入札額の上限漏洩を防止した資金拘束型の封印入札オークション

陳 浩太[1,2,a)]　江村 恵太[2]　佐藤 慎悟[2]　面 和成[1,2]

概要：公開入札オークションでは，入札額を決定する際に，他の入札者の入札額をもとに予算を推定できてしまう．そのため入札額を秘匿可能な封印入札オークションが望ましい．しかしながら，既存の封印入札方式では入札額の拘束，すなわち入札後に入札額が支払われる保証がなく，虚偽の入札に対して脆弱である．スマートコントラクトを用いて入札額より大きいデポジットを要求することにより入札額の拘束が可能であるが，入札額の上限が他者に漏洩するという問題点がある．本論文では，入札額の上限漏洩を防止した資金拘束型の封印入札オークションを提案する．Ethereum 上で行われる通常の送金とワンタイムアドレスへの入札の送金とが区別できないことに着目し，入札額の上限が他者に漏洩することを防ぐ．さらに TLS 通信で取得したデータの正当性を第三者に証明可能なプロトコル DECO (Zhang ら，CCS 2020) を用いて，入札額相当の資金が存在することをスマートコントラクトに対して証明する．提案方式を実装するとともに，Ethereum のトランザクションのデータをもとに入札の送金がどの程度秘匿されるのかを検討する．

キーワード：ブロックチェーン，オークション，資金拘束，封印入札

A sealed bid auction with binding that can prevent leakage of budget information

Kota Chin[1,2,a)]　Keita Emura[2]　Shingo Sato[2]　Kazumasa Omote[1,2]

Abstract: In an open bidding auction, a bidder can know the budgets of other bidders. Thus, a sealed-bid auction that hides bidding prices is desirable. However, it is difficult to provide a binding property, which guarantees that a bidder has a balance more than or equal to the bidding price, in previous sealed-bid auction protocols. Thus, such protocols are vulnerable to false bidding. As a solution, many protocols employ a deposit method where each bidder sends a deposit greater than or equal to the bidding price on a smart contract before the bidding phase. However, it reveals the maximum bidding price and it would be better to hide this information. In this paper, we propose a sealed-bid action protocol providing the binding property that hides the bidding price itself and the maximum bidding price, simultaneously. For hiding the maximum bidding price, we pay attention to the fact that a transaction of the usual transfer of Ethereum and a transaction for sending a bidding price to one-time addresses are indistinguishable. We also employ DECO (Zhang et al,. CCS 2020) that proves the validity of data to a verifier where the data is taken from a source without showing the data itself. Finally, we give our implementation that shows gas costs required and discuss how much bidding transactions are hidden.

Keywords: Blockchain, Auction, Binding, Sealed-bid

[1] 筑波大学
University of Tsukuba
[2] 情報通信研究機構
National Institute of Information and Communications Technology
[a)] s2120540@s.tsukuba.ac.jp

1. はじめに

1.1 背景

近年，デジタルアートなどをブロックチェーン上で扱える形にした Non-Fungible Token (NFT) と呼ばれる資産に注目が集まっており，スマートコントラクトを用いた公開入札オークションが頻繁に行われている．例えば，「Doge」のインターネット・ミームで知られる柴犬「かぼすちゃん」の NFT は，スマートコントラクトを用いたオークションで高額な落札が行われた NFT の 1 つであり，落札額は 1696 ETH（当時約 4.7 億円）である[*1]．このような高価な資産の売買に用いられることからわかるように，スマートコントラクトを用いた公開入札オークションは一定の信頼性を獲得している．なお，安価な取引も行われており，2021 年 7 月に売買が行われた NFT の平均的な価格は約 711 ドル（8 月 20 日時点で約 0.24ETH）である[*2]．

しかし，スマートコントラクトを用いた公開入札オークションでは，他の入札者の入札額だけでなく，アドレスの残高をもとに予算を推定することができる．予算を推定できるということは，最終的な落札額を低く抑えることができる可能性があり，出品者の不利益に繋がると考えられる．そのため，入札額を秘匿可能な封印入札オークションを行うことが望ましい．

スマートコントラクトを用いた封印入札オークションに関して，数多くの研究がなされているが，入札額の拘束，すなわち入札後に入札額が支払われる保証がなく，虚偽の入札に対して脆弱である．例えば，Hisham らの研究 [1] や Li らの研究 [2] では，入札時に少額のデポジットをさせ，虚偽の入札だった場合にはデポジットを取り上げ経済的なペナルティを与える．しかしながら，ペナルティを許容するユーザによる虚偽の入札を防ぐことはできない．

また，入札時に入札額より大きいデポジットを要求することで入札額の正確な金額を隠した上で資金拘束を実現することができる．この方法は様々な方式 [3] [4] [5] [6] [7] [8] [9] [10] [11] [12] [11] で利用されており，本論文ではデポジット方式と呼ぶ．しかしながら，ブロックチェーン上ではデポジットの額が公開情報となるため，デポジット方式には入札額の上限となる情報が他の入札者に対して漏洩してしまうという問題点がある．

最後に，Ma ら [13] の方式では，入札前にデポジットのコミットメントを作成しており，一見資金拘束を行っているように見える．しかしながらコミットされた値に対応する金額を入札者が所持しているかどうかは保証されていない．そのため，正しく資金拘束が行われているとは言えない．

