(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6605746号
(P6605746)
(45)発行日　令和1年11月13日(2019.11.13)　(24)登録日　令和1年10月25日(2019.10.25)

(51)Int.Cl.　F I
G09C 1/00 (2006.01)　G09C 1/00 650Z

請求項の数 8　(全 24 頁)

(21)出願番号　特願2018-542380(P2018-542380)
(86)(22)出願日　平成29年9月14日(2017.9.14)
(86)国際出願番号　PCT/JP2017/033283
(87)国際公開番号　WO2018/061800
(87)国際公開日　平成30年4月5日(2018.4.5)
審査請求日　平成31年3月5日(2019.3.5)
(31)優先権主張番号　特願2016-187782(P2016-187782)
(32)優先日　平成28年9月27日(2016.9.27)
(33)優先権主張国・地域又は機関
日本国(JP)

(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
(72)発明者　桐淵　直人
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

最終頁に続く

(54)【発明の名称】秘密等結合システム、秘密等結合装置、秘密等結合方法、プログラム

(57)【特許請求の範囲】
【請求項１】
Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数、a，bを2以上の整数、p_i(1≦i≦m，ただし、p_1,..，p_mは互いに異なる)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)を0でない有限環Z_Nの要素とし、
[[x]]をx∈Z_Nを秘匿した値、<π>を秘密計算による置換πを示すものとし、
3個以上の秘密等結合装置で構成され、各要素が秘匿されているm行a列の表L、n行b列の表Rからn行a+b-1列の表Jを生成する秘密等結合システムであって、
前記表Lの第１列([[p_1]],..，[[p_m]])、前記表Rの第１列([[q_1]],..，[[q_n]])から生成される要素列([[p_1]],..，[[p_m]]，[[q_1]],..，[[q_n]]，[[p_1]],..，[[p_m]])を安定ソートすることで置換<σ>を生成する第一置換生成手段と、
j=2,..，aに対して、
(1) 前記表Lの第j列([[$v_{1,j}$]],..，[[$v_{m,j}$]])と[[0]]をn個並べた要素列([[0]],..，[[0]])を用いて要素列[[f]]=([[$v_{1,j}$]],..，[[$v_{m,j}$]]，[[0]],..，[[0]]，[[$-v_{1,j}$]],..，[[$-v_{m,j}$]])を生成し、
(2) 前記置換<σ>を用いて前記要素列[[f]]から要素列[[g]]=[[σ([[f]])]]を生成し、
(3) 前記要素列[[g]]のプリフィックスサムを計算することにより、要素列[[g’]]=PrefixSum([[g]])を生成し、
(4) 前記置換<σ>の逆置換<$σ^{-1}$>を用いて前記要素列[[g’]]から要素列[[f’]]=[[$σ^{-1}$([[g’]])]]を生成し、

(5) 前記要素列[[f’]]の第m+1要素から第m+n要素までの部分要素列([[f’$_{m+1}$]],..,[[f’$_{m+n}$]])を取り出し、前記表Jの第j列([[v’$_{1,j}$]],..,[[v’$_{n,j}$]])=([[f’$_{m+1}$]],..,[[f’$_{m+n}$]])を生成することにより、前記表Jの第2列から第a列を生成する第一列生成手段と、

(1) m個の[[1]]からなる要素列([[1]],..,[[1]])とn個の[[0]]からなる要素列([[0]],..,[[0]])を用いて要素列[[f1]]=([[1]],..,[[1]],[[0]],..,[[0]],[[-1]],..,[[-1]])を生成し、

(2) 前記置換<σ>を用いて前記要素列[[f1]]から要素列[[g1]]=[[σ([[f1]])]]を生成し、

(3) 前記要素列[[g1]]のプリフィックスサムを計算することにより、要素列[[g1’]]=PrefixSum([[g1]])を生成し、

(4) 前記逆置換<σ^{-1}>を用いて前記要素列[[g1’]]から要素列[[f1’]]=[[σ^{-1}([[g1’]])]]を生成し、

(5) 前記要素列[[f1’]]の第m+1要素から第m+n要素までの部分要素列([[f1’$_{m+1}$]],..,[[f1’$_{m+n}$]])を取り出し、結合結果要素列([[e_1]],..,[[e_n]])=([[f1’$_{m+1}$]],..,[[f1’$_{m+n}$]])を生成する結合結果要素列生成手段と、

j=a+1,..,a+b-1に対して、前記結合結果要素列([[e_1]],..,[[e_n]])と前記表Rの第j-a+1列([[$u_{1,j-a+1}$]],..,[[$u_{m,j-a+1}$]])を用いて前記表Jの第j列([[u’$_{1,j-a+1}$]],..,[[u’$_{n,j-a+1}$]])=([[e_1]]×[[$u_{1,j-a+1}$]],..,[[e_n]]×[[$u_{n,j-a+1}$]])を生成することにより、前記表Jの第a+1列から第a+b-1列を生成する第二列生成手段と、

前記結合結果要素列([[e_1]],..,[[e_n]])と前記表Rの第１列([[q_1]],..,[[q_n]])を用いて前記表Jの第1列([[q’$_1$]],..,[[q’$_n$]])=([[e_1]]×[[q_1]],..,[[e_n]]×[[q_n]])を生成する第三列生成手段と、

を含む秘密等結合システム。

【請求項２】

Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数、a，bを2以上の整数、p_i(1≦i≦m，ただし、p_1,..,p_mは互いに異なる)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)を0でない有限環Z_Nの要素とし、

[[x]]をx∈Z_Nを秘匿した値、<π>を秘密計算による置換πを示すものとし、

3個以上の秘密等結合装置で構成され、各要素が秘匿されているm行a列の表L、n行b列の表Rから請求項１に記載の秘密等結合システムを用いてn行a+b-1列の表J’を生成する秘密等結合システムであって、

前記結合結果要素列([[e_1]],..,[[e_n]])を安定ソートすることで置換<σ~>を生成する第二置換生成手段と、

前記置換<σ~>を用いて、前記表Jの第1列([[q’$_1$]],..,[[q’$_n$]])から前記表J’の第1列([[q’’$_1$]],..,[[q’’$_n$]])=[[σ~([[q’$_1$]],..,[[q’$_n$]])]]を生成し、j=2,..,aに対して前記表Jの第j列([[v’$_{1,j}$]],..,[[v’$_{n,j}$]])から前記表J’の第j列([[v’’$_{1,j}$]],..,[[v’’$_{n,j}$]])=[[σ~([[v’$_{1,j}$]],..,[[v’$_{n,j}$]])]]を生成し、j=a+1,..,a+b-1に対して前記表Jの第j列([[u’$_{1,j-a+1}$]],..,[[u’$_{n,j-a+1}$]])から前記表J’の第j列([[u’’$_{1,j-a+1}$]],..,[[u’’$_{n,j-a+1}$]])=[[σ~([[u’$_{1,j-a+1}$]],..,[[u’$_{n,j-a+1}$]])]]を生成する行並び替え手段と、

を含む秘密等結合システム。

【請求項３】

Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数、a，bを2以上の整数、p_i(1≦i≦m，ただし、p_1,..,p_mは互いに異なる)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)を0でない有限環Z_Nの要素とし、

[[x]]をx∈Z_Nを秘匿した値、<π>を秘密計算による置換πを示すものとし、

3個以上の秘密等結合装置で構成され、各要素が秘匿されているm行a列の表L、n行b列の表Rから請求項２に記載の秘密等結合システムを用いてc行a+b-1列の表J’’を生成する秘密等結合システムであって、

前記結合結果要素列([[e_1]],..,[[e_n]])の各要素の和[[c]]を計算する結合行数計算手段と、

前記和[[c]]を復元して得られる前記cを公開、前記表J'の上からc行を抽出した前記表J''を生成する結合行数公開手段と、

を含む秘密等結合システム。

【請求項４】

Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数、a，bを2以上の整数、p_i(1≦i≦m，ただし、p_1,..,p_mは互いに異なる)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)を0でない有限環Z_Nの要素とし、

[[x]]をx∈Z_Nを秘匿した値、<π>を秘密計算による置換πを示すものとし、

各要素が秘匿されているm行a列の表L、n行b列の表Rからn行a+b-1列の表Jを生成する3個以上の秘密等結合装置で構成される秘密等結合システムの中の秘密等結合装置であって、

前記表Lの第１列([[p_1]],..,[[p_m]])、前記表Rの第１列([[q_1]],..,[[q_n]])から生成される要素列([[p_1]],..,[[p_m]],[[q_1]],..,[[q_n]],[[p_1]],..,[[p_m]])を安定ソートすることで置換<σ>を生成するための第一置換生成部と、

j=2,..,aに対して、

(1) 前記表Lの第j列([[$v_{1,j}$]],..,[[$v_{m,j}$]])と[[0]]をn個並べた要素列([[0]],..,[[0]])を用いて要素列[[f]]=([[$v_{1,j}$]],..,[[$v_{m,j}$]],[[0]],..,[[0]],[[$-v_{1,j}$]],..,[[$-v_{m,j}$]])を生成し、

(2) 前記置換<σ>を用いて前記要素列[[f]]から要素列[[g]]=[[σ([[f]])]]を生成し、

(3) 前記要素列[[g]]のプリフィックスサムを計算することにより、要素列[[g']]=PrefixSum([[g]])を生成し、

(4) 前記置換<σ>の逆置換<$σ^{-1}$>を用いて前記要素列[[g']]から要素列[[f']]=[[$σ^{-1}$([[g']])]]を生成し、

(5) 前記要素列[[f']]の第m+1要素から第m+n要素までの部分要素列([[f'_{m+1}]],..,[[f'_{m+n}]])を取り出し、前記表Jの第j列([[$v'_{1,j}$]],..,[[$v'_{n,j}$]])=([[f'_{m+1}]],..,[[f'_{m+n}]])を生成することにより、前記表Jの第2列から第a列を生成するための第一列生成部と、

(1) m個の[[1]]からなる要素列([[1]],..,[[1]])とn個の[[0]]からなる要素列([[0]],..,[[0]])を用いて要素列[[f1]]=([[1]],..,[[1]],[[0]],..,[[0]],[[-1]],..,[[-1]])を生成し、

