実数演算可能な軽量秘密計算法の一考察

金岡 晃 † 宮西 洋太郎 ‡ 韓 嘯公 § 北上 眞二 § 佐藤 文明 †
浦野 義頼 § 白鳥 則郎 §

† 東邦大学
274-8510 千葉県船橋市三山 2-2-1
{akira.kanaoka, fsato}@is.sci.toho-u.ac.jp

‡ アイエスイーエム
103-0008 東京都中央区日本橋中洲 1-5-1103
ymiyanishi@isem.co.jp

§ 早稲田大学
169-0051 東京都新宿区西早稲田 1-3-10
hanxiaogong@ruri.waseda.jp, kitagami@gem.cugat.net,
urano@waseda.jp, norio@shiratori.riec.tohoku.ac.jp

あらまし クラウドコンピューティング環境等に応用が期待される秘密計算手法はさまざまな手法が提案されているがそのほとんどは整数（有限体）の演算を対象としているものである。近年になり実数の演算に対する秘密計算手法がいくつか提案されているが、それらは整数演算の秘密計算手法を応用したものであり計算量の点では改善が必要となっている。本稿では実数演算、とくに乗算と除算に対する秘密計算手法として簡潔な手法を提案し、それらを評価する。

A Study of Lightweight Secure Multi-party Computation enabling Real Number Arithmetic

Akira Kanaoka† Yohtaro Miyanishi ‡ Xiaogong Han § Shinji Kitagami §
Fumiaki Sato † Yoshiyori Urano § Norio Shiratori§

†Toho University
2-2-1, Miyama, Funabashi, Chiba 274-8510 Japan
{akira.kanaoka, fsato}@is.sci.toho-u.ac.jp

‡ISEM, Inc.
1-5-1103, Nihonbashinakasu, Chuo-ku, Tokyo 103-0008 Japan
ymiyanishi@isem.co.jp

§Waseda University
1-3-10, Nishiwaseda, Sinjuku-ku, Tokyo 169-0051 Japan
hanxiaogong@ruri.waseda.jp, kitagami@gem.cugat.net,
urano@waseda.jp, norio@shiratori.riec.tohoku.ac.jp

Abstract Secure multi-party computation (SMC) which is expected to apply secure cloud computing environment, is widely studied. However, existing works mostly focused on finite field arithmetic. Since, several works about real number arithmetic for SMC are proposed in recent years, these works are still heavy because of applying current finite field arithmetic based SMC. In this paper, a simple SMC method for real number arithmetic especially multiplication and division, is proposed and evaluated.

1　はじめに

ネットワークに接続される機器は、PC が主体だった時代から様々な機器が柔軟に接続する時代へと移りつつある。ネットワークに接続することにより、機器そのものが持っていないリソースを外部から利用することで様々な情報取得や機能獲得ができるようになる。クラウドコンピューティングはそういった時代背景に沿い発展をしてきている。またネットワークに接続される機器等から集まった大量のデータを解析することでよりよいサービスにつなげるといった研究開発もされてきている。

ネットワークに接続する機器は十分なリソースを持たずとも、外部に情報や演算を委託することで充実したサービスを受けられる。しかしそういった委託においては、問題が生じるケースも多い。委託により渡される情報や委託により得られた演算結果が、高い機密性のある情報や、非常に機微なプライバシーの情報である場合、委託先に情報が漏えいし、悪用されかねない。あるいは、大量に集まったデータによる知見によるサービス享受において、データを注意深く観察することで大量なデータの中に潜む個々の機微なデータを抽出されてしまう脅威もある。

こういった脅威に対し、データのセキュリティやプライバシを保護したまま委託を可能にする技術に注目があつまり、多方面から研究されている。一般的にプライバシ保護技術と呼ばれるそれらの技術は、入出力の暗号化や、暗号化したまま演算を可能にする手法、またはデータベースのデータ群を性質を損なうことなく匿名化を行う手法など多岐にわたる。

データを秘匿したままさまざまな演算を可能にする秘密計算（Secure Multiparty Computation、以下 SMC）はプライバシ保護技術の 1 つとして注目がされている。秘密計算では、計算の委託側がデータの分割等の秘匿化を行い、委託される側は複数のエンティティがそれぞれ秘匿化されたデータを用いて演算を行い、最終的に委託側が各エンティティからの演算結果を集約することで秘匿化を解く。

SMC はさまざまな手法がこれまで提案されてきたが、そのほとんどは演算対象となるデータは有限体上の元であった [1, 2, 3, 4, 5]。しかし、用途によっては有限体上の元だけではなく実数上の計算が必要となるものもある。たとえば各国の衛星の軌道をもとにした衝突可能性のシミュレーションでは、座標や速度など実数計算が求められる [6]。そういった実数上の秘密計算を実現した例はまだ少なく [7, 8, 9, 10]、実用上の課題は多く残る。