1.2 貢献

本研究には 3 つの貢献がある．1 つ目は，入札額の上限漏洩を防止した資金拘束型の封印入札オークションの提案である．つまり，入札時に入札額より大きいデポジットを求めることなく資金拘束を実現する．具体的には Ethereum のスマートコントラクトのアドレスがデプロイ前に計算可能であることを用いてワンタイムアドレスを入札者自身が発行する．ここで，ワンタイムアドレスへの送金は，Ethereum 上で行われる通常の送金と区別ができないため，入札額は他のユーザの送金によって秘匿される．その上で，TLS 通信で取得したデータの正当性を第三者に証明可能なプロトコル DECO [14] を用いて入札の証明を行う．また，提案方式で DECO を実行することを想定している Chainlink [15] のノードの動作は Chainlink のプロトコルによって担保されており，提案方式では信頼できる第三者を必要とせずに封印入札オークションを実現している．

貢献の 2 つ目は，実際の Ethereum のトランザクションのデータをもとに入札の送金がどの程度秘匿されるか解析を行ったことである．結果，入札の期間をある程度長くすることで十分秘匿可能であることを示した．例えば，8 月 9 日から 8 月 15 日までの 1 週間を入札の期間とすると，ワンタイムアドレスの可能性があるアドレスは合計 158644 件あり，その内のいずれかのアドレスの残高が入札額であり，それ以外の情報は漏洩しない．

貢献の 3 つ目は，提案方式を実装し，オークションの操作にかかる手数料について検討を行ったことである．また，入札額より大きいデポジットを要求することで資金拘束を実現するナイーブなデポジット方式の封印入札オークションも同様に実装し，提案方式による入札額上限の秘匿による手数料の増加が約 6 ドル（2021 年 8 月 20 日時点）で抑えられることを示した．

1.3 関連研究

我々と同様の動機によって, David らは独立に FAST (Fair Auctions via Secret Transactions) を提案している [16]. FAST では，デポジットを Confidential Transaction [17] を用いて秘匿するとともに，その開示のためのトラップドアを公開検証可能秘密分散 [18] で入札者の集合とは異なる deposit committee members に分散している．全 ℓ ラウンド (ℓ は bid のビット数) で Anonymous Veto Protocol [19] を実行し， あるユーザが不正を働いた場合にはそのデポジットを正直なユーザに分配することで，不正を働くインセンティブをなくしている．我々の方式との違いとして、まず計算量が挙げられる．FAST において，オークション参加者は各ラウンドで自身以外の参加者とのやり取りが

[*1] 落札トランザクション https://etherscan.io/tx/0x8668bb338f7cf9896db75c00e8bef18cc549d04b2dcaf1cee01dc0e1522e7e87

[*2] NFT マーケットプレイス OpenSea の 2021 年 7 月の取引データより算出（オークション以外の取引形態も含む）https://dune.xyz/rchen8/opensea

発生するため，全体として $O(\ell n)$ の計算が必要となる (n は入札者数)．一方，我々の方式は他オークション参加者との通信は必要としない (DECO の実行時においてオラクルと通信する以外はスマートコントラクトのみと対話する)． また FAST ではトラップドアを deposit committee members で分散管理するため，結託人数に上限がある (m 人に対し，$m/2-2$ 人まで)．一方，我々の方式ではそのような制限がない．

つまり，FAST は暗号学的なアプローチによって秘匿化された資金拘束を実現したため，計算量が増加している．一方，我々はスマートコントラクトのデプロイ前に送金できることに着目し，通常のトランザクションとワンタイムアドレスへの送金トランザクションとが同じ構造であり区別できないことを利用することで，秘匿化された資金拘束の実現にかかる計算量やコストを大幅に削減した．

2. 準備

2.1 Ethereum とスマートコントラクト

Ethereum [20] とは，ブロックチェーン技術を基にした分散型アプリケーションのプラットフォームである．ユーザは，スマートコントラクトと呼ばれる Ethereum Virtual Machine（EVM）上で動作するアプリケーションを作成することができ，トランザクションを通してスマートコントラクトを実行することができる．スマートコントラクトの実行結果は全てのノード間で共有・合意されるため，改ざんすることが困難である．

また，トランザクションの手数料として支払う Ethereum のネイティブトークン ETH の量は消費するガスの量に gasPrice と呼ばれる Ethereum ネットワークの需要に応じて変動する値を乗算することで求められる．

2.2 オラクル

スマートコントラクトはブロックチェーン外のデータ，例えば ETH の価格や株価，天気，選挙結果などを直接取得することができない．そのため，これらのデータをスマートコントラクトで利用したい場合，外部からスマートコントラクトに対して入力する必要がある．このとき，外部からスマートコントラクトに対してブロックチェーン外のデータを入力するエンティティのことをオラクルと呼ぶ．

現在，Ethereum で利用されているオラクルのうち最もシェアが大きいプロトコルが Chainlink [15] である．Chainlink は分散型オラクルネットワークであり，複数のオラクルノードが取得したブロックチェーン外のデータを集約することで，オラクル自身の不正による影響を防いでいる．また，Chainlink のノードは Chainlink のプロトコルを逸脱した行動を行うと経済的なペナルティを受けるため，プロトコルに従った動作に関しては信頼できると仮定されている．

