実用的な速度で統計分析が可能な秘密計算システム MEVAL

濱田 浩気　　五十嵐 大　　菊池 亮　　千田 浩司　　諸橋 玄武　　冨士 仁
高橋 克巳

NTT セキュアプラットフォーム研究所
180-8585 東京都武蔵野市緑町 3-9-11
hamada.koki@lab.ntt.co.jp

あらまし 近年のデータ分析技術の進歩と個人の情報の保護に対する意識の高まりに伴い，データの活用と保護の両立が求められてきている．これを実現する手法として，情報を暗号化などにより秘匿したまま計算することができる，秘密計算と呼ばれる技術が知られている．秘密計算を用いることで，情報を一切開示することなく任意の分析処理ができるが，通常の計算機上での計算に比べて非常に計算コストが大きいことが実用上の大きな課題となっている．我々は秘密計算上で統計分析を高速に行うために，アルゴリズムの改良と効率の良い実装を行なってきた．本稿では我々が開発した秘密計算システム MEVAL (Multi-party EVALuator) を紹介し，その性能を評価した結果を報告する．

MEVAL: A Practically Efficient System for Secure Multi-party Statistical Analysis

Koki Hamada　　Dai Ikarashi　　Ryo Kikuchi　　Koji Chida
Gembu Morohashi　　Hitoshi Fuji　　Katsumi Takahashi

NTT Secure Platform Laboratories
3-9-11, Midori-cho Musashino-shi, Tokyo 180-8585 Japan
hamada.koki@lab.ntt.co.jp

Abstract With the recent growth of information technology, the use of personal data has been getting common. On the other hand, awareness about privacy issues has been growing, and systems that use sensitive data without breaching privacy are needed. Secure multi-party computation (MPC) is one of the techniques that realize such secure systems, however, its inefficient performance has been a bottleneck for practical use. In this paper, we propose a MPC system MEVAL (Multi-party EVALuator). We also report some experimental results on our system.

1　はじめに

近年，個人に関する様々な情報を容易に取得できる環境が整い，データ分析技術の進歩と相まって，個人に関する情報の活用への期待が高まっている．その一方で，個人情報保護やプライバシの観点から個人に関する情報は極めて慎重な取扱いが必要とされ，データの保護と活用をどう両立させるかが問題となっている．

このようなデータの保護と活用の両立を目指して，プライバシ保護データ分析 (Privacy-Preserving Data Analysis: PPDA) 技術の研究が盛んになってきている．PPDA の有力なアプローチの一つに，秘密計算や秘匿関数計算，マ

ルチパーティプロトコルと呼ばれる，暗号学に基づいた手法がある．秘密計算は Yao による基本的なアイディア [22] を端緒とする技術であり，暗号化などの方法でデータを秘匿化したまま一度も元のデータに戻すことなく任意の計算を行う．秘密計算を用いることで，通常のデータ分析と同等の結果を高いプライバシ保護の下で得ることができる．しかし，秘密計算は通常の計算機上の計算に比べて処理速度が低下してしまうことが実用上の課題となっている．その原因は大きく二つある．

一つは，基本演算のオーバーヘッドである．秘密計算ではデータの秘匿性を保つために，乗算のような通常の計算機では一命令で実行可能な基本演算にも複雑な処理を必要とする．その結果，処理時間も大きくなってしまう．

もう一つは，回路に基づいた構成による計算量の悪化である．秘密計算は Goldreich ら [12] や Ben-Or ら [4] により提案された回路に基づく構成方法を用いることで，任意の計算を実現することができる．この構成方法では，所望の計算を論理回路で表現し，秘密計算上の基本的な演算の組み合わせで論理回路を模倣することにより任意の計算をデータを秘匿したまま行う．しかしながら，この回路に基づいた一般的な構成方法を用いると，実用上重要な多くのアルゴリズムで通常の計算機上での計算に比べて計算量が大きくなってしまう．