(2) 前記置換<σ>を用いて前記要素列[[f1]]から要素列[[g1]]=[[σ([[f1]])]]を生成し、

(3) 前記要素列[[g1]]のプリフィックスサムを計算することにより、要素列[[g1']]=PrefixSum([[g1]])を生成し、

(4) 前記逆置換<$σ^{-1}$>を用いて前記要素列[[g1']]から要素列[[f1']]=[[$σ^{-1}$([[g1']])]]を生成し、

(5) 前記要素列[[f1']]の第m+1要素から第m+n要素までの部分要素列([[$f1'_{m+1}$]],..,[[$f1'_{m+n}$]])を取り出し、結合結果要素列([[e_1]],..,[[e_n]])=([[$f1'_{m+1}$]],..,[[$f1'_{m+n}$]])を生成するための結合結果要素列生成部と、

j=a+1,..,a+b-1に対して、前記結合結果要素列([[e_1]],..,[[e_n]])と前記表Rの第j-a+1列([[$u_{1,j-a+1}$]],..,[[$u_{m,j-a+1}$]])を用いて前記表Jの第j列([[$u'_{1,j-a+1}$]],..,[[$u'_{n,j-a+1}$]])=([[e_1]]×[[$u_{1,j-a+1}$]],..,[[e_n]]×[[$u_{n,j-a+1}$]])を生成することにより、前記表Jの第a+1列から第a+b-1列を生成するための第二列生成部と、

前記結合結果要素列([[e_1]],..,[[e_n]])と前記表Rの第１列([[q_1]],..,[[q_n]])を用いて前記表Jの第1列([[q'_1]],..,[[q'_n]])=([[e_1]]×[[q_1]],..,[[e_n]]×[[q_n]])を生成するための第三列生成部と、

を含む秘密等結合装置。

【請求項５】

Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数

、a，bを2以上の整数、p_i(1≦i≦m，ただし、p_1,..，p_mは互いに異なる)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)を0でない有限環Z_Nの要素とし、

[[x]]をx∈Z_Nを秘匿した値、<π>を秘密計算による置換πを示すものとし、

3個以上の秘密等結合装置で構成され、第一置換生成手段と第一列生成手段と結合結果要素列生成手段と第二列生成手段と第三列生成手段とを含む秘密等結合システムを用いて、各要素が秘匿されているm行a列の表L、n行b列の表Rからn行a+b-1列の表Jを生成する秘密等結合方法であって、

前記第一置換生成手段が、前記表Lの第１列([[p_1]],..，[[p_m]])、前記表Rの第１列([[q_1]],..，[[q_n]])から生成される要素列([[p_1]],..，[[p_m]]，[[q_1]],..，[[q_n]]，[[p_1]],..，[[p_m]])を安定ソートすることで置換<σ>を生成する第一置換生成ステップと、

前記第一列生成手段が、j=2,..，aに対して、

(1) 前記表Lの第j列([[$v_{1,j}$]],..，[[$v_{m,j}$]])と[[0]]をn個並べた要素列([[0]],..，[[0]])を用いて要素列[[f]]=([[$v_{1,j}$]],..，[[$v_{m,j}$]]，[[0]],..，[[0]]，[[$-v_{1,j}$]],..，[[$-v_{m,j}$]])を生成し、

(2) 前記置換<σ>を用いて前記要素列[[f]]から要素列[[g]]=[[σ([[f]])]]を生成し、

(3) 前記要素列[[g]]のプリフィックスサムを計算することにより、要素列[[g’]]=PrefixSum([[g]])を生成し、

(4) 前記置換<σ>の逆置換<$σ^{-1}$>を用いて前記要素列[[g’]]から要素列[[f’]]=[[$σ^{-1}$([[g’]])]]を生成し、

(5) 前記要素列[[f’]]の第m+1要素から第m+n要素までの部分要素列([[f'_{m+1}]],..，[[f'_{m+n}]])を取り出し、前記表Jの第j列([[$v'_{1,j}$]],..，[[$v'_{n,j}$]])=([[f'_{m+1}]],..，[[f'_{m+n}]])を生成することにより、前記表Jの第2列から第a列を生成する第一列生成ステップと、

前記結合結果要素列生成手段が、

(1) m個の[[1]]からなる要素列([[1]],..，[[1]])とn個の[[0]]からなる要素列([[0]],..，[[0]])を用いて要素列[[f1]]=([[1]],..，[[1]]，[[0]],..，[[0]]，[[-1]],..，[[-1]])を生成し、

(2) 前記置換<σ>を用いて前記要素列[[f1]]から要素列[[g1]]=[[σ([[f1]])]]を生成し、

(3) 前記要素列[[g1]]のプリフィックスサムを計算することにより、要素列[[g1’]]=PrefixSum([[g1]])を生成し、

(4) 前記逆置換<$σ^{-1}$>を用いて前記要素列[[g1’]]から要素列[[f1’]]=[[$σ^{-1}$([[g1’]])]]を生成し、

(5) 前記要素列[[f1’]]の第m+1要素から第m+n要素までの部分要素列([[$f1'_{m+1}$]],..，[[$f1'_{m+n}$]])を取り出し、結合結果要素列([[e_1]],..，[[e_n]])=([[$f1'_{m+1}$]],..，[[$f1'_{m+n}$]])を生成する結合結果要素列生成ステップと、

前記第二列生成手段が、j=a+1,..，a+b-1に対して、前記結合結果要素列([[e_1]],..，[[e_n]])と前記表Rの第j-a+1列([[$u_{1,j-a+1}$]],..，[[$u_{m,j-a+1}$]])を用いて前記表Jの第j列([[$u'_{1,j-a+1}$]],..，[[$u'_{n,j-a+1}$]])=([[e_1]]×[[$u_{1,j-a+1}$]],..，[[e_n]]×[[$u_{n,j-a+1}$]])を生成することにより、前記表Jの第a+1列から第a+b-1列を生成する第二列生成ステップと、

前記第三列生成手段が、前記結合結果要素列([[e_1]],..，[[e_n]])と前記表Rの第１列([[q_1]],..，[[q_n]])を用いて前記表Jの第1列([[q’$_1$]],..，[[q’$_n$]])=([[e_1]]×[[q_1]],..，[[e_n]]×[[q_n]])を生成する第三列生成ステップと、

を実行する秘密等結合方法。

【請求項６】

Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数、a，bを2以上の整数、p_i(1≦i≦m，ただし、p_1,..，p_mは互いに異なる)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)を0でない有限環Z_Nの要素とし、

[[x]]をx∈Z_Nを秘匿した値、<π>を秘密計算による置換πを示すものとし、

3個以上の秘密等結合装置で構成され、第一置換生成手段と第一列生成手段と結合結果要素列生成手段と第二列生成手段と第三列生成手段と第二置換生成手段と行並び替え手段とを含む秘密等結合システムを用いて、各要素が秘匿されているm行a列の表L、n行b列の表Rからn行a+b-1列の表J’を生成する秘密等結合方法であって、

請求項５に記載の秘密等結合方法により、前記表Lと前記表Rから前記表Jを生成した後、

前記第二置換生成手段が、前記結合結果要素列([[e_1]],..,[[e_n]])を安定ソートすることで置換<σ~>を生成する第二置換生成ステップと、

前記行並び替え手段が、前記置換<σ~>を用いて、前記表Jの第1列([[q'_1]],..,[[q'_n]])から前記表J’の第1列([[q''_1]],..,[[q''_n]])=[[σ~([[q'_1]],..,[[q'_n]])]]を生成し、j=2,..,aに対して前記表Jの第j列([[$v'_{1,j}$]],..,[[$v'_{n,j}$]])から前記表J’の第j列([[$v''_{1,j}$]],..,[[$v''_{n,j}$]])=[[σ~([[$v'_{1,j}$]],..,[[$v'_{n,j}$]])]]を生成し、j=a+1,..,a+b-1に対して前記表Jの第j列([[$u'_{1,j-a+1}$]],..,[[$u'_{n,j-a+1}$]])から前記表J’の第j列([[$u''_{1,j-a+1}$]],..,[[$u''_{n,j-a+1}$]])=[[σ~([[$u'_{1,j-a+1}$]],..,[[$u'_{n,j-a+1}$]])]]を生成する行並び替えステップと、

を実行する秘密等結合方法。

【請求項７】

Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数、a，bを2以上の整数、p_i(1≦i≦m，ただし、p_1,..,p_mは互いに異なる)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)を0でない有限環Z_Nの要素とし、

[[x]]をx∈Z_Nを秘匿した値、<π>を秘密計算による置換πを示すものとし、

3個以上の秘密等結合装置で構成され、第一置換生成手段と第一列生成手段と結合結果要素列生成手段と第二列生成手段と第三列生成手段と第二置換生成手段と行並び替え手段と結合行数計算手段と結合行数公開手段とを含む秘密等結合システムを用いて、各要素が秘匿されているm行a列の表L、n行b列の表Rからc行a+b-1列の表J’’を生成する秘密等結合方法であって、

請求項６に記載の秘密等結合方法により、前記表Lと前記表Rから前記表J’を生成した後、

前記結合行数計算手段が、前記結合結果要素列([[e_1]],..,[[e_n]])の各要素の和[[c]]を計算する結合行数計算ステップと、

前記結合行数公開手段が、前記和[[c]]を復元して得られる前記cを公開、前記表J’の上からc行を抽出した前記表J’’を生成する結合行数公開ステップと、

を実行する秘密等結合方法。

【請求項８】

請求項１ないし３のいずれか１項に記載の秘密等結合システムを構成する秘密等結合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、秘密計算によって、表に含まれる情報を秘密にしたまま、２つの表に共通のキー属性を鍵として２つの表の等結合を行う等結合技術に関する。

【背景技術】

【０００２】

暗号化された数値を復元することなく指定された演算の演算結果を得る方法として、秘密計算と呼ばれる方法がある（例えば非特許文献１参照）。非特許文献１の方法では、数値を復元することのできる複数の情報を3つの秘密計算装置に分散するという暗号化を行い、数値を復元することなく、加減算、定数和、乗算、定数倍、論理演算（否定、論理積、論理和、排他的論理和）、データ形式変換（整数、二進数）の結果を3つの秘密計算装置に分散された状態、すなわち暗号化されたまま保持させることができる。一般に、分散数は3に限らずW（Wは3以上の所定の定数）とすることができ、W個の秘密計算装置による

協調計算によって秘密計算を実現するプロトコルはマルチパーティプロトコルと呼ばれる。

【０００３】

ところで、表に対するデータベース処理では多くの場合、データは複数の属性値（属性に対応する値であり、図１Ａ、図１Ｂ、図１Ｃの表の例では属性であるNo.、身長、体重、購入品のそれぞれの具体的な値“３”,“200”,“100”,“おいしい水”などである）の組からなるレコード（図１Ａや図１Ｂに例示される表の各行のこと）の集合からなる表単位で管理される。データベース処理で重要な処理の一つに等結合がある。等結合は、図１Ａや図１Ｂのような複数の表を入力とし、キーと呼ばれる属性（キー属性）の値（キー属性値）がすべての表で共通のレコードを抜き出して、これらを横に並べた新しい表を得る計算である。例えば、図１Ａの表L_sや図１Ｂの表R_sを、各表に共通のキー属性（この例ではNo.）を基準として等結合を行うと図１Ｃのような表J_sが得られる。関係データベースではデータを多くの小さな表に分割して管理し、利用時に必要な表を等結合して処理を行うことが一般的であり、等結合は非常に重要な処理である。

【０００４】

秘密計算によって表の等結合を実現した方法として、非特許文献２の方法がある。非特許文献２の方法では、キー属性値に重複がある複数の表の等結合を実現している。

【先行技術文献】

【非特許文献】

【０００５】

【非特許文献１】千田浩司，濱田浩気，五十嵐大，高橋克巳，“軽量検証可能３パーティ秘匿関数計算の再考”，In CSS，2010.