2.3 DECO

本章では，TLS 通信で取得したデータを特定の第三者に証明可能なプロトコル DECO [14] を紹介する．一般的に，スマートコントラクトに入力される値 (例えば ETH の価格等) の信頼性が問題となる．DECO は情報ソース $\mathcal{S}$，証明者 $\mathcal{P}$，検証者 $\mathcal{V}$ の三者におけるプロトコルであり，$\mathcal{P}$ がある情報を $\mathcal{S}$ より取得したことを， その情報そのものは秘匿したまま $\mathcal{V}$ に証明することができる．また $\mathcal{P}$ が $\mathcal{S}$ にアクセスするためにはあるパスワード等の秘密の情報が必要である状況を考える．DECO の処理の流れを以下に示す．$\mathcal{P}$ と $\mathcal{S}$ 間は通常の TLS 通信を仮定する．3 者ハンドシェイクプロトコルにより， $\mathcal{S}$ が知る MAC 鍵が $\mathcal{P}$ と $\mathcal{V}$ 間で秘密分散の形でシェアされる．暗号化用の鍵は $\mathcal{P}$ と $\mathcal{S}$ 間で通常通り共有される．$\mathcal{P}$ は $\mathcal{S}$ に対する TLS 通信を介したクエリを $\mathcal{V}$ の協力の元で行うことができる．具体的には， それぞれが分散された MAC 鍵を入力とした 2 者間マルチパーティ計算を行う．ここで $\mathcal{V}$ は暗号化鍵を知らないため， クエリ内容と返答はこの時点では $\mathcal{P}$ と $\mathcal{S}$ のみが知ることに注意されたい．$\mathcal{P}$ はクエリと $\mathcal{S}$ からの返答に対するコミットメントを作成し $\mathcal{V}$ に送付，$\mathcal{V}$ は秘密分散された MAC 鍵を $\mathcal{P}$ に送付する．$\mathcal{P}$ は MAC 鍵を復元し $\mathcal{S}$ からの返答の正当性を確認した後，クエリ結果の正当性をゼロ知識証明を用いて $\mathcal{V}$ に証明する．

ここで，2.2 節で述べた Chainlink と DECO は統合が予定されており，統合された場合，ユーザは自身の個人情報を含むデータの正当性を Chainlink のノードに対して証明することができる [15]．本論文の提案方式では，DECO を用いて Chainlink のノードに対してデータの正当性を証明すると仮定する．

2.4 ハッシュ関数ベースコミットメント

本章ではコミットメント方式を紹介する．コミットメント方式 $(\mathsf{Commit}, \mathsf{ComOpen})$ は以下で定義される．Commit アルゴリズムはメッセージ M を入力とし， コミットメント com とデコミットメント dec を出力する．ComOpen アルゴリズムは com，dec，M を入力とし，0 (拒否) または 1 (受理) を出力する．安全性として，com からは M の情報が漏れない Hiding，com に対し $\mathsf{ComOpen}(\mathsf{com}, \mathsf{dec}, M) = 1$, $\mathsf{ComOpen}(\mathsf{com}, \mathsf{dec}', M') = 1$，かつ $M \neq M'$ をみたすデコミットメント dec，dec' を計算できない Binding がある．

次に提案方式におけるコミットメント方式の選定について述べる．提案方式ではスマートコントラクト内で ComOpen アルゴリズムを実行する．ガスコストを低く抑える観点から，効率的な方式を選定する必要がある．そのためハッシュ関数のみを使用した方式を採用する．以下 $\mathsf{Hash} : \{0,1\}^* \to \{0,1\}^k$ をハッシュ関数とする (k はセキュリティパラメータ)．Commit アルゴリズムでは乱数 $R \xleftarrow{\$} \{0,1\}^k$ を選び，$\mathsf{com} = \mathsf{Hash}(M||R)$ を計算，com

と $\mathsf{dec} = R$ を出力する． ComOpen アルゴリズムでは $\mathsf{com} = \mathsf{Hash}(M||\mathsf{dec})$ が成り立つときに 1 を，そうでないとき 0 を出力する．本方式はハッシュ関数がランダムオラクルとしてモデル化される場合に安全であることが示されている [21], [22]．そこで本ハッシュ関数ベースコミットメント方式を採用し，ハッシュ関数としてはランダムオラクルとして SHA256 を選択する．なぜなら，ハッシュ関数の中でも SHA256 はスマートコントラクトで実行する際にプリコンパイルされた関数として計算にかかるガスコストが非常に小さいためである．

2.5 ゼロ知識証明

オフチェーンで実行する DECO プロトコル (2.3 章参照) では, コミットメントされた値に対するゼロ知識証明を行う必要がある．本章では, ハッシュ関数ベースのコミットメント方式に対するゼロ知識証明を紹介する．まず SHA256 の計算に対する算術回路 $\mathcal{C}$ を生成する．これは SHA256 の計算を C 言語で書き下し，回路計算機 [23] を通すことで得られる．次に算術回路に対する zk-SNARK (zero-knowledge Succinct Non-interactive ARgument of Knowledge) [24] を用いることで, 所望の方式が得られる．なお論文 [25] では SHA256 の compression function に対する算術回路を自前で生成することで， 上に挙げた一般的な方法よりも回路サイズを小さくすることに成功している．このような効率化の可能性は残るものの， 我々の興味は主にスマートコントラクト上での計算効率化によるガスコストの削減であり，オフチェーンで行うゼロ知識証明部分に関してはその実現可能性を示すに留めることとする．

