

④ Garbled circuit を用いた秘密計算と混合的構成



菊池 亮 | NTT セキュアプラットフォーム研究所

Nuttapong Attrapadung | 産業技術総合研究所

Garbled circuit とは

データを隠しながらその分析が可能な秘密計算は、主に3つの作り方（構成方法）が知られており、それぞれ秘密分散、Garbled circuit^{☆1}、そして準同型暗号をパーツとして用いている。本稿では、その中でも Garbled circuit を用いた秘密計算に焦点をあて、その仕組みや特徴を紹介する。また、複数の構成方法を組み合わせた混合的構成について昨今の潮流を踏まえ紹介する。

まず、Garbled circuit とは何だろうか？ 実際の例は後述し、まずイメージだけ述べる。名前のとおり回路の一種であるが、通常の回路と異なり、入力のは0,1ではなく、0,1に対応した乱数 K_0, K_1 （これをラベルと呼ぶ）を用いて計算を行うような回路である。たとえば、通常の回路ではANDゲートに0と1を入力した場合1が出力されるが、Garbled circuit で同様の計算を行う場合、 K_0 と K_1 が Garbled circuit におけるANDゲート（これをGarbled ANDゲートと呼ぶ）に入力され、結果として K_0 が出力される。このような回路が Garbled circuit である。

Garbled circuit の構成

ではこのような Garbled circuit は実際にどう構成で

きるのか、例としてANDゲート1つだけのGarbled circuit の作り方を紹介する。具体的な例を図-1に挙げるので、適宜参照されたい。

ANDゲートには2つの入力ワイヤと1つの出力ワイヤが繋がっているものとし、2つの入力ワイヤをそれぞれ左／右入力ワイヤと呼ぶことにする。構成にはハッシュ関数 H を道具に使う。これは e ビットのラベル2つを入力とし、 e ビットの“乱数”を出力するものとする。

1. ラベルの生成：ANDゲートに繋がる左入力ワイヤを A 、右入力ワイヤを B 、出力ワイヤを C としたとき、すべてのワイヤに0と1に対応する e ビットのラベルを最下位ビットが異なるように生成する。たとえば A のラベルを (K_0^A, K_1^A) としたとき、 $e-1$ ビット列を2つ、1ビット b_A を1つランダムに選び、片方の $e-1$ ビット列に b_A を結合したものを K_0^A 、もう片方の $e-1$ ビット列に $(1-b_A)$ を結合したものを K_1^A とする。 B, C も同様である。
2. Garbled ANDゲートの生成：下記4つのゲート暗号文を生成する。ただし $\oplus$ はビットごとのXOR演算を意味する。
 - $H(K_0^A, K_0^B) \oplus K_0^C$
 - $H(K_0^A, K_1^B) \oplus K_0^C$
 - $H(K_1^A, K_0^B) \oplus K_0^C$
 - $H(K_1^A, K_1^B) \oplus K_1^C$
3. ゲート暗号文のシャッフル：LSB (K_0^A) = 1

^{☆1} 筆者の知る限りコンセンサスが得られた日本語訳がないため、本稿では Garbled circuit で統一する。

ならば4つの暗号文の上2つと下2つを入れ替える。さらに $\text{LSB}(K_0^B) = 1$ ならば、1つ目と2つ目、および3つ目と4つ目の暗号文を入れ替える。ここで LSB はビット列の最下位ビットを取り出す関数とする。

このように作られたラベルおよび Garbled AND ゲートが実際にラベルのみで AND ゲートの計算ができることを確認しよう。左入力ワイヤの値を $a \in \{0, 1\}$ 、右入力ワイヤの値を $b \in \{0, 1\}$ とし、 $a \text{ AND } b = c$ を計算する例を用いて説明する。入力 (a, b) に対応したラベル (K_a^A, K_b^B) およびゲート暗号文4つである。これらから K_c^C を計算するためには、4つのゲート暗号文のうち上から $2\text{LSB}(K_a^A) + \text{LSB}(K_b^B)$ 番目の暗号文を D としたとき、 $H(K_a^A, K_b^B) \oplus D$ を計算すればよい。ステップ2と3を確認すると、このとき $D =$