本論文では、実数上の秘密計算に着目しその実用性を高めるために、計算効率の高い乗算と除算の手法を提案する。提案手法は、実数の秘密分散をこれまでの有限体の秘密分散手法を応用して実数上で行い、それをもとに実数での乗算と除算を可能にする。秘密計算の根幹となる秘密分散を有限体上ではなく実数上で実現しているため、固定小数点表現や浮動小数点表現を問わず、ユーザは実数のデータ表現を気にすることなく直接的に演算が可能になる。

さらに、安全性と効率を評価しその高い効率を示す一方で、提案手法により起こりうる問題点についても考察を行うことで提案手法の適用分野等を検討する材料を提供する。

本論文の構成は以下の通りである。2 章において実数上の秘密計算のこれまでの研究について紹介を行い、3 章において軽量な実数上の秘密計算手法を提案する。4 章では提案手法の安全性と効率について評価を行い、5 章において提案手法の計算精度についての考察を行う。最後に 6 章でまとめる。

2　関連研究

Secure Multi-party Computation（SMC）は、さまざまな手法が提案されている。SMC の研究の初期では、その計算効率が大きな問題となっていたが、近年の手法では効率化が進んできている。一方で、これまでのほとんどの SMC の手法は、有限体を用いた手法となっている。

実数での演算の例としては、近年になり固定小数点による実数計算を実現した Catrina らの手法が提案され [7]、Franz らは限定的な問題がありながらも対数表現を可能とし、2-パーティーでの秘密計算を実現した [8]。さらに Franz ら

は浮動小数点で 4 つの基礎的な演算を 2-パーティーでの秘密計算を実現する手法を提案した [9]。しかしこの手法は実装はされていなかった。

そして 2013 年に、Aliasgari らにより浮動小数点での複数の演算を可能にする手法が提案され、さらに実装による評価が行われた [10]。Aliasgari らの手法は、浮動小数点のデータ表現と、浮動小数点を用いた各種の演算プロトコルを紐解き、そのそれぞれを従来の有限体ベースの SMC を応用して浮動小数点での秘密計算を可能とするものであり、加算、乗算、除算に加え比較演算（小なり、Less Than）、丸め（切り上げ、切り捨て）といった演算が提案され、また整数表現と浮動小数点表現の双方向のデータ変換、同じく固定小数点表現と浮動小数点表現のデータ変換が行われており、高い機能を実現している。

浮動小数点表現では、仮数と指数と符号により実数が表現され、実数上の演算は仮数同士の演算、指数同士の演算、仮数と指数を用いたサブ演算とも言える演算を組み合わせて実現される。それぞれのサブ演算は整数演算と考えることができるため、Aliasgari の手法ではそれらのサブ演算に対し従来の有限体ベースの SMC を適用する。多くのサブ演算については既存の手法が適用可能であるが、いくつかのサブ演算は従来の手法では対応できないため、新たに提案し、全体として浮動小数点表現での実数演算を可能にしている。

提案のみならず、計算回数や通信回数の評価や実装による実測評価もおこなっており、浮動小数点での SMC において大きな成果を上げている。

Aliasgari らの手法はその一方で、やはり演算の速度が問題となることがその結果から伺え、効率化が今後求められることとなる。

3 実数演算を可能にする軽量秘密計算手法

Aliasgari らの手法では、従来の有限体上での SMC の各手法を応用し浮動小数点で表現された実数演算の SMC を実現していた。有限体上での SMC の手法は、Shamir の秘密分散法を用いたものが基本となっている [11]。

直観的なアプローチとして、秘密分散の手法を有限体 (Z_n) ではなく実数 (R) で行い、それを秘密計算に応用していくことが考えられるが、われわれの調査したところでは実数上での秘密計算手法は見つからなかった。

本稿では、実数の秘密分散をこれまでの有限体の秘密分散手法を応用して実数上で行い、それをもとに実数での乗算と除算を可能にする秘密計算手法を提案する。

本提案手法では、秘密計算の根幹となる秘密分散を有限体上ではなく実数上で実現しているため、固定小数点表現や浮動小数点表現を問わず、ユーザは実数のデータ表現を気にすることなく直接的に演算が可能になる。

