台帳を用いない決済方式に関する技術面からの一考察

田村裕子（たむらゆうこ）／阿部正幸（あべまさゆき）／奥田哲矢（おくだてつや）／津川天祐（つがわひろまさ）／宮澤俊之（みやざわとしゆき）／山村和輝（やまむらかずき）／赤羽喜治（あかはねよしはる）／田口智貴（たぐちともき）／平栗勇人（ひらくりゆうと）／増田博人（ますだひろと）／山田健斗（やまだけんと）

要　旨

本稿は、サービス事業者を介さず、台帳を用いることなく決済を可能とする決済方式について、その技術的側面から考察を行うものである。これは、現金に見立てた電子データの送受信によって決済を行う手段であり、暗号技術に関する研究分野では、「電子現金」という名称で検討が進められてきた。1990 年代にはいくつかの実証実験が行われたが、当時の技術水準ではユーザビリティを確保することが難しかったものと推測される。そこで、本稿では、この間の技術進展および社会ニーズを踏まえて電子現金方式の再整理を行うとともに、スマートフォンを用いた実機検証を通して、現在の技術水準であればユーザビリティの高い電子現金を提供できる可能性があることを示す。また、技術的側面から、電子現金のさらなるユーザビリティ向上とプライバシ強化に向けて、電子現金を任意の金額に分割・集約可能な方式、および、同一ユーザが使用した電子現金の相互の関連付けを困難とする方式を提案する。なお、本稿はあくまで電子現金にかかる技術面からの考察を行うものであり、法律や制度、実運用等、社会実装に向けた実現可能性は検討の対象外であることには留意されたい。

キーワード：　スマートフォン決済、デジタル決済、電子現金、電子マネー、プライバシ保護

本稿の作成に当たっては、藤岡淳教授（神奈川大学）と國廣昇教授（筑波大学）から有益なコメントをいただいた。ただし、本稿に示されている意見は、筆者たち個人に属し、日本銀行、NTT 株式会社、および、株式会社 NTT データの公式見解を示すものではない。また、ありうべき誤りはすべて筆者たち個人に属する。

田村裕子　日本銀行金融研究所（E-mail: yuuko.tamura@boj.or.jp）
阿部正幸／奥田哲矢／津川天祐／宮澤俊之／山村和輝
NTT 株式会社社会情報研究所（E-mail: msyk.abe@ntt.com）
赤羽喜治／田口智貴／平栗勇人／増田博人／山田健斗
株式会社 NTT データ第三金融事業本部
（E-mail: yoshiharu.akahane@nttdata.com）
（宮澤俊之は、現・NTT 株式会社サービスイノベーション総合研究所）

1. はじめに

現在普及しているキャッシュレス決済には、「台帳」を用いて行う方式を採用したものが多い。ここでの台帳とは、ユーザからの指図に応じて取引内容を記載するデータベースであり、全ユーザのすべての取引内容や残高等がそこに集約管理されるものを指す。その代表例は預金取引であり、近年普及しているコード決済も同様の方式を採用しているとみられる。また、台帳の管理主体・方法に違いがあるものの、暗号資産[1]も台帳を用いた方式の一例として挙げられる。こうした台帳による方式では、ユーザがインターネットを介してサービス事業者（暗号資産の場合はブロックチェーンのノード）に送金指図を行い、サービス事業者が台帳を更新することで決済が実行される。

これに対し、「電子現金」[2]（岡本・太田［1993］、中山ほか［1997］）は、台帳を用いることなく決済を行うものであり、既存のキャッシュレス決済とは大きく異なる。電子現金とは、現金に見立てた電子データの送受信によって決済を行う手段であり、あたかも現金を受け渡すかのように、送金元と送金先の二者間におけるデータ通信によって決済を実行することができる。そのため、サーバ等の障害耐性、ネットワーク障害耐性、決済処理性能、外部システムとの相互運用性等の面で、台帳を用いた方式対比、メリットを有する可能性がある。また、決済が二者間に閉じることから、サービス事業者に対して取引内容を秘匿しうるほか、デバイスの近距離通信を使用すればインターネットから遮断された環境でも決済を実行することが可能である。

電子現金については、これまでいくつかの実証実験を通して、実運用性の検証が行われてきた[3]。1998年に行われた「インターネットキャッシュ」の実証実験では、インターネットを介して発行されたインターネットキャッシュ（電子現金）をICカードに保存し、インターネット上の店舗や友人に送信するシステムの動作検証が行われた。しかしながら、当時の技術水準ではユーザビリティ[4]を確保することが

1 2020年5月に施行された資金決済法の改正前は「仮想通貨」と呼称されていた。

2 中山ほか［1997］は、提案する決済方式を「電子マネー」と呼んでいたが、近年は非接触型ICカードを用いた決済方式を電子マネーと呼ぶことが多いため、本稿では、岡本・太田［1993］での呼称である「電子現金」を用いることとする。なお、本稿で考察を行う「電子現金」は、政府・日本銀行において検討を行っている中央銀行デジタル通貨（Central Bank Digital Currency: CBDC）とは異なるものである。