3. 提案方式

本章では，本研究で提案する方式について紹介する．また，提案方式のシーケンス図を図 1 に示す．

提案方式では, 入札時に入札額より大きいデポジットを求めることなく資金拘束を実現する．具体的には Ethereum のスマートコントラクトのアドレスがデプロイ前に計算可能であることを用いてワンタイムアドレスを入札者自身が発行する．ここで，ワンタイムアドレスへの送金は，Ethereum 上で行われる通常の送金と区別ができないため，入札額は他のユーザの送金によって秘匿される．その上で，入札額のコミットメントをスマートコントラクトに入力し，DECO [14] を用いて入札時点におけるワンタイムアドレスの残高と入札額とが同じであることを証明する．

また，本論文では入札フェーズ期間中の入札額の秘匿と資金拘束との両立を目的としている．そのため, 本論文では入札後の落札者決定時における情報の秘匿は対象外とする．なおオークションコントラクトに入札額のコミットメントが保存されているため，ゼロ知識証明やマルチパーティ計算等を用いた既存封印入札方式を併用することで，入札額を秘匿したまま落札者を決定することは可能と考えられる．

3.1 オークションの開始

オークションを開始したいユーザは，オークションを管理するスマートコントラクト（以降，オークションコントラクト）に対して入札フェーズ・公開フェーズの期間などの情報を送信する．この際，スマートコントラクトで管理されるオークションごとにユニークな番号（以降，オークション ID と呼称）が割り振られる．

また，オークションの出品物として NFT を想定する場合，オークションの開始と同時に出品者は NFT をオークションコントラクトに対して送付する．

3.2 入札フェーズ

オークションに入札したいユーザ（以降，入札者と呼称）は，入札フェーズの間に次の手順を実行する．

3.2.1 ワンタイムアドレスの発行

ワンタイムアドレスとは，何らかの用途のために発行される使い捨てのアドレスのことである．提案方式では，入札額の送金先としてワンタイムアドレスを利用する．

ワンタイムアドレスの発行方法として，信頼できる第三者がいる場合，第三者がワンタイムアドレスを発行すればよい．しかし，前提として信頼できる第三者は存在しないため，この方法を取ることができない．

そこで Ethereum において，スマートコントラクトがデプロイされるアドレスを事前に計算可能であることを利用して，ワンタイムアドレスの発行を行う．具体的にはデプロイ時に CREATE2 と呼ばれるオペコードを用いることで，ソルトと呼ばれる任意の値とデプロイされるスマートコントラクトのバイトコードなどからアドレスが計算可能である[*3]．

本論文で提案する手法では，ソルトとしてオークション ID と入札者のアドレス，入札者自身が選んだ任意の値（以降，パスワードと呼称）を連結したものを用いる．つまり，パスワードを明らかにするまでワンタイムアドレスを知るのは入札者だけである．また，ワンタイムアドレスに送金された資金を操作可能なのは，後にデプロイされるスマートコントラクトのみであることに注意する．

3.2.2 入札

入札者は，生成されたワンタイムアドレスに対して入札額の送金を行う．前述のようにワンタイムアドレスはスマートコントラクトが以降の操作でデプロイされるアドレスであり，自身で資金を引き出すことができないため，この操作が資金の拘束につながっている．その際，オークションコントラクトに対して入札情報を送信するアドレス