これらの問題に対し，我々は効率のよい基本演算を備えたシステムの設計・実装と，効率的な秘密計算上のアルゴリズムの設計を行ってきた [24]．本稿では，我々がさらに改良を進めてきた秘密計算システム MEVAL (Multi-party EVALuator) を紹介し，その性能を評価した結果を報告する．

1.1 関連研究

近年の秘密計算の効率の改善に伴い，秘密計算フレームワークの提案が進んでおり，Ben-David らによる FairplayMP [3]，Bogdanov らによる Sharemind [5]，Burkhart らによる SEPIA [8]，Henecka らによる TASTY [14]，Geisler による VIFF [10] などが提案されている．

特定の重要な演算に対する効率的なアルゴリズムの研究も進んでおり，Aggarwal らによる k 番目の要素の選択 [1]，Damgård らによるビット分解 [9]，Nishide と Ohta による比較 [18]，Ning と Xu による剰余計算 [17]，Hamada らによるソート [13] などが提案されている．

また，秘密計算を使った実験も行われるようになってきており，Bogetoft らによるテンサイ (サトウダイコン) のオークション [7]，Burkhart らによるネットワークの異常検知 [8] などが報告されている．

1.1.1 データ加工に基づく PPDA

秘密計算とは別の PPDA の有力なアプローチとして，たとえ公開されてもプライバシ侵害が起こらない程度にまでデータを加工した上で分析を行う手法がある．Sweeney による k-匿名化 [21] や Agrawal と Srikant による再構築法 [2] が代表的である．これらの手法は秘密計算に比べると計算コストが非常に小さい．また，データを非可逆な方法で加工することによりプライバシ保護を実現するため，加工後のデータを公開可能であるという特長を持つ．一方，その非可逆性により，加工前の元のデータに対して行った場合と完全に同等の分析結果を得ることは不可能であり，加工後のデータを用いた分析結果の精度をいかにして担保するかが課題となっている．

1.2 論文の構成

まず 2 節で我々の用いた秘密計算技術についての概念的な説明を行う．続いて 3 節で我々の開発した秘密計算システム MEVAL の紹介を行う．最後に 4 節で MEVAL に実装したアルゴリズムの詳細を述べ，さらに性能評価の結果を報告する．

2 秘密計算の実現方法

秘密計算は，暗号化などの方法でデータを秘匿したまま，一度も元のデータに戻すことなく計算を行う技術である．本節では，秘密計算システム MEVAL で使っている秘密計算の実現方法について，概念的な説明を行う．