【非特許文献２】濱田浩気，桐淵直人，五十嵐大，“キーに重複がある場合の秘密計算向け結合アルゴリズム”，暗号と情報セキュリティシンポジウム(SCIS)2015，電子情報通信学会，2015.

【発明の概要】

【発明が解決しようとする課題】

【０００６】

秘密計算がマルチパーティプロトコルで実現されるものとして計算効率の評価を行う場合、マルチパーティプロトコルは複数のパーティ（参加者）間で通信を行いながら協調計算を行う方式であり、一般的なシステム構成では各パーティが単独で行うローカルの計算に比べて通信に要する時間が著しく大きいので、ローカルの計算は無視できるものとみなせる。したがって、通信したデータの量（通信量）の尺度で計算効率の評価を行う。

【０００７】

このとき、非特許文献２の方法では、結合する２つの表の行数をそれぞれm，n、結合するキー属性の重複する要素の最大数をkとすると、$O(k(m+n)\log(m+n))$の通信が必要となり、等結合を行う際にデータを秘匿して格納したサーバ間で必要となる通信が多くなってしまうという問題があった。特に、先ほどの図１Ｂの属性“購入品”のようにキー属性“No.”に対して何度も何度も出てくる可能性がある場合、kの値は大きいものとなり問題が顕在化することとなる。

【０００８】

そこで本発明は、秘密計算によって表に含まれる情報を秘密にしたまま、通信量を抑えつつ、秘密にされた２つの表から秘密にされた１つの表を生成する等結合技術を提供することを目的とする。

【課題を解決するための手段】

【０００９】

本発明の一態様は、Z_Nを0からNまでの整数の集合（Nは１以上の整数）からなる有限環、m，nを1以上の整数、a，bを2以上の整数、$p_i(1\leqq i\leqq m)$，$v_{i,j}(1\leqq i\leqq m,\ 2\leqq j\leqq a)$，$q_i(1\leqq i\leqq n)$，$u_{i,j}(1\leqq i\leqq n,\ 2\leqq j\leqq b)$を0でない有限環$Z_N$の要素とし、$[[x]]$をx∈$Z_N$を秘匿した値、$<\pi>$を秘密計算による置換πを示すものとし、3個以上の秘密等結合装置で構成

され、各要素が秘匿されているm行a列の表L、n行b列の表Rからn行a+b-1列の表Jを生成する秘密等結合システムであって、前記表Lの第１列([[p_1]],..,[[p_m]])、前記表Rの第１列([[q_1]],..,[[q_n]])から生成される要素列([[p_1]],..,[[p_m]],[[q_1]],..,[[q_n]],[[p_1]],..,[[p_m]])を安定ソートすることで置換<σ>を生成する第一置換生成手段と、j=2,..,aに対して、(1) 前記表Lの第j列([[$v_{1,j}$]],..,[[$v_{m,j}$]])と[[0]]をn個並べた要素列([[0]],..,[[0]])を用いて要素列[[f]]=([[$v_{1,j}$]],..,[[$v_{m,j}$]],[[0]],..,[[0]],[[$-v_{1,j}$]],..,[[$-v_{m,j}$]])を生成し、(2) 前記置換<σ>を用いて前記要素列[[f]]から要素列[[g]]=[[σ([[f]])]]を生成し、(3) 前記要素列[[g]]のプリフィックスサムを計算することにより、要素列[[g']]=PrefixSum([[g]])を生成し、(4) 前記置換<σ>の逆置換<$σ^{-1}$>を用いて前記要素列[[g']]から要素列[[f']]=[[$σ^{-1}$([[g']])]]を生成し、(5) 前記要素列[[f']]の第m+1要素から第m+n要素までの部分要素列([[f'$_{m+1}$]],..,[[f'$_{m+n}$]])を取り出し、前記表Jの第j列([[v'$_{1,j}$]],..,[[v'$_{n,j}$]])=([[f'$_{m+1}$]],..,[[f'$_{m+n}$]])を生成することにより、前記表Jの第2列から第a列を生成する第一列生成手段と、(1) m個の[[1]]からなる要素列([[1]],..,[[1]])とn個の[[0]]からなる要素列([[0]],..,[[0]])を用いて要素列[[f1]]=([[1]],..,[[1]],[[0]],..,[[0]],[[-1]],..,[[-1]])を生成し、(2) 前記置換<σ>を用いて前記要素列[[f1]]から要素列[[g1]]=[[σ([[f1]])]]を生成し、(3) 前記要素列[[g1]]のプリフィックスサムを計算することにより、要素列[[g1']]=PrefixSum([[g1]])を生成し、(4) 前記逆置換<$σ^{-1}$>を用いて前記要素列[[g1']]から要素列[[f1']]=[[$σ^{-1}$([[g1']])]]を生成し、(5) 前記要素列[[f1']]の第m+1要素から第m+n要素までの部分要素列([[f1'$_{m+1}$]],..,[[f1'$_{m+n}$]])を取り出し、結合結果要素列([[e_1]],..,[[e_n]])=([[f1'$_{m+1}$]],..,[[f1'$_{m+n}$]])を生成する結合結果要素列生成手段と、j=a+1,..,a+b-1に対して、前記結合結果要素列([[e_1]],..,[[e_n]])と前記表Rの第j-a+1列([[$u_{1,j-a+1}$]],..,[[$u_{m,j-a+1}$]])を用いて前記表Jの第j列([[u'$_{1,j-a+1}$]],..,[[u'$_{n,j-a+1}$]])=([[e_1]]×[[$u_{1,j-a+1}$]],..,[[e_n]]×[[$u_{n,j-a+1}$]])を生成することにより、前記表Jの第a+1列から第a+b-1列を生成する第二列生成手段と、前記結合結果要素列([[e_1]],..,[[e_n]])と前記表Rの第１列([[q_1]],..,[[q_n]])を用いて前記表Jの第1列([[q'$_1$]],..,[[q'$_n$]])=([[e_1]]×[[q_1]],..,[[e_n]]×[[q_n]])を生成する第三列生成手段とを含む。

【発明の効果】

【００１０】

本発明によれば、等結合する２つの表の行数をm，nとして、等結合に必要な通信量を０((m+n)log(m+n))に削減することができる。

【図面の簡単な説明】

【００１１】

【図１Ａ】２つの表から等結合により１つの表を生成する例を示す図（入力される表L_sの図）である。

【図１Ｂ】２つの表から等結合により１つの表を生成する例を示す図（入力される表R_sの図）である。

【図１Ｃ】２つの表から等結合により１つの表を生成する例を示す図（出力される表Jの図）である。

【図２Ａ】第一実施形態の秘密等結合アルゴリズムの入力となる２つの表と出力となる１つの表を示す図（入力される表Lの図）である。

【図２Ｂ】第一実施形態の秘密等結合アルゴリズムの入力となる２つの表と出力となる１つの表を示す図（入力される表Rの図）である。

【図２Ｃ】第一実施形態の秘密等結合アルゴリズムの入力となる２つの表と出力となる１つの表を示す図（出力される表Jの図）である。

【図３】第一実施形態の秘密等結合アルゴリズムの処理手順を示す図である。

【図４】第一実施形態の秘密等結合アルゴリズムの出力結果である表J（平文）の例を示す図である。

【図５】秘密等結合システム１０の構成を示すブロック図である。

【図６】秘密等結合装置１００$_i$の構成を示すブロック図である。
【図７】秘密等結合システム１０の動作を示すフローチャートである。
【図８】第二実施形態の秘密等結合アルゴリズムの処理手順を示す図である。
【図９】第二実施形態の秘密等結合アルゴリズムの出力結果である表J’（平文）の例を示す図である。
【図１０】秘密等結合装置２００$_i$の構成を示すブロック図である。
【図１１】秘密等結合システム２０の動作を示すフローチャートである。
【図１２】第三実施形態の秘密等結合アルゴリズムの処理手順を示す図である。
【図１３】第三実施形態の秘密等結合アルゴリズムの出力結果である表J’’（平文）の例を示す図である。
【図１４】秘密等結合装置３００$_i$の構成を示すブロック図である。
【図１５】秘密等結合システム３０の動作を示すフローチャートである。
【発明を実施するための形態】
【００１２】
以下、本発明の実施の形態について、詳細に説明する。なお、同じ機能を有する構成部には同じ番号を付し、重複説明を省略する。
【００１３】
後述する秘密等結合アルゴリズムは、既存の秘密計算上の演算の組み合わせで構築される。これらの秘密等結合アルゴリズムが必要とする演算は、秘匿化と復元、加算、乗算、プリフィックスサム、置換、逆置換、安定ソートである。各演算について説明していく前に、必要になる定義や記法について説明する。
【００１４】
＜定義と記法＞
Z_Nを0からNまでの整数の集合（Nは１以上の整数）とする。つまり、$Z_N=\{0,..,N\}$であり、Z_Nは有限環をなす。
【００１５】
[[x]]を$x\in Z_N$を暗号化ないし秘密分散で秘匿した値（秘匿文）とする。また、xを[[x]]の平文という。
【００１６】
[[x]]+[[y]]を秘密計算による加算とし、[[x]],[[y]]を入力とし、[[x+y]]を出力する。
【００１７】
[[x]]×[[y]]を秘密計算による乗算とし、[[x]],[[y]]を入力とし、[[x×y]]を出力する。
【００１８】
PrefixSum([[x_1]],.., [[x_n]])は、要素列([[x_1]],.., [[x_n]])からプリフィックスサムと呼ばれる要素列を求める演算であり、詳細については後ほど説明する。
【００１９】
<σ>は、秘密計算による置換σを表す。詳細については後ほど説明する。
【００２０】
要素列$f=(f_1,..,f_n)$の各要素を秘匿化した要素列([[f_1]],.., [[f_n]])を[[f]]と表す。つまり、[[f]]=([[f_1]],.., [[f_n]])である。
【００２１】
[[σ([[f]])]]は置換σによって要素列[[f]]を置換した要素列を表す。
【００２２】
Sort([[x_1]],.., [[x_n]])は、要素列([[x_1]],.., [[x_n]])を入力とし、置換<σ>を出力する安定ソートを表す。
【００２３】
＜演算アルゴリズム＞
［秘匿化と復元］

$x \in Z_N$から[[x]]を求める方法（秘匿化）、[[x]]から$x \in Z_N$を求める方法（復元）としては、具体的には、千田らの手法（非特許文献１）やShamirの手法（参考非特許文献１）がある。

（参考非特許文献１）Shamir, A., “How to share a secret”, Communications of the ACM, Vol.22, No.11, pp.612-613, 1979.