[*3] 事前にアドレスを計算する手法[*4]は Ethereum 上の Argent[*5]などのプロトコルで用いられている．

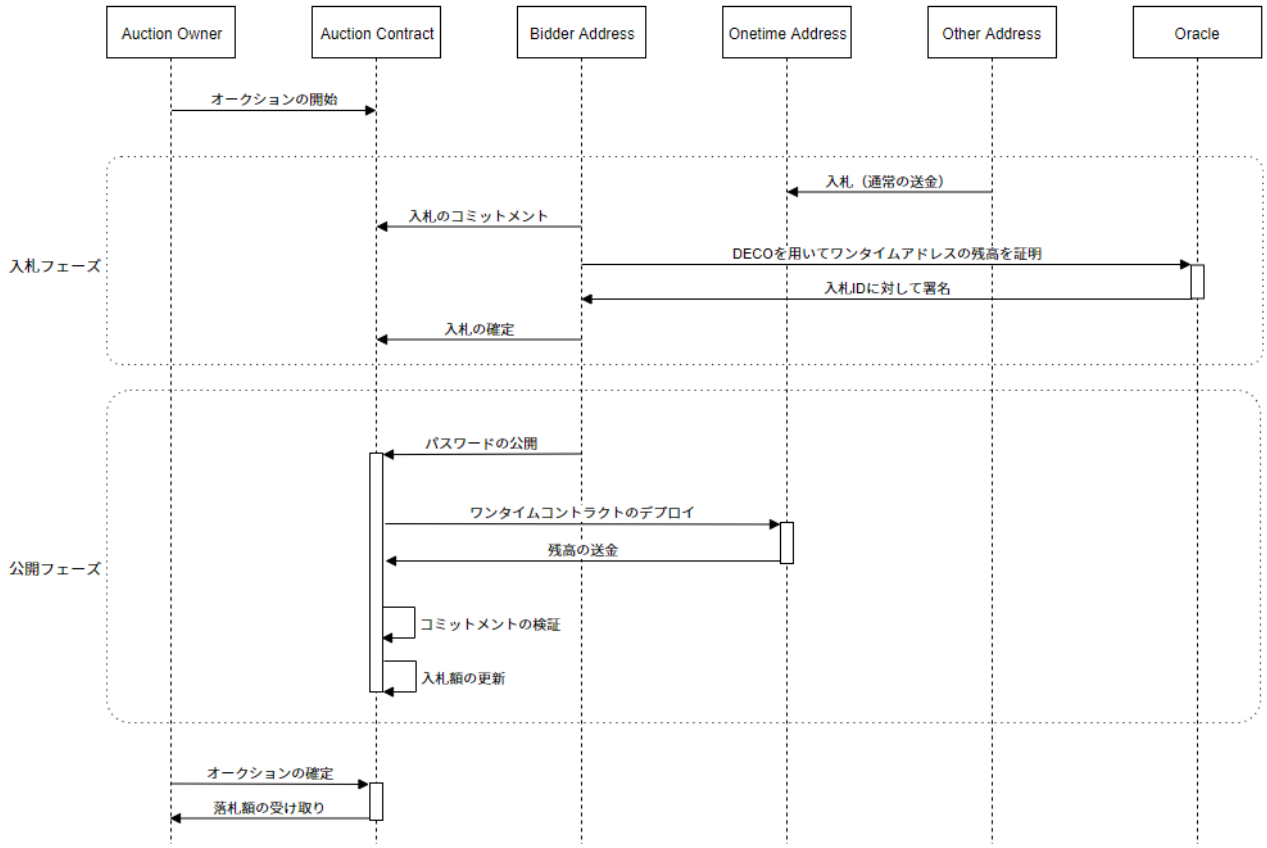

図 1　提案方式のシーケンス図

とは別のアドレスから送金を行う必要がある．

3.2.3　DECO を用いた入札の証明

入札者はワンタイムアドレスに送金後，入札額の送金を証明する必要がある．なぜなら，入札額の送金を証明せずに入札を受け付けた場合，複数の残高で入札しておくことで，後述する公開フェーズの間に任意のワンタイムアドレスに送金し，入札を公開することが可能だからである．

そこで，提案方式では，DECO を用いて Etherscan[*6]にて表示されるワンタイムアドレスの残高が入札額分と等しいことをオラクルノードに対して入札者が証明する．[*7] Etherscan を $\mathcal{S}$，入札者を $\mathcal{P}$，オラクルを $\mathcal{V}$ とし，$(\mathsf{vk}_{\mathcal{O}}, \mathsf{sigk}_{\mathcal{O}})$ をオラクルの署名検証鍵，署名鍵とする．入札者が Etherscan にアクセスする際の秘密情報はワンタイムアドレス θ_p とする．

まず，入札者は P を入札額とし，$(\mathsf{com}_P, \mathsf{dec}_P) = \mathsf{Commit}(P)$ を計算，(com_P) をオークションコントラクトに送付する．オークションコントラクトはコミットメントに対応するユニークな番号（以降，入札 ID と呼称）を発行する．

次に Etherscan，入札者，オラクルで DECO を実行し，入札者は Etherscan よりワンタイムアドレスの残高 Balance 値が P^* であることを取得する．なおここで Etherscan のアドレスが含まれるクエリ内容については $\mathcal{V}$ に隠されることから，オラクルは Etherscan 内のどのページが入札者に対応するのかはわからないことに注意されたい．つまり，オラクルは入札者のワンタイムアドレスを学習することができない．

そして，入札者はオラクルに対して P^* が θ_p より取得したことのゼロ知識証明に加え，$\mathsf{zk\text{-}PoK}\{P^* : \mathsf{com}_P = \mathsf{Commit}(P^*)\}$ を証明する． これは DECO 実行前にオークションコントラクトに保存されているコミットメントの中身が P^* であることを示している．オラクルは検証が正しいことの証明として，$\mathsf{sigk}_{\mathcal{O}}$ を用いて入札 ID に対する署名を作成する．

最後に，入札者はオラクルによる署名をオークションコントラクトに送信し，スマートコントラクトは署名が正しければ入札 ID に対応する入札を確定させる．

3.3　公開フェーズ

入札者は公開フェーズにおいて，次の手順に従って入札額の公開を行う．

3.3.1　パスワードの公開

入札者は，オークションコントラクトに対してソルトとなるオークション ID，入札額，パスワードを送信する．オークションコントラクトは，ソルトを元に入札者のワン

*6　Ethereum のアドレスの残高など各種情報を確認できるサービス
*7　論文 [14] における DECO を用いたバイナリーオプションへの応用を修正した．

タイムアドレスに対して資金回収用のスマートコントラクト（以降，ワンタイムコントラクトと呼称）をデプロイする．ワンタイムコントラクトは自身のアドレスの残高（入札者の入札額）をオークションコントラクトに対して送金する．

その後，オークションコントラクトは，2.4 節で示した ComOpen アルゴリズムを実行しコミットメントの検証を行う．検証後，DECO によって確認された有効な入札として存在するかを確認し，有効な場合，入札額の更新を行う．

この時，ワンタイムコントラクトから回収した残高がコミットした入札額より多い場合でもコミットした入札額を優先する．これも DECO を用いる理由と同様に，残高を入札額とすると公開フェーズの間に送金を行うことで入札額を上げることが可能だからである．