3.1 乗除算用のシェアの分割

乗除算用のデータの分割は、以下のように行う。もとのデータを x、分割数を n とした場合、

$$x_0 \cdot x_1 \cdot \cdots \cdot x_{n-1} = x \tag{1}$$

となるようにランダムに $x_i (i = 0, \ldots, n-2)$ を選択し、x_{n-1} は $x/\Pi_{i=0}^{n-2} x_i$ で求める。

3.1.1 n out of n のケース

すべてのシェアが集まることで秘密計算が可能になる n out of n での実現では、各パーティー P_i に x_i を配布する。

3.1.2 2 out of 3 のケース

2 つのパーティーが集まることで秘密計算が可能になる 2 out of 3 での実現では、各パーティー P_i に $x_i, x_{(i+1)mod3}$ を配布する。

3.2 乗算

もとのデータを x, y とし、$z = x \cdot y$ を求める場合、以下のケースに分かれる。

3.2.1 n out of n のケース

パーティー P_i は、依頼者に $z_i = x_i \cdot y_i$ を返す。依頼者は集まった z_i より $z = \Pi_{i=0}^{n-1} z_i$ を求める

3.2.2 2 out of 3 のケース

パーティー P_i, は、依頼者に $c_i = x_i \cdot y_i$、$c_j = x_{(i+1)mod3} \cdot y_{(i+1)mod3}$ とどのシェアを持っているかの情報である $(i, (i+1)mod3)$ を返す。依頼者は集まった c_i とシェアの情報から、$z = \Pi_{i=0}^{2} c_i$ を得る。

3.3 除算

もとのデータを x, y とし、$z = x/y$ を求める場合、以下のケースに分かれる。

3.3.1 n out of n のケース

パーティー P_i は、依頼者に $z_i = x_i/y_i$ を返す。依頼者は集まった z_i より $z = \Pi_{i=0}^{n1} z_i$ を求める

3.3.2 2 out of 3 のケース

パーティー P_i, は、依頼者に $c_i = x_i/y_i$、$c_j = x_{(i+1)mod3}/y_{(i+1)mod3}$ とどのシェアを持っているかの情報である $(i, (i+1)mod3)$ を返す。依頼者は集まった c_i とシェアの情報から、$z = \Pi_{i=0}^{2} c_i$ を得る。

3.4 四則演算の組み合わせ

シェアを加減算用と乗除算用の 2 種類用意することにより、組み合わせた四則演算の利用が可能になる。加減算用のシェアはすでに有限体で提案されている加算型の秘密分散を用いて実現する。乗除算の後の加減算といった復号計算については、乗除算終了後に加減算用のシェアを生成・配布を行い加減算を実施する。

4 提案手法の評価

4.1 安全性

本手法では、従来の SMC 研究と同様 Semi-honest モデルを採用する。Semi-honest モデルでは、各パーティーは自身に与えられたデータは閲覧するもののプロトコルで定められた動きは逸脱しない。

各パーティーの結託による情報の漏えいについては、n out of n のケースでは全てのパーティーが結託することで情報が漏えいする。2 out of 3 のケースでは 3 パーティーのうち 2 つのパーティーが結託することで情報が漏えいする。

各パーティーに与えられたシェアによる情報漏えいでは、いずれの方式も乱数を用いることでもとの情報の特徴を予測困難なものとしている。しかし本方式は実数での秘密計算手法であるため、実装を考慮すると乱数がとりうる値域と秘密計算対象のデータの値域を定めなければならない。値域の設定は演算種類、演算結果に求められる演算精度により異なる。値域設定の考察については 5 章で述べる。

4.2 計算効率

計算の効率については、同じく浮動小数点での SMC を実現した Aliasgari らの手法 [10] と、ラウンド数とオペレーションの回数数について比較する。

ラウンド数は、連続する呼び出し（invocation）の数で通信による遅延と関連する。ここでは通信の発生を 1 回と数えた。オペレーションの回数は、シェアの作成･配布、乗算、演算結果の収集と計算の回数である。これらの定義は Aliasgari らも用いており、Aliasagari らは Catrira らの論文の定義に従っている [12]。Catrira らの定義ではオペレーション回数で各パーティーの乗算回数を論じているが、本稿では除算も同様のオペレーション回数と数えた。

Aliasgari らによる秘密計算手法の乗算（FLMul）と除算（FLDiv）と提案手法の比較を表 1 に示す。ここでは、単精度浮動小数点数を対象とした。倍精度の場合、提案手法のラウンド数とオ