【００２４】

秘匿化の一例について説明する。マルチパーティプロトコルにおける参加者をX, Y, Zとするとき、$x \in Z_N$は複数（例えば3個）の秘密値に分散されており、[[x]]は複数の秘密値x_i（$i \in \{1,2,3\}$）の集合を表している。参加者X, Y, Zは、各人に割り当てられた秘密値x_i（$i \in \{1,2,3\}$）の一部を保有しているが、秘密値x_i（$i \in \{1,2,3\}$）の全部を保有していない。例えば、参加者Xは、集合$\{x_2, x_3\}$、参加者Yは、集合$\{x_1, x_3\}$、参加者Zは、集合$\{x_1, x_2\}$をそれぞれ保有するものとする。

【００２５】

［加算、乗算］

加算は、$a, b \in Z_N$の[[a]], [[b]]が与えられたとき、c=a+bとなる[[c]]を秘匿した状態で求めるアルゴリズムである。具体的には、Ben-Orらの手法（参考非特許文献２）が知られており、マルチパーティプロトコルにおける参加者間の通信は不要である。

【００２６】

乗算は、$a, b \in Z_N$の[[a]], [[b]]が与えられたとき、c=a×bとなる[[c]]を秘匿した状態で求めるアルゴリズムである。具体的には Gennaroらの方法（参考非特許文献３）を用いることができる。この方法ではマルチパーティプロトコルにおける参加者間の通信が必要となる。

（参考非特許文献２）Ben-Or, M., Goldwasser, S. and Wigderson, A., “Completeness theorems for non-cryptographic fault-tolerant distributed computation”, Proceedings of the twentieth annual ACM symposium on Theory of computing, ACM, pp. 1-10, 1988.

（参考非特許文献３）Gennaro, R., Rabin, M. O. and Rabin, T., “Simplified VSS and fast-track multiparty computations with applications to threshold cryptography”, Proceedings of the seventeenth annual ACM symposium on Principles of distributed computing, ACM, pp.101-111, 1998.

【００２７】

［Prefix-sum（プリフィックスサム）］

順序立てられた複数の要素（要素列）に対して、ある要素とそれまでに現れたすべての要素との和を求める操作およびその結果の要素列をプリフィックスサムと呼ぶ。すなわち、要素列([[x_1]],.., [[x_n]])が与えられたとき、次式で求められるy_iを用いて要素列([[y_1]],.., [[y_n]])を求める操作である。

【００２８】

【数１】

$$y_i = \sum_{j=1}^{i} x_j \quad \cdots(1)$$

【００２９】

以下では、この操作を([[y_1]],.., [[y_n]])←PrefixSum([[x_1]],.., [[x_n]])と表すこととする。加算アルゴリズムとしてBen-Orらの手法（参考非特許文献２）を用いることにより、プリフィックスサムはマルチパーティプロトコルにおける参加者間の通信が不要となる。

【００３０】

［置換］

順序立てられた要素を並べ替える操作、その写像を置換と呼ぶ。例えば、(1, 2, 3)を(3, 1, 2)に並べ替える操作は、$\sigma(1)=3$, $\sigma(2)=1$, $\sigma(3)=2$を満たす写像σによる置換と

みなされる。写像の性質から複数の置換を合成できる。すなわち、置換σ，πに対して合成置換σ・πは要素xをσ(π(x))に写す。

【００３１】

秘密計算による置換<σ>を実現する方法の一つとして、五十嵐らの手法（参考非特許文献４）を利用することができる。

（参考非特許文献４）五十嵐大，濱田浩気，菊池亮，千田浩司，“インターネット環境レスポンス1秒の統計処理を目指した，秘密計算基数ソートの改良”，暗号と情報セキュリティシンポジウム(SCIS)2014，電子情報通信学会，2014.

【００３２】

この方法では、例えば、マルチパーティプロトコルにおける参加者をX，Y，Zとするとき、置換σを$\sigma=\sigma_{ZX}\cdot\sigma_{YZ}\cdot\sigma_{XY}$を満たす置換$\sigma_{XY}$，$\sigma_{YZ}$，$\sigma_{ZX}$の合成とする。置換$\sigma_{XY}$は参加者Xと参加者Yのみに共有された置換であり、参加者Zには知らされない。参加者Zは、参加者Xと参加者Yからの再分散によってσ_{XY}による置換後の秘匿要素列を得る。置換σ_{YZ}，σ_{ZX}も同様であり、いずれの参加者にとっても自分の知らない置換が含まれることになる。したがって、すべての参加者が対応関係を知らないまま秘密計算による置換<σ>を実行することができる。

【００３３】

ここで、置換<σ>は、置換σ_{XY}，σ_{YZ}，σ_{ZX}に分割されて適用順序が決められた複製秘密分散（参考非特許文献５）とみなすことができる。これは、σをランダム置換とした濱田らのシャッフル（参考非特許文献６）の一般化であり、通信コストは入力サイズに対して線形となる。

（参考非特許文献５）Ito，M.，Saito，A. and Nishizeki，T.，“Secret sharing scheme realizing general access structure”，Electronics and Communications in Japan (Part III: Fundamental Electronic Science)，Vol.72，No.9，pp.56-64，1989.

（参考非特許文献６）濱田浩気，五十嵐大，千田浩司，高橋克巳，“3パーティ秘匿関数計算上のランダム置換プロトコル”，コンピュータセキュリティシンポジウム(CSS)2010，情報処理学会コンピュータセキュリティ研究会，2010.

【００３４】

［逆置換］

置換は全単射であるため、逆写像を持つ。そこで、ある置換σの逆写像を逆置換と呼び、σ^{-1}で表す。すなわち、σ(x)=yのとき、$\sigma^{-1}(y)=x$である。特に、$\sigma^{-1}\cdot\sigma$は恒等置換と呼ばれる順序を変えない置換である。

【００３５】

秘密計算による逆置換<σ^{-1}>の方法は、置換の場合と同様である。置換<σ>が$\sigma=\sigma_{ZX}\cdot\sigma_{YZ}\cdot\sigma_{XY}$となる置換$\sigma_{XY}$，$\sigma_{YZ}$，$\sigma_{ZX}$の合成として与えられたとき、逆置換は$\sigma^{-1}=\sigma_{XY}^{-1}\cdot\sigma_{YZ}^{-1}\cdot\sigma_{ZX}^{-1}$となることから、各参加者X，Y，Zが持つ置換の逆置換の順序を反転して適用すればよい。

【００３６】

［安定ソート］

同じ要素の順序関係が保存される並べ替えを安定ソートという。すなわち、$(x_1,..,x_n)$を$(y_1,..,y_n)$に並べ替える安定ソートσにおいて$x_i=x_j$のとき$\sigma(x_i)=y_u$，$\sigma(x_j)=y_v$とすると、i<jのときのみu<vとなる。

【００３７】

秘密計算による手法としては、例えば五十嵐らの方法（参考非特許文献４）が挙げられる。以下では、要素列$([[x_1]],..,[[x_n]])$に対する安定ソートによる出力を置換<σ>とし、<σ>←$Sort([[x_1]],..,[[x_n]])$と表す。

【００３８】

＜第一実施形態＞

第一実施形態の秘密等結合アルゴリズムの入力、出力、処理手順、処理コスト、秘密等結合アルゴリズムを実現する秘密等結合システムについて、以下説明していく。

【００３９】
なお、以下では、表の列を抜き出して扱う場合、縦方向に並べたものとしてではなく、横方向に並べた要素列として扱う。
【００４０】
［入力］
結合する２つの表をそれぞれL，Rとする（図２Ａ及び図２Ｂ参照）。表の大きさは表Lがm行a列、表Rがn行b列とする（m，nは1以上の整数、a，bは2以上の整数）。表L，Rはともにキー属性値を1列目に持ち、それぞれ([[p_1]],..,[[p_m]])，([[q_1]],..,[[q_n]])とする。表L，Rはキー属性値以外の属性値を2列目以降に持ち、i行j列の属性値をそれぞれ[[$v_{i,j}$]](1≦i≦m，2≦j≦a)，[[$u_{i,j}$]] (1≦i≦n，2≦j≦b)とする。
【００４１】
表L，Rの各要素を秘匿化前の平文に戻した要素（以下、平文要素ともいう）からなる表をそれぞれL_{plain}，R_{plain}とする。表L_{plain}のキー属性値の要素列(p_1,..,p_m)には同じ値の要素が存在しない（つまり、p_1,..,p_mは互いに異なる）こととし、表R_{plain}のキー属性値の要素列(q_1,..,q_n)には重複する値が存在してもよいこととする。また、入力されるすべての要素の平文要素p_i(1≦i≦m)，$v_{i,j}$(1≦i≦m，2≦j≦a)，q_i(1≦i≦n)，$u_{i,j}$(1≦i≦n，2≦j≦b)についてその値が0ではない有限環Z_N上の値とする。もし、平文要素に0が含まれる可能性がある場合には、例えば一律に加減算することにより0とならないように前処理をしておくこととする。また、平文要素が文字列など有限環Z_N上の値以外である可能性がある場合には、有限環Z_N上の値とするように前処理をしておくこととする。
【００４２】
［出力］
出力する表をJとする（図２Ｃ参照）。表Jは、i行目(1≦i≦n)が([[q'_i]]，[[$v'_{i,2}$]],..,[[$v'_{i,a}$]]，[[$u'_{i,2}$]],..,[[$u'_{i,b}$]])となるn行からなる表であり、表R_{plain}のキー属性値q_iと同じ値が表L_{plain}のキー属性値の要素列(p_1,..,p_m)の中に存在しないときは、i行目の値はすべて0、すなわち、2≦α≦a，2≦β≦bについて、q'_i=0，$v'_{i,\alpha}$=0，$u'_{i,\beta}$=0となる。一方、表R_{plain}のキー属性値q_iに対してq_i=p_jとなる表L_{plain}のキー属性値p_jが存在するときは、q'_i=q_i，$v'_{i,\alpha}$=$v_{j,\alpha}$，$u'_{i,\beta}$=$u_{i,\beta}$となる。
【００４３】
［処理手順］
図３に示す第一実施形態の秘密等結合アルゴリズムの処理手順について説明する。その際、図３の左端の数字を用いてステップ１、ステップ２等と表現することにする。また、アルゴリズムの挙動が分かりやすくなるように、m=3，a=3，n=4，b=2として図１Ａの表L_s，図１Ｂの表R_sの値を図２Ａの表L，図２Ｂの表Rに代入して説明する。つまり、平文の状態で説明する。
【００４４】
まず、ステップ１では、入力された２つの表L，Rのキー属性値から生成される要素列([[p_1]],..,[[p_m]]，[[q_1]],..,[[q_n]]，[[p_1]],..,[[p_m]])を安定ソートすることで置換<σ>を生成する。
【００４５】
置換<σ>は、以下の１行目の要素列を２行目の順序に並べ替える置換である。１行目の要素列の左から１番目から３番目まで、４番目から７番目まで、８番目から１０番目までがそれぞれ表L_sのキー属性値、表R_sのキー属性値、表L_sのキー属性値となっている。
（１行目）([[3，[[5，[[9，[3]，[7]，[9]，[9]，3]]，5]]，9]])
（２行目）([[3，[3]，3]]，[[5，5]]，[7]，[[9，[9]，[9]，9]])
【００４６】
なお、ここでは説明が分かりやすくなるように、置換前後の位置関係を[[x，[y]，z]]という記号を用いて同じ値について区別して表現することにする。ただし、実際には区別されずに処理される。[[xとz]]が表Lから、[y]が表Rから得られた属性値に対応する平文である。