また，ワンタイムアドレスに送金された資金はオークションコントラクトを経由してワンタイムコントラクトをデプロイするか，2 の 256 乗個の秘密鍵から探し当てるかでしか回収できない．つまり，パスワードを公開しないユーザはペナルティとして入札額の全額がロックされる．

3.4 オークションの確定

出品者は，オークションコントラクトに対してトランザクションを送信することでオークションを確定させる．つまり，公開フェーズで公開された入札に基づいて落札者を決定する．

また，オークションの出品物として NFT を想定する場合，オークションの確定と同時に NFT が落札者に対してオークションコントラクトから送付される．

4. 実装評価

4.1 ワンタイムアドレスの安全性

本章では，ワンタイムアドレスの安全性について検討を行う．ワンタイムアドレスは入札者のみが知るパスワードによって生成されるため，他者はワンタイムアドレスを推測することが難しい．また，ワンタイムアドレスへの送金は，送金時点においてコントラクトのデプロイ前であるため，Ethereum において行われる通常の送金と区別がつかない．

ここで，悪意ある入札者が他者の入札を探し当てる方法を考えると，ワンタイムアドレスが持つ次のような性質からワンタイムアドレスそのものを推測することができる．

(1) 入札フェーズの間に他のアドレスに対する送金がないアドレス
(2) 入札フェーズの間に初めて ETH を受け取ったアドレス

1 つ目の条件は，ワンタイムアドレスの残高を操作可能なのは，公開フェーズにてデプロイされるワンタイムコントラクトだけであることから明らかである．また，2 つ目の条件はワンタイムアドレスが入札フェーズ期間より以前に利用されていないことから明らかである．したがって，これらの条件を満たすアドレスを抽出することで，ワンタイムアドレスが推測可能であり，候補となるアドレスが少ない場合には入札額を推定できる可能性がある．

表 1 ワンタイムアドレスの可能性があるアドレスの数

	3 日	1 週間	2 週間
0∼0.1ETH	45575	99193	177796
0.1∼0.5ETH	15837	34939	63656
0.5∼1ETH	4148	9481	17766
1∼10ETH	5670	12597	23052
10∼50ETH	772	1751	3307
50∼100ETH	125	262	464
100∼1000ETH	186	352	608
1000∼ETH	39	69	124

そこで，Ethereum のトランザクション（送金履歴）の全データが利用できる GCP の BigQuery を用いて，これらの条件を満たすアドレスの抽出を行った．送金額別のワンタイムアドレスの可能性があるアドレスの数を表 1 に示す．また，提案方式の入札フェーズの期間として 8 月 13 日から 8 月 15 日までの 3 日間，8 月 9 日から 8 月 15 日までの 1 週間，8 月 2 日から 8 月 15 日までの 2 週間を利用している．

ここでオークションの出品物として想定している NFT の平均的な価格は，1 章で述べたように約 711 ドル（8 月 20 日時点で約 0.24ETH）である．そこで，オークションにおける入札額を 0.1∼0.5ETH と想定すると，表 1 からワンタイムアドレスの可能性があるアドレスは 3 日で 15837 件，1 週間で 34939 件，2 週間で 63656 件あることが分かる．

また，8 月 2 日から 8 月 15 日までの 2 週間のデータを用いて，0.1ETH から 10ETH まで 1.25 倍刻みでワンタイムアドレスの可能性があるアドレスの数を表したのが図 2 である．これらのアドレスの全てが提案方式における入札に見えるため，期間を長くすればするほど入札を秘匿することが可能である．

一方，提案方式の制限として，ワンタイムアドレスの可能性があるアドレスの残高のうち，最も多い残高が全ての入札額の上限となる．しかしながら，Ethereum 上では日々多額の送金が行われており，8 月 2 日から 8 月 15 日までの 2 週間を入札フェーズとして想定した場合，上限となるアドレスの残高は 146223ETH であった．つまり，入札額の上限からオークションの入札額を推定することは困難である．また，デポジット方式の封印入札オークションにおいて，同様に上限を高くすることで推定を防ぐ場合，入札者の資金の多寡による制限がある．