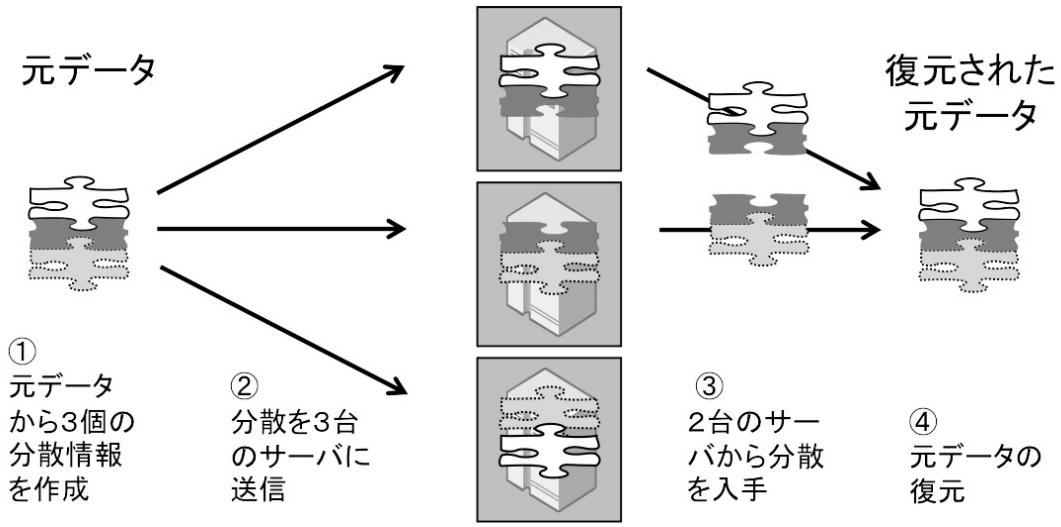

図 1: $(2,3)$ 閾値秘密分散の概念図

2.1 秘密分散

秘密分散は，秘密にしたい情報を複数の分散情報に分けることによって保護する手法である．分散情報は，一定数集めると情報を復元することができ，逆に一定数集めない限り秘密が漏れることがないように作られる．

秘密分散は，n 人の参加者 $P_1,\ldots,P_n$ とディーラ D によって行われるプロトコルであり，**分散**と**復元**の 2 つの処理から構成される．分散の処理においては，ディーラ D は秘密 s からシェアと呼ばれる $(v_1,\ldots,v_n)$ の n 個の分散情報を作成し，各参加者 P_i にシェア v_i を与える．復元の処理においては，n 人の参加者のうち何人かでシェアを持ち寄り，シェアから秘密の情報 s を求める．復元の処理でどの参加者が持ち寄ったときに秘密が復元できるかどうかの条件は秘密分散ごとに定められる．最もよく用いられる条件は，「$v_1,\ldots,v_n$ のうち，任意の k 個から秘密 s が復元できるが，どの $k-1$ 個を集めても s について何もわからない」という条件である．この条件を満たす秘密分散を特に (k,n) **閾値秘密分散**と呼ぶ．

図 1 は，情報を 3 つに分散させ，任意の 2 つを集めることによって復元できる，$(2,3)$ 閾値秘密分散の概念図である．

2.2 秘密計算

秘密計算は，暗号化や秘密分散などによって秘匿されたデータに対し，データを秘匿したまま各種の計算を行う技術である．

2.2.1 利用モデル: 委託型秘密計算

秘密計算システム MEVAL は，**委託型秘密計算**と呼ばれるモデルの秘密計算を実現している．委託型秘密計算は，次のような要求に応える秘密計算のモデルである．データを所有している主体 (**提供者**) と別の主体 (**分析者**) がおり，分析者は提供者の持つデータを分析したいが，提供者のプライバシ保護のため，提供者の持つデータは必要以上に開示したくない．

委託型秘密計算は，上述の提供者と分析者に**計算者**と呼ばれる主体を加えて構成される．委託型秘密計算システムの概念図を図 2 に示す．まず，提供者の持つ入力データは秘密分散されて複数のコンピュータからなる計算者に配置される．計算者は，分散データの中身を閲覧することなしに，あらかじめ定められた手順に従ってデータ処理を行い，最終的に求める計算結果を，結果の値の分散情報として得る．最後に，計算結果は秘密分散と同様に，分析者に必要な数の分散情報を集めて復元される．この一連の計算過程において，提供者以外のどの主体もデータ