【００４７】
次に、ステップ２から８までは表Lの列ごとに繰り返される処理である(j=2,..，a)。図１Ａの表L_sの例では、「身長」と「体重」の列があるが(a=3)、ここでは列「体重」を用いて処理を追っていくことにする。
【００４８】
ステップ３では、表Lの第j列([[$v_{1,j}$]],..，[[$v_{m,j}$]])と[[0]]をn個並べた要素列([[0]],..，[[0]])を用いて要素列[[f]]=([[$v_{1,j}$]],..，[[$v_{m,j}$]]，[[0]],..，[[0]]，[[-$v_{1,j}$]],..，[[-$v_{m,j}$]])を生成する。
【００４９】
要素列[[f]]の平文要素列は、列「体重」の値である100，19，85を用いると、次のようになる。なお、要素列([0]，[0]，[0]，[0])は、([[0]],..，[[0]])の平文要素列である。

([[100，[[19，[[85，[0]，[0]，[0]，[0]，-100]]，-19]]，-85]])

【００５０】
ステップ４では、置換<σ>を用いて要素列[[f]]から要素列[[g]]=[[σ([[f]])]]を生成する。
【００５１】
置換<σ>によって要素列[[f]]の平文要素列を並び替えると、要素列[[g]]の平文要素列が得られる。
【００５２】

([[100，[0]，-100]]，[[19，-19]]，[0]，[[85，[0]，[0]，-85]])

ステップ５では、要素列[[g]]のプリフィックスサムを計算することにより、要素列[[g’]]=PrefixSum([[g]])を生成する。
【００５３】
プリフィックスサムによって得られる要素列[[g’]]の平文要素列は、以下のようになる。

([[100，[100]，0]]，[[19，0]]，[0]，[[85，[85]，[85]，0]])

この手順により表Lの列の値に対応する平文[[xが[y]にコピーされていることがわかる。また、z]]はプログラミングにおける番兵であり、[[xの値がコピーされる終端を示している。
【００５４】
ステップ６では、置換<σ>の逆置換<$σ^{-1}$>を用いて要素列[[g’]]から要素列[[f’]]=[[$σ^{-1}$([[g’]])]]を生成する。
【００５５】
逆置換<$σ^{-1}$>によって要素列[[g’]]の平文要素列を並び替えると、要素列[[f’]]の平文要素列が得られる。

([[100，[[19，[[85，[100]，[0]，[85]，[85]，0]]，0]]，0]])

【００５６】
ステップ７では、要素列[[f’]]の第m+1要素から第m+n要素までの部分要素列([[f’$_{m+1}$]],..，[[f’$_{m+n}$]])を取り出し、表Jの第j列([[v’$_{1,j}$]],..，[[v’$_{n,j}$]])=([[f’$_{m+1}$]],..，[[f’$_{m+n}$]])を生成する。
【００５７】
生成される表Jの第j列の値は、要素列[[f’]]の平文要素列の第4(=3+1)要素から第7(=3+4)要素の、[100]，[0]，[85]，[85]となる。ここで、結合されなかった行の値である表Jの第２行の値は、平文では、[0]となる。
【００５８】
ステップ９から１３までの処理は、表Lの列の値をすべて[[1]]とした仮想的な列に対してステップ３から７までの処理を適用したものとなっている。具体的には、以下のようになる。
【００５９】

ステップ９では、m個の[[1]]からなる要素列([[1]],.., [[1]])とn個の[[0]]からなる要素列([[0]],.., [[0]])を用いて要素列[[f1]]=([[1]],.., [[1]], [[0]],.., [[0]], [[-1]],.., [[-1]])を生成する。

【００６０】

要素列[[f1]]の平文要素列は次のようになる。

([[1, [[1, [[1, [0], [0], [0], [0], -1]], -1]], -1]])

【００６１】

ステップ１０では、置換<σ>を用いて要素列[[f1]]から要素列[[g1]]=[[σ([[f1]])]]を生成する。

【００６２】

置換<σ>によって要素列[[f1]]の平文要素列を並び替えると、要素列[[g1]]の平文要素列が得られる。

([[1, [0], -1]], [[1, -1]], [0], [[1, [0], [0], -1]])

【００６３】

ステップ１１では、要素列[[g1]]のプリフィックスサムを計算することにより、要素列[[g1’]]=PrefixSum([[g1]])を生成する。

【００６４】

プリフィックスサムによって得られる要素列[[g1’]]の平文要素列は、以下のようになる。

([[1, [1], 0]], [[1, 0]], [0], [[1, [1], [1], 0]])

【００６５】

ステップ１２では、逆置換<σ^{-1}>を用いて要素列[[g1’]]から要素列[[f1’]]=[[σ^{-1}([[g1’]])]]を生成する。

【００６６】

逆置換<σ^{-1}>によって要素列[[g1’]]の平文要素列を並び替えると、要素列[[f1’]]の平文要素列が得られる。

([[1, [[1, [[1, [1], [0], [1], [1], 0]], 0]], 0]])

【００６７】

ステップ１３では、要素列[[f1’]]の第m+1要素から第m+n要素までの部分要素列([[f1’$_{m+1}$]],.., [[f1’$_{m+n}$]])を取り出し、結合結果要素列([[e_1]],.., [[e_n]])=([[f1’$_{m+1}$]],.., [[f1’$_{m+n}$]])を生成する。ここで、結合結果要素列([[e_1]],.., [[e_n]])は表Rの行に対して表Lに結合する行があったか否かを示す要素の列であり、[[e_i]]= [[1]]である場合、結合する行があったことを示し、[[e_i]]= [[0]]である場合、結合する行がなかったことを示す。

【００６８】

生成される結合結果要素列の値は、要素列[[f1’]]の平文要素列の第4(=3+1)要素から第7(=3+4)要素の、[1], [0], [1], [1]となる。

【００６９】

ステップ１４から２１までの処理は、結合結果要素列([[e_1]],.., [[e_n]])を用いて表Rの結合されなかった行の値を[[0]]にする処理である。

【００７０】

ステップ１５～１７では、結合結果要素列([[e_1]],.., [[e_n]])と表Rの第j-a+1列([[$u_{1,j-a+1}$]],.., [[$u_{m,j-a+1}$]])を用いて表Jの第j列([[u’$_{1,j-a+1}$]],.., [[u’$_{n,j-a+1}$]])=([[e_1]]×[[$u_{1,j-a+1}$]],.., [[e_n]]×[[$u_{n,j-a+1}$]])を生成する(j=a+1,.., a+b-1)。

【００７１】

ステップ１９～２１では、結合結果要素列([[e_1]],.., [[e_n]])と表Rの第１列([[q_1]],.., [[q_n]])を用いて表Jの第1列([[q’$_1$]],.., [[q’$_n$]])=([[e_1]]×[[q_1]],.., [[e_n]]×[[q_n]])を生成する。

【００７２】

図１Ｂの表R_sの例では、結合されない「ミックスオレ」の行（第２行）が0となり、他

の行には元と同じ値が代入されることになる。したがって、出力結果である表J（平文）は図４のようになる。

【００７３】

［処理コスト］

第一実施形態の秘密等結合アルゴリズムの中で通信が必要となる処理は、ステップ１における長さ2m+nの安定ソートが1回、ステップ４及びステップ１０における長さ2m+nの置換が合わせてa回、同様にステップ６及びステップ１２における長さ2m+nの逆置換が合わせてa回、ステップ１６及びステップ２０における乗算がbn回である。

【００７４】

安定ソートの通信コストは、濱田らの手法（参考非特許文献６）により、O((m+n)log(m+n))であり、置換および逆置換については入力に対して線形、乗算については1回につき定数量の通信が必要となる。各表の列の数a，bは定数とみなせるので、通信量は入力する2つの表の行数をm，nとすると、O((m+n)log(m+n))となる。

【００７５】

［秘密等結合システム］

以下、図５～図７を参照して第一実施形態の秘密等結合システム１０について説明する。図５は、秘密等結合システム１０の構成を示すブロック図である。秘密等結合システム１０は、W個（Wは3以上の所定の整数）の秘密等結合装置１００_1、…、１００_Wを含む。秘密等結合装置１００_1、…、１００_Wは、ネットワーク８００に接続しており、相互に通信可能である。ネットワーク８００は、例えば、インターネットなどの通信網あるいは同報通信路などでよい。図６は、秘密等結合装置１００_i(1≦i≦W)の構成を示すブロック図である。図７は、秘密等結合システム１０の動作を示すフローチャートである。

【００７６】

図６に示すように秘密等結合装置１００_iは、第一置換生成部１１０_iと、第一列生成部１２０_iと、結合結果要素列生成部１３０_iと、第二列生成部１４０_iと、第三列生成部１５０_iと、記録部１９０_iを含む。記録部１９０_iを除く秘密等結合装置１００_iの各構成部は、秘密等結合アルゴリズムで必要とされる演算、つまり、少なくとも秘匿化、復元、加算、乗算、プリフィックスサム、置換、逆置換、安定ソートのうち、各構成部の機能を実現するうえで必要になる演算を実行できるように構成されている。本発明において個々の演算を実現するための具体的な機能構成は、例えば非特許文献１及び参考非特許文献１～６のそれぞれで開示されるアルゴリズムを実行できるような構成で十分であり、これらは従来的構成であるから詳細な説明については省略する。また、記録部１９０_iは、秘密等結合装置１００_iの処理に必要な情報を記録する構成部である。

【００７７】

W個の秘密等結合装置１００_iによる協調計算によって、秘密等結合システム１０はマルチパーティプロトコルである秘密等結合アルゴリズムを実現する。よって、秘密等結合システム１０の第一置換生成手段１１０（図示していない）は第一置換生成部１１０_1、…、１１０_Wで構成され、第一列生成手段１２０（図示していない）は第一列生成部１２０_1、…、１２０_Wで構成され、結合結果要素列生成手段１３０（図示していない）は結合結果要素列生成部１３０_1、…、１３０_Wで構成され、第二列生成手段１４０（図示していない）は第二列生成部１４０_1、…、１４０_Wで構成され、第三列生成手段１５０（図示していない）は第三列生成部１５０_1、…、１５０_Wで構成される。