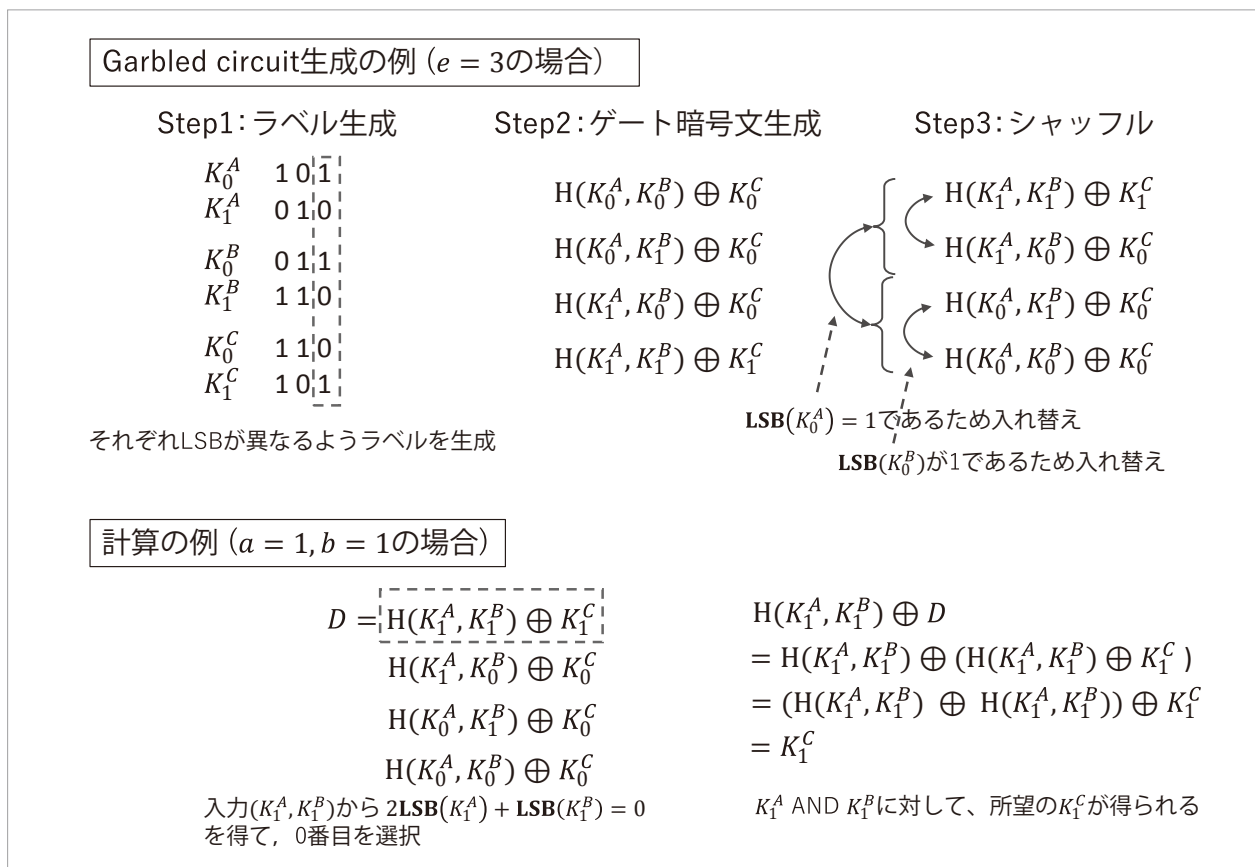
$K_c^C \oplus H(K_a^A, K_b^B)$ が成り立っているため、出力として所望の K_c^C を得られている。

4つの暗号文の K_0^C の並び方がそのままゲートの真理値表になっているので、並び順を入れ替えば任意のゲートを対象とできる。たとえば上から順に $K_0^C, K_1^C, K_1^C, K_0^C$ とすれば、Garbled XOR ゲートが得られる。回路がより深い場合は (K_0^C, K_1^C) を入力ワイヤのラベルと見て、順にゲートごとにゲート暗号文を作ればよい。

Garbled circuit を用いた秘密計算

次に上記の方法で Garbled circuit が作れたとして、それを用いてどのように秘密計算ができるかを見ていく。

まず、Garbled circuit を生成する人と、ラベルを



用いて Garbled circuit を計算する人を別のひととしよう。すると、Garbled circuit を生成した人であれば、自分が作ったものなので $0, 1$ と K_0, K_1 の対応がとれる、言い換えると、 K_0 を見たときに「これは 0 に対応するラベルだな」ということが分かる。一方で計算する人は、ラベルを見ても一体 0 と 1 のどちらに対応するラベルか分からない。この非対称性を使って秘密計算を作ることができる。

ここでは単純な例として、委託計算と呼ばれる「A さんのデータは隠しつつ、分析を B さんに委託する（計算してもらう）」というケースを考える。A さんはまず、委託したい分析を通常の回路で表現し、その回路の Garbled circuit を生成する。次に Garbled circuit を B さんに送るのだが、このとき、Garbled circuit の入力になるラベルには、A さんの持っているデータに対応したラベルを B さんへ送るものとする。たとえば A さんのデータが $0, 1, 1, 1$ であれば、 K_0, K_1, K_1, K_1 を B さんへ送るということである。すると Garbled circuit の性質から、B さんはラベルを使って、A さんの入力を知ることなく回路を計算し、計算結果のラベルを手に入れることができる。最後に B さんは計算結果のラベルを A さんに返せば、A さんはラベルから対応する値に戻すことができ、分析結果が得られる。