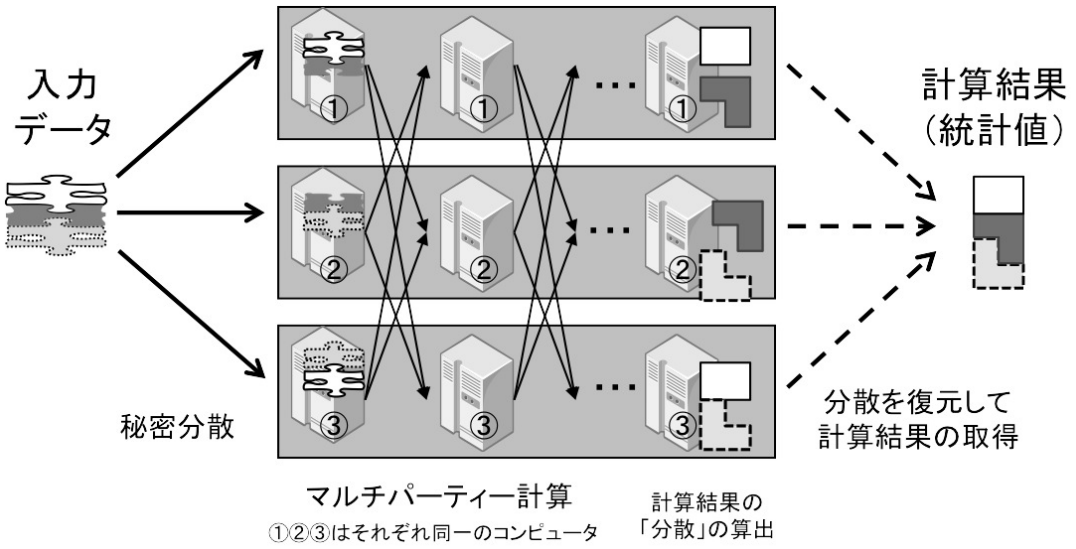

図 2: 委託型秘密計算システムの概念図

に関する一切の情報が得られず，また，計算者は計算結果のみを得ており，提供者の持つデータを必要以上に開示しないという要求は満たされている．

委託型秘密計算で計算者により行われる処理は，**マルチパーティ計算**と呼ばれる処理を繰り返し行うことにより実現される．マルチパーティ計算は，秘密分散された分散情報を入力として，別の秘密分散された分散情報を計算する処理である．行うデータ処理に応じて，コンピュータ間で必要なデータ処理とデータの送受信 (通信) を必要回数行う．この際，各コンピュータにおいて，他から受信したデータと各コンピュータが保有する分散情報を合わせても各コンピュータは何も元データに関する情報を得られないように作られる．入力も出力も秘密分散された分散情報という同じ形式であるため，異なる演算を実現する複数のマルチパーティ計算を組み合わせてより複雑な処理を実現するマルチパーティ計算を作ることができる．

なお，利用可能な演算の種類やデータの範囲をユーザー毎に制御する機能は委託型秘密計算システムとして重要ではあるが，秘密計算システムの中心的な機能であるマルチパーティ計算部分とは独立に議論可能な内容であるため，別の方法で実現されるものとして，本稿では扱わない．

2.2.2 マルチパーティ計算の具体例

加算 秘密計算は，独特の方法で算術演算を実現する．まず始めに秘密計算による加算の実現方法を説明する．図 3 は秘密計算で 2 つのテストの点 (整数値) の加算 (足し算) を行う概念を説明する例である．ここではテストの点は適当な数字 (乱数) を使って，3 つの数の和で表現し，3 つの数のうちの異なる 2 つずつを分散情報として，秘密分散される．各コンピュータでは，各々が分散データを使ってローカルな加算器として振る舞い，全体の足し算の秘密分散を得る．

任意の計算の実現 秘密計算では，いくつかの基本的な演算のみを用意し，その他の計算は基本的な演算の組み合わせにより実現する．任意の計算を実現可能な基本的な演算の組の選び方は多くあるが，例えば，加算と乗算は，その組み合わせにより任意の計算を実現できることが知られている．すなわち，加算に加えて乗算を実現するマルチパーティ計算を用意し，これらを適切に組み合わせることで，任意の計算処理のマルチパーティ計算を実現できる．