【００７８】

秘密等結合システム１０は、各要素が秘匿されているm行a列の表Lと、n行b列の表Rを入力とし、表Lと表Rを秘密等結合することによりn行a+b-1列の表Jを生成する（図２Ｃ参照）。以下、図７に従い秘密等結合システム１０の動作について説明する。

【００７９】

第一置換生成手段１１０は、表Lの第1列($[[p_1]],..,[[p_m]]$)、表Rの第1列($[[q_1]],..,[[q_n]]$)から生成される要素列($[[p_1]],..,[[p_m]],[[q_1]],..,[[q_n]],[[p_1]],..,[[p_m]]$)を安定ソートすることで置換$<\sigma>$を生成する（Ｓ１１０）。図３の秘密等結合アルゴ

リズムのステップ１に対応する。

【００８０】

第一列生成手段１２０は、j=2,..., aに対して以下の処理を実行することにより、表Jの第2列から第a列を生成する（Ｓ１２０）。図３の秘密等結合アルゴリズムのステップ２～８に対応する。

(1) 表Lの第j列([[$v_{1,j}$]],..., [[$v_{m,j}$]])と[[0]]をn個並べた要素列([[0]],..., [[0]])を用いて要素列[[f]]=([[$v_{1,j}$]],..., [[$v_{m,j}$]], [[0]],..., [[0]], [[-$v_{1,j}$]],..., [[-$v_{m,j}$]])を生成する。

(2) 置換<σ>を用いて要素列[[f]]から要素列[[g]]=[[σ([[f]])]]を生成する。

(3) 要素列[[g]]のプリフィックスサムを計算することにより、要素列[[g']]=PrefixSum([[g]])を生成する。

(4) 置換<σ>の逆置換<$σ^{-1}$>を用いて要素列[[g']]から要素列[[f']]=[[$σ^{-1}$([[g']])]]を生成する。

(5) 要素列[[f']]の第m+1要素から第m+n要素までの部分要素列([[f'$_{m+1}$]],..., [[f'$_{m+n}$]])を取り出し、表Jの第j列([[v'$_{1,j}$]],..., [[v'$_{n,j}$]])=([[f'$_{m+1}$]],..., [[f'$_{m+n}$]])を生成する。

【００８１】

結合結果要素列生成手段１３０は、以下の処理を実行することにより、結合結果要素列([[e_1]],..., [[e_n]])を生成する（Ｓ１３０）。図３の秘密等結合アルゴリズムのステップ９～１３に対応する。

(1) m個の[[1]]からなる要素列([[1]],..., [[1]])とn個の[[0]]からなる要素列([[0]],..., [[0]])を用いて要素列[[f1]]=([[1]],..., [[1]], [[0]],..., [[0]], [[-1]],..., [[-1]])を生成する。

(2) 置換<σ>を用いて要素列[[f1]]から要素列[[g1]]=[[σ([[f1]])]]を生成する。

(3) 要素列[[g1]]のプリフィックスサムを計算することにより、要素列[[g1']]=PrefixSum([[g1]])を生成する。

(4) 逆置換<$σ^{-1}$>を用いて要素列[[g1']]から要素列[[f1']]=[[$σ^{-1}$([[g1']])]]を生成する。

(5) 要素列[[f1']]の第m+1要素から第m+n要素までの部分要素列([[f1'$_{m+1}$]],..., [[f1'$_{m+n}$]])を取り出し、結合結果要素列([[e_1]],..., [[e_n]])=([[f1'$_{m+1}$]],..., [[f1'$_{m+n}$]])を生成する。

【００８２】

第二列生成手段１４０は、j=a+1,..., a+b-1に対して以下の処理を実行することにより、表Jの第a+1列から第a+b-1列を生成する（Ｓ１４０）。図３の秘密等結合アルゴリズムのステップ１４～１８に対応する。

(1) 結合結果要素列([[e_1]],..., [[e_n]])と表Rの第j-a+1列([[$u_{1,j-a+1}$]],..., [[$u_{m,j-a+1}$]])を用いて表Jの第j列([[u'$_{1,j-a+1}$]],..., [[u'$_{n,j-a+1}$]])=([[e_1]]×[[$u_{1,j-a+1}$]],..., [[e_n]]×[[$u_{n,j-a+1}$]])を生成する。

【００８３】

第三列生成手段１５０は、結合結果要素列([[e_1]],..., [[e_n]])と表Rの第１列([[q_1]],..., [[q_n]])を用いて表Jの第1列([[q'$_1$]],..., [[q'$_n$]])=([[e_1]]×[[q_1]],..., [[e_n]]×[[q_n]])を生成する（Ｓ１５０）。図３の秘密等結合アルゴリズムのステップ１９～２１に対応する。

【００８４】

本実施形態の発明によれば、等結合する2つの表のキー属性値を所定の方法で並べた要素列に対して安定ソートを行った結果生成される置換を用いて、片方の表のキー属性値に重複がある場合でも等結合することができる。従来技術では、重複する要素の最大数をkとして1つの要素をk個の重複しない情報に置き換えていたが、この置き換えが不要になるため、データを秘匿した状態での等結合に必要なサーバ間の通信量を削減することができる。具体的には、等結合する２つの表の行数をm, nとして、等結合に必要な通信量をO((m

+n)log(m+n))に削減することができる。また、従来技術で必要であった結合するキー属性の重複する要素の最大数が既知である必要がなくなる。

【００８５】

＜第二実施形態＞

第一実施形態の秘密等結合アルゴリズムが出力する表Jは、入力した表Rの行の順序が保たれたまま、結合されなかった表Rの行の各要素がすべて[[0]]となっている（図４の表J（平文）では0となっている）。このため、表Jを復元した際に、入力した表Rの何行目が結合されなかったのかが分かってしまう。この問題を解決するために、第一実施形態の秘密等結合アルゴリズムを実行後、結合した行を表の上に寄せる処理を実行する第二実施形態の秘密等結合アルゴリズムについて説明する。

【００８６】

第二実施形態の秘密等結合アルゴリズムの入力、出力、処理手順、処理コスト、秘密等結合アルゴリズムを実現する秘密等結合システムについて、以下説明していく。

【００８７】

［入力］

第一実施形態の秘密等結合アルゴリズムにおける結合結果要素列([[e_1]],..,[[e_n]])と出力結果である表Jが入力となる。

【００８８】

［出力］

表Jの行を並び替えた表J’が出力となる。表J’は、表Jの行の中ですべての要素が[[0]]となる行を下に寄せた表となる。

【００８９】

表J’は、i行目(1≦i≦n)が([[q’’$_i$]],[[v’’$_{i,2}$]],..,[[v’’$_{i,a}$]],[[u’’$_{i,2}$]],..,[[u’’$_{i,b}$]])となるn行からなる表である。ただし、[[q’’$_i$]]はキー属性値または[[0]]、[[v’’$_{i,2}$]],..,[[v’’$_{i,a}$]]は表Lから結合された行またはa-1個の[[0]]、[[u’’$_{i,2}$]],..,[[u’’$_{i,b}$]]は表Rから結合された行またはb-1個の[[0]]となっている（図８参照）。

【００９０】

［処理手順］

第二実施形態の秘密等結合アルゴリズムを図８に示す。

【００９１】

まず、ステップ１では、結合結果要素列([[e_1]],..,[[e_n]])を安定ソートすることで置換<σ~>を生成する。

【００９２】

置換<σ~>は、第一実施形態の結合結果要素列([[e_1]],..,[[e_n]])の平文要素列([1],[0],[1],[1])を用いると、以下の１行目の要素列を２行目の順序に並べ替える置換である。

（１行目）([1],[0],[1],[1])

（２行目）([1],[1],[1],[0])

【００９３】

ステップ２では、置換<σ~>を用いて表Jの第1列([[q’$_1$]],..,[[q’$_n$]])から表J’の第1列([[q’’$_1$]],..,[[q’’$_n$]])=[[σ~([[q’$_1$]],..,[[q’$_n$]])]]を生成する。

【００９４】

ステップ４では、置換<σ~>を用いて表Jの第j列([[v’$_{1,j}$]],..,[[v’$_{n,j}$]])から表J’の第j列([[v’’$_{1,j}$]],..,[[v’’$_{n,j}$]])=[[σ~([[v’$_{1,j}$]],..,[[v’$_{n,j}$]])]]を生成する(j=2,..,a)。

【００９５】

ステップ７では、置換<σ~>を用いて表Jの第j列([[u’$_{1,j-a+1}$]],..,[[u’$_{n,j-a+1}$]])から表J’の第j列([[u’’$_{1,j-a+1}$]],..,[[u’’$_{n,j-a+1}$]])=[[σ~([[u’$_{1,j-a+1}$]],..,[[u’$_{n,j-a+1}$]])]]を生成する(j=a+1,..,a+b-1)。