4.2 提案方式のオークションの操作にかかる手数料

本章ではオークションの操作にかかる手数料について考

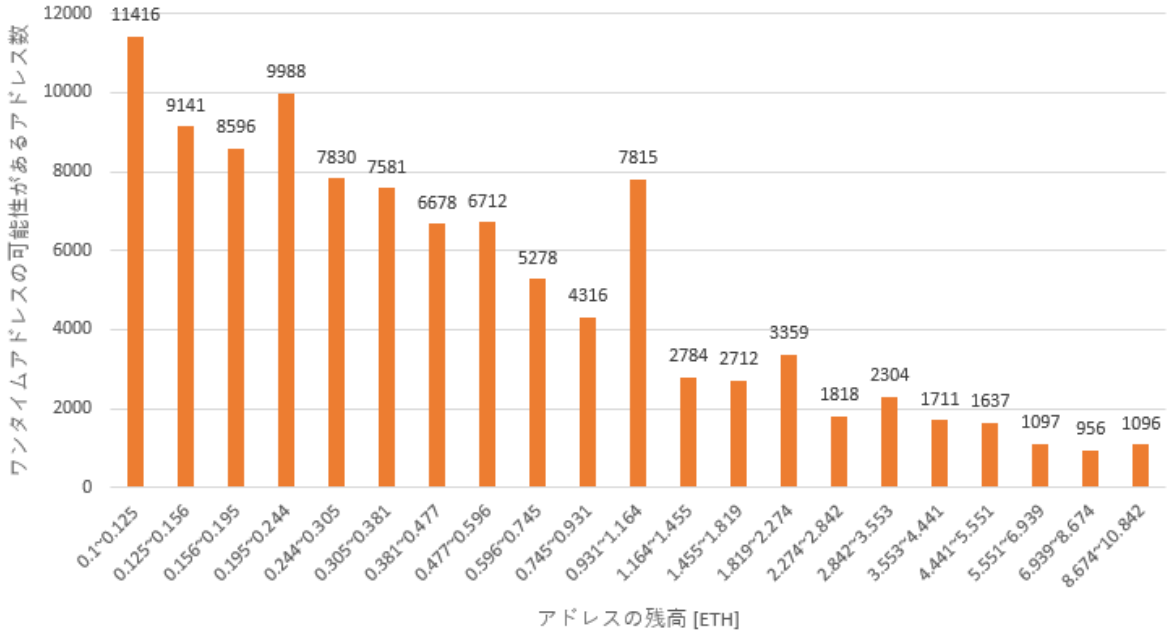

図 2　ワンタイムアドレスの可能性があるアドレスの数

表 2　入札者の操作にかかる手数料（提案方式）

操作	消費する gas	手数料
入札額の送金	21000	1.89 USD
入札のコミットメント	68903	6.20127 USD
入札の証明	52755	4.74795 USD
入札の公開	122546	11.02914 USD

表 3　出品者の操作にかかる手数料（提案方式）

操作	消費する gas	Fee
オークションの登録	166510	14.9859 USD
オークションの終了	40312	3.62808 USD

察する．2.1 章で述べたように，スマートコントラクトの操作を行うにはガスと呼ばれる手数料を支払う必要がある．

提案方式のオークションにおいて入札者が行う処理にかかるガスと USD 建ての手数料を表 2 に示す．本論文では，gasPrice を 30gwei，1ETH の価格を 3000 ドルと仮定する (2021 年 8 月時点)．

通常，入札者は表 2 に示す操作を 1 度ずつ行うため，入札者は一連の操作で約 23.87 ドル分の ETH を消費する．消費するガスは，入力するパスワード長などによって変動するが，gasPrice 及び ETH 価格の変動が激しいため誤差程度であると考えられる．また，一連の操作にかかる手数料は入札者の人数に依存しない．

次に，出品者の操作にかかるガスと手数料を表 3 に示す．消費するガスは，オークションの開始時にスマートコントラクトに対して入力する情報の長さによって変動するものの，概ね 18 ドル程度である．また，NFT のオークションを想定する場合，表 3 に示したガスに加えて，NFT の送付にかかるガスが必要となる．

4.3　操作にかかる手数料の比較

まず，入札額より大きいデポジットを要求することで入札額を秘匿するナイーブなデポジット方式を考える．1 章で述べたように，この方式では入札額の上限が漏れるという問題点が存在する．そこで，提案方式とデポジット方式を比較することで，入札額の上限の秘匿にかかる手数料の増加を検討することができる．また，手数料はスマートコントラクトにおいて保存するデータ量や計算量に依存するため，提案方式で実装したスマートコントラクトの構造を可能な限り変更せずにデポジット方式のスマートコントラクトを実装した．

表 4 に入札者が行う一連の操作にかかるガスと手数料を示す．4.2 節で述べたように提案方式にかかる手数料は約 23.9 ドルであり，提案方式による入札額の上限の秘匿にかかる手数料の増加は約 6 ドルに抑えられている．また，本論文の条件の下では Ethereum 上の通常の送金に 1.89USD かかることに注意されたい．

次に，公開入札オークションの操作にかかる手数料と提案方式にかかる手数料を比較する．この方式でも，提案方式で実装したスマートコントラクトの構造を可能な限り変更せず実装した．入札者が行う一連の操作にかかるガスと手数料を表 5 に示す．入札にかかる手数料は約 6.4 ドルと封印入札オークションに比べて低いものの，一般に公開入札オークションでは複数回の入札が行われることを考慮すると，結果的に一連の操作が 1 度で済む封印入札オークションの方が手数料が低くなる可能性も考えられる．

表 4 入札者の操作にかかる手数料（デポジット方式）

操作	消費する gas	Fee
入札のコミットメント	110928	9.98352 USD
入札の公開	83119	7.48071 USD

表 5 入札者の操作にかかる手数料（公開入札方式）

操作	消費する gas	Fee
入札	71137	6.40233 USD

5. 結論

本論文では，ワンタイムアドレスへの送金と TLS 通信で取得したデータの正当性を第三者に証明可能なプロトコル DECO を組合せることで，入札額の上限漏洩を防止した資金拘束型の封印入札オークションを提案した．また，DECO が実行されることを想定している Chainlink のノードの動作は Chainlink のプロトコルによって担保されており，信頼できる第三者を必要とせずに封印入札オークションを実現している．

更には，実利用を想定し，ガスコスト及び入札がどの程度秘匿されるか実装及び解析を行った．その結果，ナイーブなデポジット方式と比べても手数料の増加が本論文の条件の下で約 6 ドルに抑えられたこと，入札の秘匿化は入札フェーズを 1 週間から 2 週間とある程度長くすることで十分秘匿可能であることを示した．

最後に，提案方式の入札額の秘匿方法において，ワンタイムアドレスの可能性があるトランザクションを統計的に分析することが入札額の推定に繋がるかに関しては今後の課題としたい．例えば，提案方式の入札フェーズの前後の期間における送金の増加を調べることで，主な入札帯の推定が可能であるかなどである．

謝辞 本研究は JSPS 科研費 JP19H04107, JP21K11897 の助成を受けたものです．

参考文献

[1] Hisham S. Galal and A. Youssef. Trustee: Full privacy preserving vickrey auction on top of ethereum. In *3rd Workshop on Trusted Smart Contracts*, pp. 190–207, 2019.