3 秘密計算システム MEVAL

本節では，我々が設計・実装した秘密計算システム MEVAL (Multi-party EVALuator) の

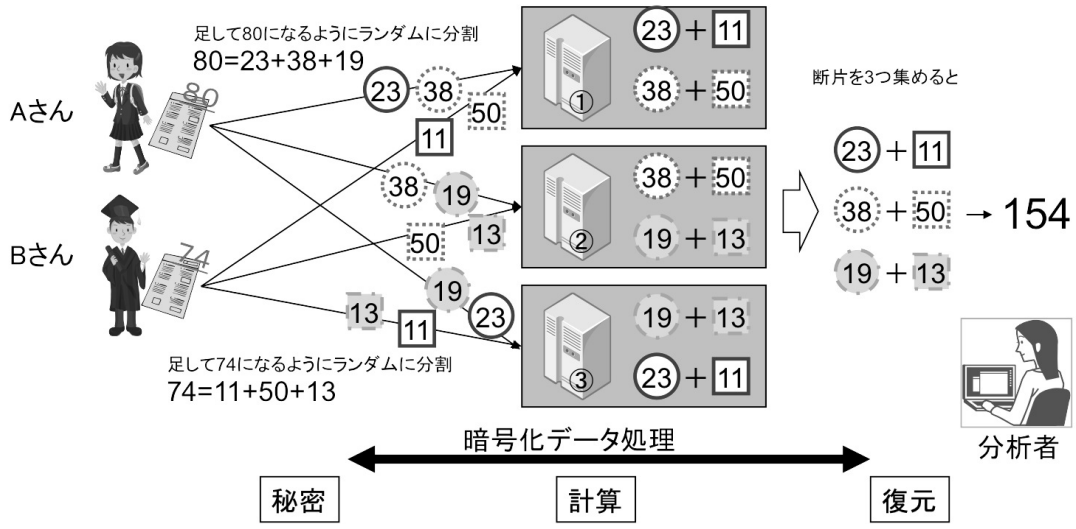

図 3: マルチパーティ計算による加算の例

説明を行う．

3.1 システム構成

秘密計算システム MEVAL は 2.2.1 節で述べた委託型秘密計算を実現している．すなわち，提供者，分析者，計算者に相当する複数のコンピュータで構成される．MEVAL では，提供者が 1 台，分析者が 1 台，計算者が 3 台のコンピュータでそれぞれ実現され，これらは互いにネットワークで接続される．1 台のコンピュータが，提供者や 分析者，計算者のうちのいくつかを兼ねることもできる．

3.2 操作イメージ

秘密計算システム MEVAL を用いて分析を行う際には，ユーザーは分析者コンピュータを通して操作を行う．ユーザが計算対象のデータと所望の演算を指定すると，分析者コンピュータから 3 台の計算者コンピュータに実行要求が送られ，秘密分散された対象データに対して 3 台の計算者コンピュータで要求された演算のマルチパーティ計算が行われ，最後に分析者コンピュータに計算結果のみが返される．

ユーザに提供される分析者コンピュータのインターフェースは，Microsoft Excel に基づいたクライントと，R に基づいたクライアントの

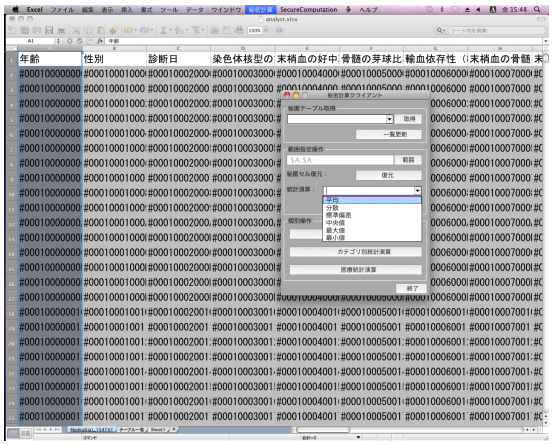

図 4: Excel に基づくユーザーインターフェース

2 種類が用意されている．Excel に基づいたクライントの場合は，図 4 のように所望の演算と計算対象にしたいセルを選択して秘密計算を実行する．R に基づいたクライントの場合は，図 5 のように，秘密分散された入力データの表を対象とした統計値を計算する関数がユーザ定義関数として用意されており，これを呼び出して対話的に秘密計算を実行する．

3.3 MEVAL の提供する演算

秘密計算システム MEVAL は以下の演算を提供する:

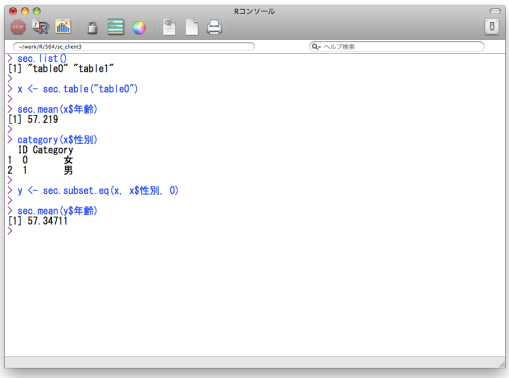

図 5: R に基づくユーザーインターフェース

- 秘密分散，復元
- 基本統計値 (平均値，分散値，最小値，最大値，中央値) 計算
- 条件検索
- 集約関数 (集計，平均値，分散値，最小値，最大値，中央値)
- Kaplan-Meier 法 [15]
- ランダム置換

4　実装したアルゴリズムと性能

4.1　前提

秘密計算システム MEVAL の実装した秘密計算は秘密分散に基づいており，秘密分散には，Shamir による $(2,3)$ 秘密分散 [20] を使っている．すべての値は有限体 $\mathbb{Z}_p$ の要素で，素数 p は $p = 2^{61} - 1$ である．また，すべてのプロトコルは semi-honest (honest-but-curious) である攻撃者に対して情報理論的安全である．

4.2　具体的なアルゴリズム

MEVAL の提供する演算は，分散，復元，加算，減算，低数倍，乗算，内積，ビット分解，等号判定，比較，ランダム置換，ソートをそれぞれ実現するマルチパーティ計算の組み合わせにより実現している．MEVAL では，分散と復元には Shamir による $(2,3)$ 秘密分散 [20] の分散と復元を，加算，減算，定数倍は Shamir の秘密分散の準同型性を使った標準的な手法 [19] を，乗算は Gennaro らの手法 [11] を，内積の計算は Gennaro らの乗算 [11] に基づく標準的な手法 [19] を，ビット分解は五十嵐らの手法 [25] を，等号判定および比較はビット分解と論理回路の組み合わせによる手法を，ランダム置換は Laur らの手法 [16] を，ソートは濱田らの手法 [23] を，それぞれ実装した．

4.3　処理性能

4.3.1　測定環境

測定は，3 台の計算者コンピュータのうち 1 台が提供者と分析者を兼ねて行った．3 台の計算者コンピュータは CPU が Intel Core i7 3930K 3.2 GHz，メモリが 20 GB，SSD が 128 GB，OS が Linux(Ubuntu 12.04) のデスクトップ PC である．3 台のコンピュータは有線のネットワークにより互いに接続した．回線の環境は，1 Gbps LAN 環境，10 Gbps LAN 環境，インターネット環境の 3 種類で測定を行った．インターネット環境は，3 台のコンピュータごとに異なる地点で家庭用の光ファイバー回線 (ベストエフォートで 200 Mbps) を使い，2 地点間の遅延はそれぞれ，24.6 ms，36.1 ms，46.7 ms であった．

提供者，計算者，分析者の各コンピュータ上のプログラムは C++言語で実装を行った．

等号判定，大小比較，ソートについては，秘密の値が 20 bit である (すなわち，秘密の値 s が $0 \leq s < 2^{20}$ を満たす) 入力を用いた．

4.3.2　測定結果

加算，乗算，シャッフル，等号判定，大小比較，ソートの各演算について，環境と入力件数を変えながら実行時間を測定した．入力件数は，LAN 環境では，100,000 件，1,000,000 件，10,000,000 件，100,000,000 件の 4 条件，インターネット環境では，1 件，100 件，1547 件，10,829 件，108,290 件の 5 条件の下でそれぞれ測定を行った．

1 Gbps LAN 環境，10 Gbps LAN 環境，インターネット環境での測定結果をそれぞれ表 1，

表 1: 1 Gbps LAN 環境での実行時間 (ミリ秒)