【００９６】
ステップ２、ステップ４、ステップ７により要素がすべて[[0]]となっている行が下に寄せられ、表J’が出力される。したがって、出力結果である表J’（平文）は図９のようになる。
【００９７】
なお、ステップ１で用いる安定ソートとして昇順のものを採用した場合、出力の行の上下を反転することとなり、要素がすべて[[0]]の行を上に寄せることができる。
【００９８】
［処理コスト］
第二実施形態の秘密等結合アルゴリズムによる通信コストは、安定ソートに必要なO(nlog(n))である。
【００９９】
［秘密等結合システム］
以下、図１０～図１１を参照して第二実施形態の秘密等結合システム２０について説明する。秘密等結合システム２０は、W個（Wは3以上の所定の整数）の秘密等結合装置１００$_1$、…、１００$_W$を含む代わりに、W個の秘密等結合装置２００$_1$、…、２００$_W$を含む点において秘密等結合システム１０と異なる。図１０は、秘密等結合装置２００$_i$(1≦i≦W)の構成を示すブロック図である。図１１は、秘密等結合システム２０の動作を示すフローチャートである。
【０１００】
図１０に示すように秘密等結合装置２００$_i$は、第二置換生成部２６０$_i$と、行並び替え部２７０$_i$を更に含む点において秘密等結合装置１００$_i$と異なる。第二置換生成部２６０$_i$と、行並び替え部２７０$_i$も、秘密等結合アルゴリズムで必要とされる演算のうち、その機能を実現するうえで必要になる演算を実行できるように構成されている。
【０１０１】
秘密等結合システム２０の第二置換生成手段２６０（図示していない）は第二置換生成部２６０$_1$、…、２６０$_W$で構成され、行並び替え手段２７０（図示していない）は行並び替え部２７０$_1$、…、２７０$_W$で構成される。
【０１０２】
秘密等結合システム２０は、各要素が秘匿されているm行a列の表Lと、n行b列の表Rを入力とし、表Lと表Rを秘密等結合したn行a+b-1列の表Jからn行a+b-1列の表J’を生成する。以下、図１１に従い秘密等結合システム２０の動作について説明する。Ｓ１１０～Ｓ１５０までの処理は秘密等結合システム１０と同様であるので、Ｓ２６０とＳ２７０について説明する。
【０１０３】
第二置換生成手段２６０は、結合結果要素列([[e_1]],..,[[e_n]])を安定ソートすることで置換<σ~>を生成する（Ｓ２６０）。図８の秘密等結合アルゴリズムのステップ１に対応する。
【０１０４】
行並び替え手段２７０は、置換<σ~>を用いて、表Jの第1列([[q’$_1$]],..,[[q’$_n$]])から表J’の第1列([[q’’$_1$]],..,[[q’’$_n$]])=[[σ~([[q’$_1$]],..,[[q’$_n$]])]]を生成し、j=2,..,aに対して表Jの第j列([[v’$_{1,j}$]],..,[[v’$_{n,j}$]])から表J’の第j列([[v’’$_{1,j}$]],..,[[v’’$_{n,j}$]])=[[σ~([[v’$_{1,j}$]],..,[[v’$_{n,j}$]])]]を生成し、j=a+1,..,a+b-1に対して表Jの第j列([[u’$_{1,j-a+1}$]],..,[[u’$_{n,j-a+1}$]])から表J’の第j列([[u’’$_{1,j-a+1}$]],..,[[u’’$_{n,j-a+1}$]])=[[σ~([[u’$_{1,j-a+1}$]],..,[[u’$_{n,j-a+1}$]])]]を生成する（Ｓ２７０）。図８の秘密等結合アルゴリズムのステップ２～８に対応する。
【０１０５】
本実施形態の発明によれば、新たに必要になる通信コストはO(nlog(n))であるため、全体としてはO((m+n)log(m+n))の通信量で実行することができる。
【０１０６】

＜第三実施形態＞

第二実施形態の秘密等結合アルゴリズムが出力する表J’は、結合しなかった行をすべての要素が[[0]]である行として含んだものとなっている。しかし、結合した行の数が参加者に知られてもよい場合、表J’の中から結合し上に寄せた行だけを出力することとしてもよい。第二実施形態の秘密等結合アルゴリズムを実行後、結合した行のみを含む表を出力する第三実施形態の秘密等結合アルゴリズムについて説明する。

【０１０７】

第三実施形態の秘密等結合アルゴリズムの入力、出力、処理手順、処理コスト、秘密等結合アルゴリズムを実現する秘密等結合システムについて、以下説明していく。

【０１０８】

［入力］

第一実施形態の秘密等結合アルゴリズムにおける結合結果要素列([[e_1]],..., [[e_n]])と第二実施形態の秘密等結合アルゴリズムの出力結果である表J’が入力となる。

【０１０９】

［出力］

表J’の行の中ですべての要素が[[0]]となる行を表J’から削除した表が出力となる表J’’となる。

【０１１０】

［処理手順］

第三実施形態の秘密等結合アルゴリズムを図１２に示す。

【０１１１】

まず、ステップ１では、結合した行を数え上げるため、結合結果要素列([[e_1]],..., [[e_n]])の各要素の和[[c]]を計算する。

【０１１２】

ステップ２では、ステップ１で求めた[[c]]を復元して得られるcを公開、表J’の上からc行を抽出した表J’’を生成する。

【０１１３】

ステップ２により要素がすべて[[0]]となっている行が削除された表J’’が出力される。したがって、出力結果である表J’’（平文）は図１３のようになる。

【０１１４】

［処理コスト］

第三実施形態の秘密等結合アルゴリズムによる通信コストは、1回の公開に必要な定数量 O(1)のみである。

【０１１５】

［秘密等結合システム］

以下、図１４～図１５を参照して第三実施形態の秘密等結合システム３０について説明する。秘密等結合システム３０は、W個（Wは3以上の所定の整数）の秘密等結合装置２００$_1$、…、２００$_W$を含む代わりに、W個の秘密等結合装置３００$_1$、…、３００$_W$を含む点において秘密等結合システム２０と異なる。図１４は、秘密等結合装置３００$_i$(1≦i≦W)の構成を示すブロック図である。図１５は、秘密等結合システム３０の動作を示すフローチャートである。

【０１１６】

図１４に示すように秘密等結合装置３００$_i$は、結合行数計算部３８０$_i$と、結合行数公開部３９０$_i$を更に含む点において秘密等結合装置２００$_i$と異なる。結合行数計算部３８０$_i$と、結合行数公開部３９０$_i$も、秘密等結合アルゴリズムで必要とされる演算のうち、その機能を実現するうえで必要になる演算を実行できるように構成されている。

【０１１７】

秘密等結合システム３０の結合行数計算手段３８０（図示していない）は結合行数計算部３８０$_1$、…、３８０$_W$で構成され、結合行数公開手段３９０（図示していない）は結合行数公開部３９０$_1$、…、３９０$_W$で構成される。

【０１１８】
秘密等結合システム３０は、各要素が秘匿されているm行a列の表Lと、n行b列の表Rを入力とし、表Lと表Rを秘密等結合したn行a+b-1列の表Jからn行a+b-1列の表J’を生成し、すべての要素が[[0]]である行を表J’から削除することで、c行a+b-1列の表J’’を生成する。以下、図１５に従い秘密等結合システム３０の動作について説明する。Ｓ１１０～Ｓ２７０までの処理は秘密等結合システム２０と同様であるので、Ｓ３８０とＳ３９０について説明する。
【０１１９】
結合行数計算手段３８０は、結合結果要素列([[e_1]],..,[[e_n]])の各要素の和[[c]]を計算する（Ｓ３８０）。図１２の秘密等結合アルゴリズムのステップ１に対応する。
【０１２０】
結合行数公開手段３９０は、[[c]]を復元して得られるcを公開、表J’の上からc行を抽出した表J’’を生成する（Ｓ３９０）。図１２の秘密等結合アルゴリズムのステップ２に対応する。
【０１２１】
本実施形態の発明によれば、新たに必要になる通信コストは0(1)であるため、全体としては0((m+n)log(m+n))の通信量で実行することができる。
【０１２２】
＜補記＞
本発明の装置は、例えば単一のハードウェアエンティティとして、キーボードなどが接続可能な入力部、液晶ディスプレイなどが接続可能な出力部、ハードウェアエンティティの外部に通信可能な通信装置（例えば通信ケーブル）が接続可能な通信部、ＣＰＵ（Central Processing Unit、キャッシュメモリやレジスタなどを備えていてもよい）、メモリであるＲＡＭやＲＯＭ、ハードディスクである外部記憶装置並びにこれらの入力部、出力部、通信部、ＣＰＵ、ＲＡＭ、ＲＯＭ、外部記憶装置の間のデータのやり取りが可能なように接続するバスを有している。また必要に応じて、ハードウェアエンティティに、ＣＤ－ＲＯＭなどの記録媒体を読み書きできる装置（ドライブ）などを設けることとしてもよい。このようなハードウェア資源を備えた物理的実体としては、汎用コンピュータなどがある。
【０１２３】
ハードウェアエンティティの外部記憶装置には、上述の機能を実現するために必要となるプログラムおよびこのプログラムの処理において必要となるデータなどが記憶されている（外部記憶装置に限らず、例えばプログラムを読み出し専用記憶装置であるＲＯＭに記憶させておくこととしてもよい）。また、これらのプログラムの処理によって得られるデータなどは、ＲＡＭや外部記憶装置などに適宜に記憶される。
【０１２４】
ハードウェアエンティティでは、外部記憶装置（あるいはＲＯＭなど）に記憶された各プログラムとこの各プログラムの処理に必要なデータが必要に応じてメモリに読み込まれて、適宜にＣＰＵで解釈実行・処理される。その結果、ＣＰＵが所定の機能（上記、…部、…手段などと表した各構成要件）を実現する。
【０１２５】
本発明は上述の実施形態に限定されるものではなく、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。また、上記実施形態において説明した処理は、記載の順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るとしてもよい。
【０１２６】
既述のように、上記実施形態において説明したハードウェアエンティティ（本発明の装置）における処理機能をコンピュータによって実現する場合、ハードウェアエンティティが有すべき機能の処理内容はプログラムによって記述される。そして、このプログラムをコンピュータで実行することにより、上記ハードウェアエンティティにおける処理機能が

コンピュータ上で実現される。

【０１２７】

この処理内容を記述したプログラムは、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としては、例えば、磁気記録装置、光ディスク、光磁気記録媒体、半導体メモリ等どのようなものでもよい。具体的には、例えば、磁気記録装置として、ハードディスク装置、フレキシブルディスク、磁気テープ等を、光ディスクとして、ＤＶＤ（Digital Versatile Disc）、ＤＶＤ－ＲＡＭ（Random Access Memory）、ＣＤ－ＲＯＭ（Compact Disc Read Only Memory）、ＣＤ－Ｒ（Recordable）／ＲＷ（ReWritable）等を、光磁気記録媒体として、ＭＯ（Magneto-Optical disc）等を、半導体メモリとしてＥＥＰ－ＲＯＭ（Electronically Erasable and Programmable-Read Only Memory）等を用いることができる。

【０１２８】

また、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。

【０１２９】

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。

【０１３０】

また、この形態では、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、ハードウェアエンティティ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。

【０１３１】

上述の本発明の実施形態の記載は、例証と記載の目的で提示されたものである。網羅的であるという意思はなく、開示された厳密な形式に発明を限定する意思もない。変形やバリエーションは上述の教示から可能である。実施形態は、本発明の原理の最も良い例証を提供するために、そして、この分野の当業者が、熟考された実際の使用に適するように本発明を色々な実施形態で、また、色々な変形を付加して利用できるようにするために、選ばれて表現されたものである。すべてのそのような変形やバリエーションは、公正に合法的に公平に与えられる幅にしたがって解釈された添付の請求項によって定められた本発明のスコープ内である。

【図１Ａ】

入力される表L_s

No.	身長(cm)	体重(kg)
3	200	100
5	110	19
9	160	85

図１Ａ

【図１Ｂ】

入力される表R_s

No.	購入品
3	おいしい水
7	ミックスオレ
9	傷薬
9	おいしい水

図１Ｂ

【図１Ｃ】

出力される表J_s

No.	身長(cm)	体重(kg)	購入品
3	200	100	おいしい水
9	160	85	傷薬
9	160	85	おいしい水

図１Ｃ

【図２Ａ】

入力される表L

$[[p_1]]$	$[[v_{1,2}]]$	…	$[[v_{1,a}]]$
$[[p_2]]$	$[[v_{2,2}]]$	…	$[[v_{2,a}]]$
…	…	…	…
$[[p_m]]$	$[[v_{m,2}]]$	…	$[[v_{m,a}]]$