拡張として、もし A さんと B さんの両者がデータを持っており、それらを互いに隠しつつ分析したい場合はどうすればよいだろうか。もしも、A さんが B さんのデータを知ることなく、B さんの保持するデータに対応したラベルを B さんが得ることができれば、上記と同様に（A さんの入力を知ることなく）Garbled circuit を用いて計算することができる。実はそのようなことを実現する紛失通信と呼ばれる暗号プロトコルが知られているため、紛失通信を用いることで、両者がデータを持っている場合も秘密計算を行うことができる。

ほかの構成方法との比較

このような Garbled circuit を用いた秘密計算は、ほかの秘密分散を用いたもの、および準同型暗号を用いたものに比べて一般にどのような特徴があるだろうか。

秘密分散を用いた秘密計算に比べてとき、Garbled circuit を用いた秘密計算の 1 つの特徴は、通信回数（ラウンド数）が少ないことである。前述の A さんと B さんがデータを持っている場合を思い出すと、通信が必要なタイミングは

- A さんから Garbled circuit と（A さんのデータに対応した）ラベルを送るとき
- 紛失通信を使って B さんのデータに対応したラベルを送るとき
- B さんが A さんに Garbled circuit の計算結果を送るとき

の 3 つであり、紛失通信は数回の通信で実現できるため、全体としてラウンド数はやはり数回程度である。秘密分散を用いた秘密計算は一般的にラウンド数が大きくなるため、A さんと B さんの通信遅延が非常に大きいような場合、Garbled circuit を用いた秘密計算は優位性がある。また別の特徴として 2 者間計算が可能という特徴もある。秘密分散を用いた秘密計算では 3 者が必須である場合が多く、この点も優位性と考えることができる。

一方でデメリットとしては、入力データの 1 ビットにつきラベル e ビットの通信が必要となり、現在一般的に用いられる $e = 160$ だと元データのサイズから 160 倍程度に通信量が増えてしまい、さらにラベルに加えてゲート暗号文も送らなくてはならない。秘密分散はパラメータにもよるがたとえば通信量は 3 倍程度の増加で済むため、A さんと B さんの間の通信スピードが遅い場合にはあまり向いていない。また、基本的には $0, 1$ を入力とする論理回路を計算する技術のため、整数を入力とする算術回路（複数の 32 ビット整数の平均など）の計算はそれほど効

率的ではなく、さらに紛失通信はハッシュ関数の計算などに比べて重い計算を伴うため、AさんとBさんの計算機が非力な場合もあり向いていない。

準同型暗号を用いた秘密計算との比較は、準同型暗号に何を使うかや、その具体的なパラメータ設定に大きく依存するため、一般に比較することは難しい。特に完全準同型暗号と呼ばれる種類の暗号方式を用いた場合と比べると、通信ラウンドは完全準同型暗号を用いた秘密計算の方が小さくなるが、多くの場合計算コストはGarbled circuitの方が低くなる。また、Garbled circuitは論理回路を得意とするが、準同型暗号は算術回路を得意とするという違いがある。

なおこれらの比較はあくまで一般的な傾向の比較であり、より詳細には計算したい関数（分析方法）やデータ形式などにも依存するため、特定のアプリケーションにおける最も効率的な秘密計算の構成方法を見極めるには、具体的なパラメータでコストの見積りを行い比較する必要があることに注意されたい。

近年の発展

Garbled circuitは現在でもさまざまな面から研究が行われているが、より実用的な高速化に繋がる観点の研究結果としては、下記のようなものが挙げられる。通信量の削減のために、XORゲートであればゲート暗号文を送る必要がないfree-XORというテクニックや、XORゲート以外でもゲート暗号文を4つから半分の2つにできる手法が提案されている。Garbled circuitで効率的に算術回路を計算する方法は知られていなかったが、近年その方法についても進展があった¹⁾。また、紛失通信についてはその計算コストを減らすために、今まで大きい数（2048ビット数など）のべき乗剰余のような計算が必要だったものを、160ビット程度の数の演算に置き換える方法や、もう1人参加者を増やすことで計算を1ビットXORのような軽い演算のみにできる方法も知られている²⁾。

混合的構成による秘密計算

前述したとおり、秘密計算の構成方法には主に秘密分散に基づくもの、Garbled circuitに基づくもの、および準同型暗号に基づくものの3つがあり、それぞれの構成方法はいずれも異なる利点を持っている。そのため、これら3つのアプローチを用いた混合的構成が提案されている。本章では、そのような混合的構成についての研究動向を簡単に紹介する。