件数	100,000	1,000,000	10,000,000	100,000,000
加算	1.3	1.5	12.4	138.4
乗算	17.4	135.1	1,191.4	11,448.9
シャッフル	31.4	233.9	2,603.2	29,073.0
等号判定	838.9	667.6	879.8	9,023.6
大小比較	431.0	286.8	592.0	13,680.4
ソート	738.0	6,875.2	73,381.6	-

表 2: 10 Gbps LAN 環境での実行時間 (ミリ秒)

件数	100,000	1,000,000	10,000,000	100,000,000
加算	1.3	1.4	11.9	138.9
乗算	17.0	50.2	469.1	4,752.0
シャッフル	20.1	118.0	1,314.9	15,073.0
等号判定	710.1	663.6	674.2	2,688.9
大小比較	322.1	262.9	287.4	1,699.4
ソート	253.0	2,210.9	30,207.4	-

表 2，表 3 に示す．インターネット環境においても，1,000 件程度のデータに対しては 3.3 節のいずれの演算も 1 秒程度で計算可能であり，MEVAL はインターネット環境でも実用的な性能を持つと言える．著者らが知る限り現在のところ最も性能の良い実装である Bogdanov らによる Sharemind の最近の報告 [6] と 1 Gbps LAN での測定結果で比較しても，乗算 (Sharemind は 32 bit，MEVAL は 61 bit) で約 10 倍，等号判定 (Sharemind は 32 bit，MEVAL は 20 bit) で約 100 倍，比較 (Sharemind は 32 bit，MEVAL は 20 bit) で約 400 倍高速であった．

5 おわりに

本稿では，我々が開発した秘密計算システム MEVAL を紹介し，その性能測定結果を報告した．MEVAL は，インターネット環境においても 1,000 件程度のデータの分析が実用的な時間で行え，また，1 Gbps LAN 環境では，現在最も性能のよい Sharemind の測定結果と比較しても，演算の種類によって 10 倍から 400 倍程度高速である．

今後は五十嵐らが提案した active な攻撃者に対しても安全な秘密計算 [26] の実装を進めていく予定である．

参考文献

[1] Gagan Aggarwal, Nina Mishra, and Benny Pinkas. Secure computation of the k th-ranked element. In *EUROCRYPT*, pp. 40–55, 2004.

[2] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. In *SIGMOD Conference*, pp. 439–450. ACM, 2000.

[3] Assaf Ben-David, Noam Nisan, and Benny Pinkas. Fairplaymp: a system for secure multi-party computation. In *ACM Conference on Computer and Communications Security*, pp. 257–266. ACM, 2008.

[4] Michael Ben-Or, Shafi Goldwasser, and Avi Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation (extended abstract). In *STOC*, pp. 1–10. ACM, 1988.

[5] Dan Bogdanov, Sven Laur, and Jan Willemson. Sharemind: A framework for fast privacy-preserving computations. In *ESORICS*, Vol. 5283 of *LNCS*, pp. 192–206. Springer, 2008.

表 3: インターネット環境での実行時間 (ミリ秒)

件数	1	100	1,547	10,829	108,290
加算	-	0.580	0.603	0.667	2.005
乗算	-	91	63	74	233
シャッフル	-	59	62	125	671
等号判定	970	930	1,030	1,591	5,468
大小比較	634	771	961	1,647	6,174
ソート	1,075	1,032	772	1,595	12,723

[6] Dan Bogdanov, Margus Niitsoo, Tomas Toft, and Jan Willemson. High-performance secure multiparty computation for data mining applications. *Int. J. Inf. Sec.*, Vol. 11, No. 6, pp. 403–418, 2012.