図２Ａ

【図２Ｂ】

入力される表R

$[[q_1]]$	$[[u_{1,2}]]$	…	$[[u_{1,b}]]$
$[[q_2]]$	$[[u_{2,2}]]$	…	$[[u_{2,b}]]$
…	…	…	…
$[[q_n]]$	$[[u_{n,2}]]$	…	$[[u_{n,b}]]$

図２Ｂ

【図２Ｃ】

出力される表J

$[[q'_1]]$	$[[v'_{1,2}]]$	…	$[[v'_{1,a}]]$	$[[u'_{1,2}]]$	…	$[[u'_{1,b}]]$
$[[q'_2]]$	$[[v'_{2,2}]]$	…	$[[v'_{2,a}]]$	$[[u'_{2,2}]]$	…	$[[u'_{2,b}]]$
…	…	…	…	…	…	…
$[[q'_n]]$	$[[v'_{n,2}]]$	…	$[[v'_{n,a}]]$	$[[u'_{n,2}]]$	…	$[[u'_{n,b}]]$

図２Ｃ

【図３】

秘密等結合アルゴリズム（第一実施形態）

Input: 表L, R
Output: 表J
1: $<\sigma>\leftarrow$sort($[[p_1]],..,[[p_m]],[[q_1]],..,[[q_n]],[[p_1]],..,[[p_m]]$)
2: for j=2 to a do
3: $[[f]]\leftarrow([[v_{1,j}]],..,[[v_{m,j}]],\underbrace{[[0]],...,[[0]]}_{\text{n個}},[[-v_{1,j}]],..,[[-v_{m,j}]])$
4: $[[g]]\leftarrow[[\sigma([[f]])]]$
5: $[[g']]\leftarrow$PrefixSum($[[g]]$)
6: $[[f']]\leftarrow[[\sigma^{-1}([[g']])]]$
7: $([[v'_{1,j}]],..,[[v'_{n,j}]])\leftarrow([[f'_{m+1}]],..,[[f'_{m+n}]])$
8: end for
9: $[[f1]]\leftarrow([[1]],...,\underbrace{[[1]]}_{\text{m個}},\underbrace{[[0]],...,[[0]]}_{\text{n個}},\underbrace{[[1]]}_{\text{m個}},...,[[1]])$
10: $[[g1]]\leftarrow[[\sigma([[f1]])]]$
11: $[[g1']]\leftarrow$PrefixSum($[[g1]]$)
12: $[[f1']]\leftarrow[[\sigma^{-1}([[g1']])]]$
13: $([[e_1]],..,[[e_n]])\leftarrow([[f1'_{m+1}]],..,[[f1'_{m+n}]])$
14: for j=2 to b do
15: for i=1 to n do
16: $[[u'_{i,j}]]\leftarrow[[e_i]]\times[[u_{i,j}]]$
17: end for
18: end for
19: for i=1 to n do
20: $[[q'_i]]\leftarrow[[e_i]]\times[[q_i]]$
21: end for

図３

【図４】

出力される表J（平文）

No.	身長(cm)	体重(kg)	購入品
3	200	100	おいしい水
0	0	0	0
9	160	85	傷薬
9	160	85	おいしい水

図４

【図５】

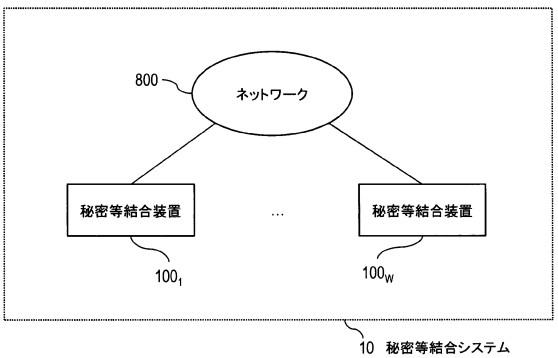

図５

【図６】

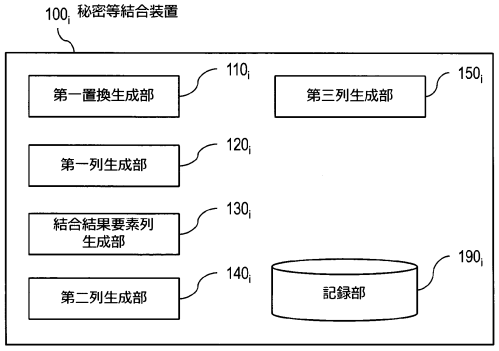

図６

【図７】

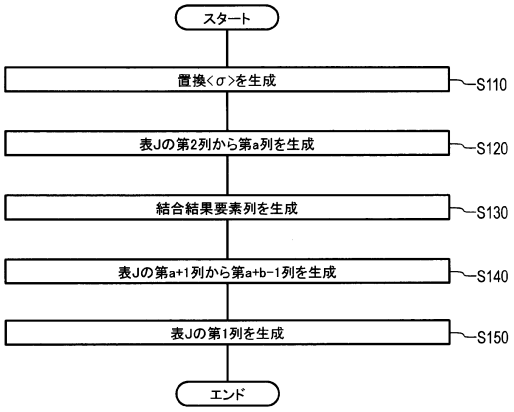

図７

【図８】

秘密等結合アルゴリズム（第二実施形態）

Input: 結合結果要素列($[[e_1]],..,[[e_n]]$)と表J
Output: 表J'（要素がすべて[[0]]である行を下に寄せたもの）
1: $<\sigma\sim>\leftarrow sort([[e_1]],..,[[e_n]])$
2: $([[q''_1]],..,[[q''_n]])\leftarrow[[\sigma\sim([[q'_1]],..,[[q'_n]])]]$
3: for j=2 to a do
4: $([[v''_{1,j}]],..,[[v''_{n,j}]])\leftarrow[[\sigma\sim([[v'_{1,j}]],..,[[v'_{n,j}]])]]$
5: end for
6: for j=2 to b do
7: $([[u''_{1,j}]],..,[[u''_{n,j}]])\leftarrow[[\sigma\sim([[u'_{1,j}]],..,[[u'_{n,j}]])]]$
8: end for

図８

【図９】

出力される表J'（平文）

No.	身長(cm)	体重(kg)	購入品
3	200	100	おいしい水
9	160	85	傷薬
9	160	85	おいしい水
0	0	0	0

図９

【図１０】

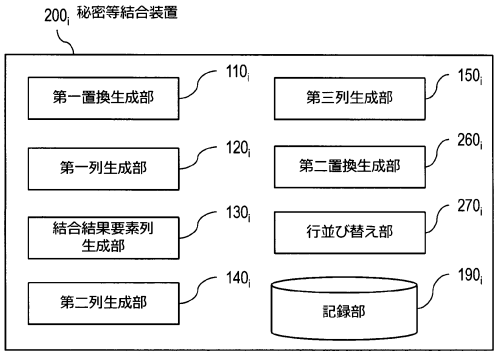

図１０

【図１１】

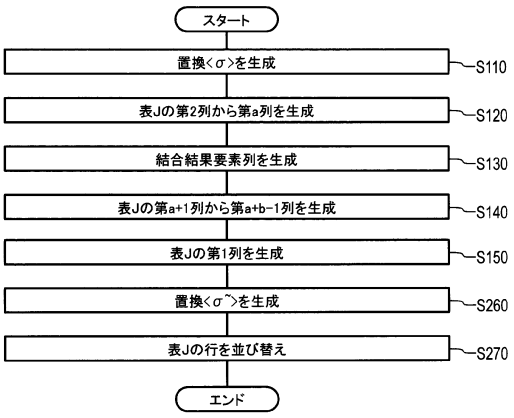

図１１

【図１２】

秘密等結合アルゴリズム（第三実施形態）

Input: 等結合アルゴリズム（第二実施形態）の($[[e_1]],..,[[e_n]]$)と表J'
Output: 表J''（要素がすべて0である行を除去したもの）
1: $[[c]]\leftarrow\Sigma_{i=1,..,n}[[e_i]]$
2: [[c]]を復元して得られるcを公開、表J'の上からc行だけを出力

図１２

【図１３】

出力される表J''（平文）

No.	身長(cm)	体重(kg)	購入品
3	200	100	おいしい水
9	160	85	傷薬
9	160	85	おいしい水

図１３

【図１４】

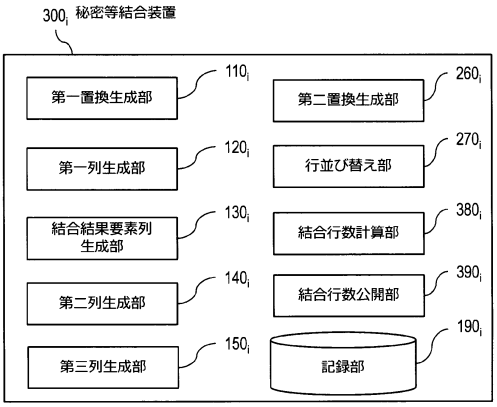

図１４

【図１５】

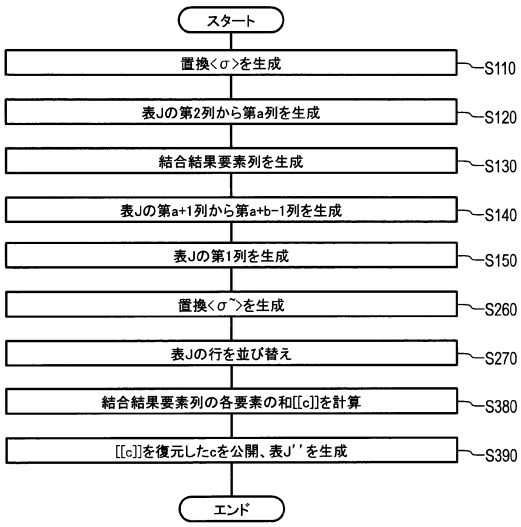

図１５

フロントページの続き

(72)発明者　濱田　浩気
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
(72)発明者　諸橋　玄武
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

(56)参考文献　特開２０１４－１３９６４０（ＪＰ，Ａ）
特開２０１３－２００４６１（ＪＰ，Ａ）
国際公開第２０１６／１１４３０９（WO，Ａ１）
国際公開第２０１５／１０７９５１（WO，Ａ１）
米国特許出願公開第２０１３／０１７９６８４（ＵＳ，Ａ１）
米国特許第６４５７０００（ＵＳ，Ｂ１）
桐淵　直人　ほか，属性情報と履歴情報の秘匿統合分析に向けた秘密計算による高速な等結合アルゴリズムとその実装，コンピュータセキュリティシンポジウム２０１６論文集，日本，一般社団法人情報処理学会，２０１６年１０月　４日，第2016巻，第2号，p.1072－1078

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０９Ｃ　　１／００