我々が知る限り、Garbled circuitによる構成と準同型暗号に基づく構成の融合による分岐プログラムの秘匿化手法³⁾が初めての混合的構成による秘密計算と考えられる。文献4)においては、Garbled circuitと準同型暗号に基づく混合的構成をモジュラー的に設計するための一般的フレームワークが提案されており、後にTASTYフレームワークにおける自動化コンパイラとして拡張がなされている。また、通常の準同型暗号と完全準同型暗号の組合せによる手法の提案もなされている。算術的秘密分散^{☆2}、論理的秘密分散、Garbled circuitの混合的構成が可能なABYフレームワークも提案されている（A,B,YはそれぞれArithmetic, Boolean, Yaoの頭文字である）。ABYフレームワークは元々2者秘匿計算への適用を想定して開発がなされていたが、最近において3者計算に拡張された手法の提案もなされている。混合的構成による秘密計算の開発を補助することを目的としたツールや言語の提案もなされており、これらを利用することで、秘密計算で計算する関数に応じた適切な構成方法の選択をより簡便に行うことができる。特に文献5)においては、計算処理に応じて性能を最適化する構成方法の自動的選択機能の提供がなされている。

上述のとおり、混合的構成を用いることでさまざまな計算処理に適した構成方法を適応的に選択する

☆2 秘密分散は算術回路も論理回路も計算可能であるため、ここでは区別のため、特に算術回路を計算する場合に算術的秘密分散、論理回路を計算する場合に論理的秘密分散と呼ぶ。

ことができ、データを秘密にしたまま計算したいさまざまなアプリケーションへの適用が期待される。そのようなアプリケーションの具体例として、遠隔医療診断システム、生体認証、リッジ回帰などが挙げられる。また、現時点において、最も注目されているアプリケーションの1つとして特にプライバシー保護機械学習が挙げられる。たとえば、ニューラルネットワークに基づく機械学習を実行する場合、線形関数と非線形関数が入り混じる多様な計算処理が必要となるため、異なる構成方法の混合的構成による秘密計算が必要となる。実際、最近においてきわめて多数のプライバシー保護機械学習の提案がなされており、これらのほとんどが ABY フレームワークを利用したものとなっている。また、やはり最近提案がなされた現在最も効率が良い手法と考えられている Gazelle は Garbled circuit と準同型暗号に基づく混合的構成になっている。その一方で、混合的構成に依存せず、単一のアプローチのみを用いた方式も依然として数多く提案がなされているが、現在プライバシー保護機械学習において Gazelle の性能を超えるものは知られておらず、今後も混合的構成を用いることで、機械学習および人工知能分野で数多くの実用的アプリケーションが得られるものと期待できる。

参考文献

- 1) Ball, M., Malkin, T. and Rosulek, M. : Garbling Gadgets for Boolean and Arithmetic Circuits, ACM CCS (2016).
- 2) Chida, K. and Takahashi, K. : Privacy Preserving Computations without Public Key Cryptographic Operation, IWSEC (2008).
- 3) Brickell, J., Porter, D. E., Shmatikov, V. and Witchel, E. : Privacy-Preserving Remote Diagnostics, ACM CCS (2007).
- 4) Kolesnikov, V., Sadeghi, A. and Schneider, T. : A Systematic Approach to Practically Efficient General Two-party Secure Function Evaluation Protocols and Their Modular Design, J. Comput. Secur., 21(2), pp.283-315 (2013).
- 5) Kerschbaum, F., Schneider, T. and Schropfer, A. : Automatic Protocol Selection in Secure Two-party Computations, ACNS (2014).

(2018年6月29日受付)

菊池 亮 (正会員) kikuchi_ryo@fw.ipsj.or.jp

2010年NTT入社, 2015年東工大博士課程後期修了。博士(工学)。秘密計算・匿名化などデータの2次利用に資する研究開発に従事。ISO/IEC JTC 1/SC 27/WG 2 エキスパート, (独)統計センター非常勤研究員。CSS2012/CSS2013/CSS2017 各論文賞, SCIS2017 イノベーション論文賞各受賞。

Nuttapong Attrapadung n.attrapadung@aist.go.jp

2007年東京大学大学院博士学位取得。博士(情報理工)。2008年産業技術総合研究所(AIST)入所。公開鍵暗号に関する研究に従事。SITA2005/SCIS2006 論文賞受賞。2010年エリクソンヤングサイエンティストアワード受賞。2017年度科学技術分野の文部科学大臣表彰・若手科学者賞受賞。