データが出力される出力端子、ディスプレイ等、所定のプログラムを読み込んだＣＰＵ１０ａによって制御されるＬＡＮカード等である。また、ＲＡＭ１０ｄは、ＳＲＡＭ (Static Random Access Memory)、ＤＲＡＭ (Dynamic Random Access Memory)等であり、所定のプログラムが格納されるプログラム領域１０ｄａ及び各種データが格納されるデータ領域１０ｄｂを有している。また、補助記憶装置１０ｆは、例えば、ハードディスク、ＭＯ（Magneto-Optical disc）、半導体メモリ等であり、所定のプログラムが格納されるプログラム領域１０ｆａ及び各種データが格納されるデータ領域１０ｆｂを有している。また、バス１０ｇは、ＣＰＵ１０ａ、入力部１０ｂ、出力部１０ｃ、ＲＡＭ１０ｄ、ＲＯＭ１０ｅ及び補助記憶装置１０ｆを、情報のやり取りが可能なように接続する。ＣＰＵ１０ａは、読み込まれたＯＳ（Operating System）プログラムに従い、補助記憶装置１０ｆのプログラム領域１０ｆａに格納されているプログラムをＲＡＭ１０ｄのプログラム領域１０ｄａに書き込む。同様にＣＰＵ１０ａは、補助記憶装置１０ｆのデータ領域１０ｆｂに格納されている各種データを、ＲＡＭ１０ｄのデータ領域１０ｄｂに書き込む。そして、このプログラムやデータが書き込まれたＲＡＭ１０ｄ上のアドレスがＣＰＵ１０ａのレジスタ１０ａｃに格納される。ＣＰＵ１０ａの制御部１０ａｂは、レジスタ１０ａｃに格納されたこれらのアドレスを順次読み出し、読み出したアドレスが示すＲＡＭ１０ｄ上の領域からプログラムやデータを読み出し、そのプログラムが示す演算を演算部１０ａｂに順次実行させ、その演算結果をレジスタ１０ａｃに格納していく。このような構成により、秘密計算装置１，２，３の機能構成が実現される。

【００４２】

上述のプログラムは、コンピュータで読み取り可能な記録媒体に記録しておくことができる。コンピュータで読み取り可能な記録媒体の例は非一時的な（non-transitory）記録媒体である。このような記録媒体の例は、磁気記録装置、光ディスク、光磁気記録媒体、半導体メモリ等である。

【００４３】

このプログラムの流通は、例えば、そのプログラムを記録したＤＶＤ、ＣＤ－ＲＯＭ等の可搬型記録媒体を販売、譲渡、貸与等することによって行う。さらに、このプログラムをサーバコンピュータの記憶装置に格納しておき、ネットワークを介して、サーバコンピュータから他のコンピュータにそのプログラムを転送することにより、このプログラムを流通させる構成としてもよい。上述のように、このようなプログラムを実行するコンピュータは、例えば、まず、可搬型記録媒体に記録されたプログラムもしくはサーバコンピュータから転送されたプログラムを、一旦、自己の記憶装置に格納する。そして、処理の実行時、このコンピュータは、自己の記憶装置に格納されたプログラムを読み取り、読み取ったプログラムに従った処理を実行する。また、このプログラムの別の実行形態として、コンピュータが可搬型記録媒体から直接プログラムを読み取り、そのプログラムに従った処理を実行することとしてもよく、さらに、このコンピュータにサーバコンピュータからプログラムが転送されるたびに、逐次、受け取ったプログラムに従った処理を実行することとしてもよい。また、サーバコンピュータから、このコンピュータへのプログラムの転送は行わず、その実行指示と結果取得のみによって処理機能を実現する、いわゆるＡＳＰ（Application Service Provider）型のサービスによって、上述の処理を実行する構成としてもよい。なお、本形態におけるプログラムには、電子計算機による処理の用に供する情報であってプログラムに準ずるもの（コンピュータに対する直接の指令ではないがコンピュータの処理を規定する性質を有するデータ等）を含むものとする。

【００４４】

各実施形態では、コンピュータ上で所定のプログラムを実行させることにより、本装置を構成することとしたが、これらの処理内容の少なくとも一部をハードウェア的に実現することとしてもよい。