ペレーション回数に変化はないが、FLMul と FLDiv は大きく増加する。

5 計算精度についての考察

提案方式は乗除算を行う際に乱数を用いて分割したシェアを用いて演算を行うことで、もとの情報をパーティーに知らせることなく演算が可能となっている。一般的に、演算に使われるデータはそのデータ群の特性があり演算の対象となるデータの値は類似しているあるいは推測可能な値域に存在していることが多い。たとえば人間の身長データをを単位として扱う場合、その値域はこれまでの統計データからあるていどの推測が可能である。またたとえば 100,000cm といった外れ値も考えにくい。こういったケースでは身長の平均や中央値のデータもある値域に収まる。これを浮動小数点で考えると、似通ったデータ同士での演算であるため桁落ちや情報落ちの発生確率も十分に低いものであることが推測できる。

本方式では、乱数を用いて分割するシェアを用いることで、乱数によりもとのデータの値域特性を希薄させパーティーに特性把握をさせないようになっているが、乱数の値域設定によっては桁落ちや情報落ちが高頻度で発生することが考えられる。桁落ちや情報落ちが起きにくいシェアを作成するために、もとの情報の特性と利用される演算の特性を考慮し、影響が起きにくいように調整することは可能である。しかしこういった調整はパーティーに対する情報特性隠避の効果を薄れさせる。そのため、適切な乱数値域の設定は利用されるデータセットのデータ特性と、SMC によって計算される計算の内容と、それを用いたアプリケーションで求められる情報の精度に強く依存する。これらはトレードオフの関係にあることが予想される。

いくつかをパラメータ化し、トレードオフを可視化することも検討できるが、今後の課題とする。

6 まとめ

本稿では、今後要求が増すと考えられる実数上の秘密計算について、軽量な乗算と除算の方式を提案した。提案方式では Aliasgari らの手法に比べ、少ない演算時間で実数の乗除算を行うことをしめした。本方式の特性上、利用する乱数の値域により演算結果の計算精度に影響ができることが考えられるが、利用対象のアプリケーションと入出力のデータ特性を十分に注意して値域を設定すれば精度が高い計算が実現可能であると考えられる。

参考文献

[1] I. Damgard, M. Fitzi, E. Kiltz, J. Nielsen, and T. Toft. Unconditionally secure constant-rounds multi-party computation for equality, comparison, bits and exponentiation. In Theory of Cryptography Conference (TCC), volume 3876 of LNCS, pages 285304, 2006.

[2] J. Garay, B. Shoenmakers, and J. Villegas. Practical and secure solutions for integer comparison. In Conference on Theory and Practice of Public Key Cryptography (PKC), pages 330342, 2007.

[3] T. Reistad. Multiparty comparison An improved multiparty protocol for comparison of secret-shared values. In International Conference on Security and Cryptography (SECRYPT), pages 325330, 2009.

[4] T. Reistad and T. Toft. Linear, constant-rounds bit-decomposition. In International Conference on Information, Security and Cryptology (ICISC), pages 245257, 2009.

[5] T. Toft. Constant-rounds, almost-linear bit-decomposition of secret shared values. In Topics in Cryptology CT-RSA, pages 357371, 2009.

プロトコル	ラウンド数	オペレーション回数
FLMul [10]	11	266
FLDiv [10]	17	444
提案手法による乗算 (n out of n)	2	4
提案手法による除算 (n out of n)	2	4
提案手法による乗算 (2 out of 3)	2	5
提案手法による除算 (2 out of 3)	2	5

表 1: Aliasgari らの手法との計算効率比較（単精度浮動小数点）

[6] Cybernetica. Cybernetica develops a solution for increasing space security. Cybernetica News blog, 2013, https://sharemind.cyber.ee/news-blog/cybernetica-develops-a-solution-for-increasing-space-security

[7] O. Catrina and A. Saxena. Secure computation with fixed-point numbers. In Financial Cryptography and Data Security (FC), pages 3550, 2010.

[8] M. Franz, B. Deiseroth, K. Hamacher, S. Jha, S. Katzenbeisser, and H. Schr ¨ oder. Secure computations on non-integer values. In IEEE International Workshop on Information Forensics and Security (WIFS), pages 16, 2010.

[9] M. Franz and S. Katzenbeisser. Processing encrypted floating point signals. In ACM Workshop on Multimedia and Security (MMSEC ' 11), pages 103108, 2011.

[10] M. Aliasgari, M. Blanton, Y. Zhang, and A. Steele. Secure Computation on Floating Point Numbers. In 20th Annual Network & Distributed System Security Symposium (NDSS 2013), 2013

[11] A. Shamir. How to share a secret. Comm. ACM, Vol. 22, No. 11, pp 612-613, 1979

[12] O. Catrina and S.D. Hoogh. Improved Primitives for Secure Multiparty Integer Computation. In Security and Cryptography for Networks (SCN 2010), 2010