[7] Peter Bogetoft, Dan Lund Christensen, Ivan Damgård, Martin Geisler, Thomas P. Jakobsen, Mikkel Krøigaard, Janus Dam Nielsen, Jesper Buus Nielsen, Kurt Nielsen, Jakob Pagter, Michael I. Schwartzbach, and Tomas Toft. Secure multiparty computation goes live. In *Financial Cryptography*, Vol. 5628 of *LNCS*, pp. 325–343. Springer, 2009.

[8] Martin Burkhart, Mario Strasser, Dilip Many, and Xenofontas A. Dimitropoulos. Sepia: Privacy-preserving aggregation of multi-domain network events and statistics. In *USENIX Security Symposium*, pp. 223–240. USENIX Association, 2010.

[9] Ivan Damgård, Matthias Fitzi, Eike Kiltz, Jesper Buus Nielsen, and Tomas Toft. Unconditionally secure constant-rounds multi-party computation for equality, comparison, bits and exponentiation. In *TCC*, pp. 285–304, 2006.

[10] Martin Geisler. *Cryptographic Protocols: Theory and Implementation*. PhD thesis, University of Aarhus, 2010.

[11] Rosario Gennaro, Michael O. Rabin, and Tal Rabin. Simplified vss and fact-track multiparty computations with applications to threshold cryptography. In *PODC*, pp. 101–111. ACM, 1998.

[12] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or a completeness theorem for protocols with honest majority. In *STOC*, pp. 218–229. ACM, 1987.

[13] Koki Hamada, Ryo Kikuchi, Dai Ikarashi, Koji Chida, and Katsumi Takahashi. Practically efficient multi-party sorting protocols from comparison sort algorithms. In *ICISC*, Vol. 7839 of *LNCS*, pp. 202–216. Springer, 2012.

[14] Wilko Henecka, Stefan Kögl, Ahmad-Reza Sadeghi, Thomas Schneider, and Immo Wehrenberg. Tasty: tool for automating secure two-party computations. In *ACM Conference on Computer and Communications Security*, pp. 451–462. ACM, 2010.

[15] E L Kaplan and Paul Meier. Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, Vol. 53, No. 282, pp. 457– 481, 1958.

[16] Sven Laur, Jan Willemson, and Bingsheng Zhang. Round-efficient oblivious database manipulation. In *ISC*, Vol. 7001 of *LNCS*, pp. 262–277. Springer, 2011.

[17] Chao Ning and Qiuliang Xu. Multiparty computation for modulo reduction without bit-decomposition and a generalization to bit-decomposition. In *ASIACRYPT*, pp. 483–500, 2010.

[18] Takashi Nishide and Kazuo Ohta. Multiparty computation for interval, equality, and comparison without bit-decomposition protocol. In *PKC*, pp. 343–360, 2007.

[19] SecureSCM. Security analysis. Deliverable D9.2, SecureSCM Project, 2009.

[20] Adi Shamir. How to share a secret. *Commun. ACM*, Vol. 22, No. 11, pp. 612–613, 1979.

[21] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 10, No. 5, pp. 557–570, 2002.

[22] Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In *FOCS*, pp. 162–167, 1986.

[23] 濱田浩気, 五十嵐大, 千田浩司, 高橋克巳. 秘匿関数計算上の線形時間ソート. In *SCIS*, pp. 1–7, 2011.

[24] 濱田浩気, 大竹茂樹, 五十嵐大, 竹之内大地, 千田浩司, 冨士仁, 高橋克巳, 村田節子, 熊田総佳. 秘匿関数計算システムによる医療データのプライバシ保護統計分析. 信学技報, 第 111 巻 of *LOIS2011-102*, pp. 177–181, 2012.

[25] 五十嵐大, 濱田浩気, 菊池亮, 千田浩司. 少パーティ秘密分散ベース秘密計算における高速なビット分解法と modulus 変換法. In *CSS*, 2013.

[26] 五十嵐大, 濱田浩気, 菊池亮, 千田浩司. 非常に高効率な $n \geq 2k-1$ の malicious モデル上秘密分散ベース秘密計算. In *SCIS*, pp. 1–6, 2013.