【００４５】
＜その他の変形例等＞
なお、本発明は上述の実施の形態に限定されるものではない。例えば、実施形態の秘密計算装置１，２，３は、実数xの秘密分散値［x］を用いた秘密計算によって$f_t(x)-f'_t(x)$の秘密分散値［$f_t(x)-f'_t(x)$］を得、秘密分散値［$f_t(x)-f'_t(x)$］を用いた秘密計算によって$f_t(x)-f'_t(x)$を所定ビット数だけ右シフトした$(f_t(x)-f'_t(x))_r$の秘密分散値［$f_t(x)-f'_t(x)$］$_r$を得、秘密分散値［$f_t(x)-f'_t(x)$］$_r$と秘密分散値［$f'_t(x)$］を用いた秘密計算によって関数$f_t(x)$の秘密分散値［$f_t(x)$］を得ていた。しかしながら、秘密分散値［$f_t(x)$］を得る前に秘密分散値［$f_t(x)-f'_t(x)$］$_r$が別の秘密計算に用いられてもよい。
【００４６】
上記の実施形態では、秘密計算部１１が秘密分散値［x］を用いた積和演算の秘密計算によって秘密分散値［$f_t(x)-f'_t(x)$］を得ていたが、積和演算の秘密計算以外の秘密計算によって秘密分散値［$f_t(x)-f'_t(x)$］を得てもよい。
【００４７】
また、上述の各種の処理は、記載に従って時系列に実行されるのみならず、処理を実行する装置の処理能力あるいは必要に応じて並列的にあるいは個別に実行されてもよい。その他、本発明の趣旨を逸脱しない範囲で適宜変更が可能であることはいうまでもない。
【産業上の利用可能性】
【００４８】
本発明は、例えば、データを秘匿化しつつ秘密計算で行う機械学習やデータマイニングでの逆数関数、平方根関数、指数関数、対数関数などの初等関数の計算に利用できる。
【符号の説明】
【００４９】
１，２，３　秘密計算装置
１１，２１，３１，１２，２２，３２，１３，２３，３３　秘密計算部