[2] Honglei Li and Weilian Xue. A blockchain-based sealed-bid e-auction scheme with smart contract and zero-knowledge proof. pp. 5523394:1–5523394:10, 2021.

[3] Ahmed E. Kosba, Andrew Miller, Elaine Shi, Zikai Wen, and Charalampos Papamanthou. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *IEEE Symposium on Security and Privacy*, pp. 839–858, 2016.

[4] Michal Król, Alberto Sonnino, Argyrios G. Tasiopoulos, Ioannis Psaras, and Etienne Rivière. PASTRAMI: privacy-preserving, auditable, scalable & trustworthy auctions for multiple items. In *ACM Middleware*, pp. 296–310, 2020.

[5] Maha Kadadha, Rabeb Mizouni, Shakti Singh, Hadi Otrok, and Anis Ouali. *ABCrowd*: An *A*uction mechanism on *B*lockchain for spatial *Crowd*sourcing. *IEEE Access*, Vol. 8, pp. 12745–12757, 2020.

[6] Gaurav Sharma, Denis Verstraeten, Vishal Saraswat, Jean-Michel Dricot, and Olivier Markowitch. Anonymous fair auction on blockchain. In *IFIP NTMS*, pp. 1–5. IEEE, 2021.

[7] Iman Vakilinia, Shahriar Badsha, and Shamik Sengupta. Crowdfunding the insurance of a cyber-product using blockchain. In *IEEE UEMCON*, pp. 964–970, 2018.

[8] Alberto Sonnino, Michal Król, Argyrios G. Tasiopoulos, and Ioannis Psaras. Asterisk: Auction-based shared economy resolution system for blockchain. *CoRR*, Vol. abs/1901.07824, , 2019.

[9] Bader Al-Sada, Noureddine Lasla, and Mohamed M. Abdallah. Secure scalable blockchain for sealed-bid auction in energy trading. In *IEEE ICBC*, pp. 1–3.

[10] Po-Chu Hsu and Atsuko Miyaji. Verifiable M+lst-price auction without manager. In *IEEE DSC*, pp. 1–8, 2021.

[11] 杉谷勇気, 宮地充子. スマートコントラクトを用いた安全なセカンドプライスオークションの提案. 信学技報, No. 437, pp. 289–294, 2020.

[12] 杉谷勇気, 宮地充子. Financial fairness を実現するセカンドプライスオークションの提案. 信学技報, No. 257, pp. 63–70, 2019.

[13] Jie Ma, Bin Qi, and Kewei Lv. Fully private auctions for the highest bid. In *ACM TUR-C*, pp. 64:1–64:6, 2019.

[14] Fan Zhang, Deepak Maram, Harjasleen Malvai, Steven Goldfeder, and Ari Juels. DECO: liberating web data using decentralized oracles for TLS. In *ACM CCS*, pp. 1919–1938, 2020.

[15] Benedict Chan Alex Coventry Steve Ellis Ari Juels Farinaz Koushanfar Andrew Miller Brendan Magauran Daniel Moroz Sergey Nazarov Alexandru Topliceanu Florian Tramèr Breidenbach Lorenz, Christian Cachin and Fan Zhang. Chainlink 2.0: Next steps in the evolution ofdecentralized oracle networks. 2021. https://research.chain.link/whitepaper-v2.pdf.

[16] B. David, Lorenzo Gentile, and M. Pourpouneh. Fast: Fair auctions via secret transactions. p. 264, 2021.

[17] Greg Maxwell. Confidential transactions.

[18] Ignacio Cascudo and Bernardo David. ALBATROSS: publicly attestable batched randomness based on secret sharing. In *ASIACRYPT*, pp. 311–341, 2020.

[19] Feng Hao and Piotr Zielinski. A 2-round anonymous veto protocol. In *Security Protocols*, pp. 202–211, 2006.

[20] Vitalik Buterin. A next-generation smart contract and decentralized application platform. 2015.

[21] Rafael Pass. Alternative variants of zero-knowledge proofs. Technical report, 2004.

[22] Dominique Unruh. Computationally binding quantum commitments. In *EUROCRYPT*, pp. 497–527, 2016.

[23] Bryan Parno, Jon Howell, Craig Gentry, and Mariana Raykova. Pinocchio: Nearly practical verifiable computation. In *IEEE Symposium on Security and Privacy*, pp. 238–252, 2013.

[24] Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza. Succinct non-interactive zero knowledge for a von neumann architecture. In *USENIX Security Symposium*, pp. 781–796, 2014.

[25] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *IEEE Symposium on Security and Privacy*, pp. 459–474, 2014.