【図面】

【図１】

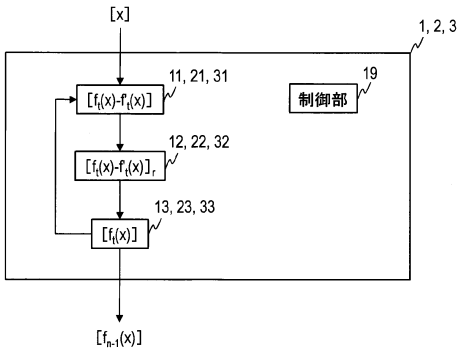

図1

【図２】

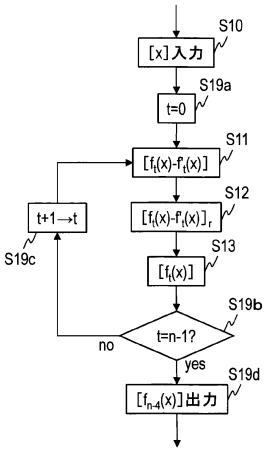

図2

【図３】

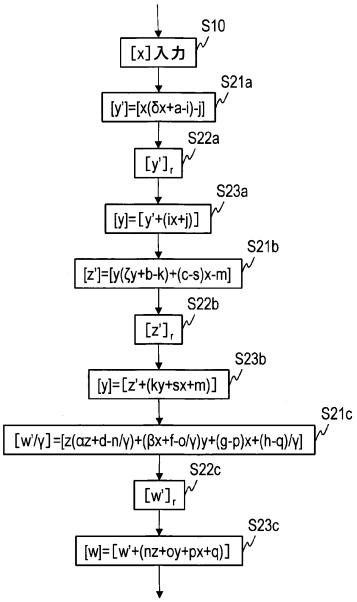

図3

【図４】

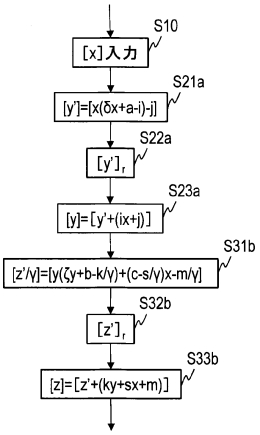

図4

【図５】

図5

パラメータ	逆数	平方根	平方根の逆数	指数	対数
l	0.75	1	0.5	0	2
R	1.125	2	1	2	4
a	-2.110464393923928	-0.4281414400291061	-1.696628555353	0.01502453639009133	-0.8466962732121107
b	2.434960779692772	0.4101200793876874	1.549574093768	0.4098327753215 8	0.7847029513335529
c	-0.18413217288124 9	-0.01203095843274 84	-0.11015688365438 4	0.2185722478671 26	-0.0353259366062005
d	9.889932879695 75	-3.717956726390 17	4.697457088531 76	6.648269557208433	-3.026502814095 83
f	1.843813235555315	-0.643779936629 56	0.910418285014127	-0.737980772377752	-0.5226874847 4532
g	-2.074844182182 56	0.447097638921 85	-0.959053601554654	-0.580955576157224	0.3780821789024 87
h	9.792700357955 59	0.232903741490693	3.971290599099 28	1.000000030026 2	-1.687553742176 25
i	0	0	-0.25	0.5	0
j	-1.113514988782 71	-0.366610117286381	-0.523183544290677	-0.470402400605697	-1.433789157834 34
k	1	0	0	1	0
s	0	0	-0.25	0	0
m	-0.568915141305 53	-0.106711930503672	-0.503025809551099	0	-0.748724878178412
n	16	0	0	20	0
o	-4	0	0	0	0
p	2	0	0	-4	0
q	11.24208874577 71	0.765	-2.971290599099 28	-0.105107110464577	2.38
α	2.6875	-14.25	3	3.875	-1.75
β	-0.90625	0.125	-0.5	0.4375	0.0625
y^-1	1.378714399100087	1.074207336578 23	1.031634745737 52	0.901983541508 68	1.033061782443 71
δ	1	2^-3	1	2^-3	2^-3
ζ	1	2^-1	1	2^-2	2^-2
ex	28	28	28	28	27
ey	29	29	30	29	28
ez	29	29	30	29	28
ew	27	28	28	27	28
e'y	61	62	62	60	61
e'x	63	64	63	61	62
e'w	63	60	60	61	59

【図６】

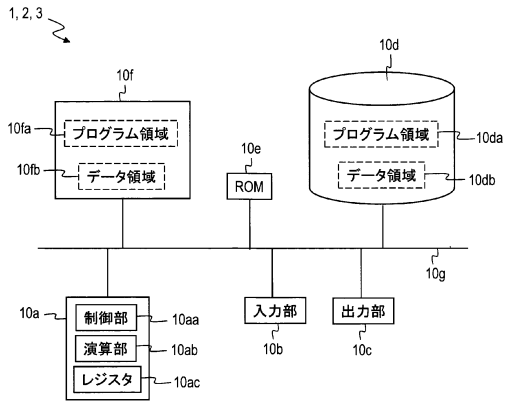

図6

フロントページの続き

(56)参考文献 特表２０２０－５２５８１４（ＪＰ，Ａ）
大畑 幸矢，秘匿深層学習再考，２０１８年 暗号と情報セキュリティシンポジウム（ＳＣＩＳ２０１８）予稿集 ［ＵＳＢ］，日本，2018年01月23日，3F-1，p. 1-8
CATRINA, Octavian et al.，Secure Computation With Fixed-Point Numbers，LNCS, Financial Cryptography and Data Security，Vol. 6052，ドイツ，Springer，2010年，p. 35-50
五十嵐 大，秘密計算上の理論最適な単精度関数近似法，２０２０年 暗号と情報セキュリティシンポジウム，日本，2020年01月21日，2C3-1，p. 1-8
天田 拓磨 ほか，浮動小数点演算のための通信量を削減したマルチパーティ計算，２０１８年 暗号と情報セキュリティシンポジウム（ＳＣＩＳ２０１８）予稿集 ［ＵＳＢ］，日本，2018年01月23日，2A2-2，p. 1-8
三品 気吹 ほか，高精度かつ高効率な秘密ロジスティック回帰の設計と実装，ＣＳＳ２０１８ コンピュータセキュリティシンポジウム２０１８論文集，日本，一般社団法人情報処理学会，2018年10月25日，p. 1229-1236

(58)調査した分野 (Int.Cl.，ＤＢ名)
Ｇ０９Ｃ １／００
ＪＳＴＰｌｕｓ／ＪＭＥＤＰｌｕｓ／ＪＳＴ７５８０（ＪＤｒｅａｍＩＩＩ）
ＩＥＥＥ Ｘｐｌｏｒｅ
ＴＨＥ ＡＣＭ ＤＩＧＩＴＡＬ ＬＩＢＲＡＲＹ