3 電子現金方式をベースとした実証実験には、日本電信電話株式会社［1995, 1996］、サイバービジネス協議会［2000］、エヌ・ティ・ティ・コミュニケーションズ株式会社［2000］の4つがある。

4 ユーザビリティとは、ユーザにとっての使いやすさを示す。利便性も類似語ではあるが、本稿では、2節（2）で定義する電子現金に求められる性質（安全性、利便性、現金がもつメリットの継承、透明性）の1つを指すものとする。

難しく、電子現金が実現することはなかった。その後、非接触型 IC カードを用いた電子マネーが普及したこともあり、電子現金に関する検討はいったん終息した。

近年、政府によるキャッシュレス推進の取組みにより、電子マネーをはじめとする各種キャッシュレス決済サービスの利用は年々増加している。2023 年には、国内のキャッシュレス決済比率が約 4 割まで上昇した（経済産業省［2024］）。政府は、キャッシュレス決済比率の最終目標を 8 割に据えており（経済産業省 商務・サービスグループ消費・流通政策課［2018］）、キャッシュレス決済サービスの利用は、今後より一層増えていくものと想定される。その場合、サーバ等の障害やネットワーク障害が国内の決済サービスに及ぼしうる影響について、より一層の考慮が必要になるように思われる。このように、安定したキャッシュレス社会の実現に向けては、台帳を用いない方式についても検討を進めておくことが有益であろう。

そこで、本稿では、まず、この間の技術進展および社会ニーズを踏まえて電子現金方式の再整理を行うとともに、現在広く利用されている一般的なスマートフォンを用いて実機検証を行った結果を報告する。さらに、電子現金のさらなるユーザビリティ向上に向けた方法、および、ユーザのプライバシを強化する方法について考察を行う。なお、本稿はあくまで電子現金にかかる技術面からの考察を行うものであり、法律や制度、実運用等、社会実装に向けた実現可能性についての検討は行っていない。

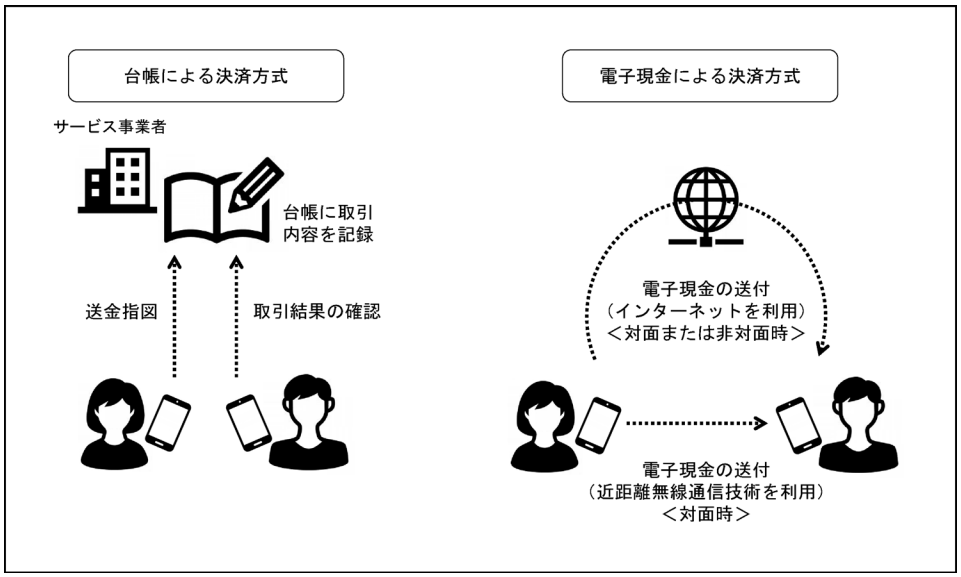
本稿の構成は以下のとおりである。まず、2 節において電子現金方式の基本的な構成とともに電子現金に求められる性質を整理するほか、プライバシ保護と透明性のバランスについて考察を行う。3 節では、電子現金の送受信に関する実機検証の結果を整理したうえで、現行の技術水準を前提とした場合のユーザビリティについて評価を行う。4 節では、電子現金方式の効率化に向けた検討として、送受信にかかる効率化方式、還収にかかる効率化方式、電子現金の分割・集約を可能とする方式について整理するほか、プライバシを強化した方式の考察を行う。最後に、5 節において今後の展望等を述べ、本稿を締め括る。

2. 電子現金方式の基本構成

(1) 台帳方式と電子現金方式との違い

現在提供されている決済サービスの多くは、サービス事業者が決済を仲介する方式となっており、ユーザから送金の指図を受けたサービス事業者がその内容を台帳に記載することで決済が行われる。預金取引の例であれば、金融機関が台帳にある

図表 1　台帳による決済方式と電子現金による決済方式



送金元ユーザの口座残高を減額するとともに、送金先ユーザの口座残高を増額することで、送信処理（振込処理）が実行される（図表 1 参照）。

一方、こうした台帳を用いずに決済を可能とする方法についても、従来検討が行われてきた（岡本・太田［1993］、中山ほか［1997］）。これらは、現金がそうであるように、送金元から送金先に電子データを送ることによる決済を目指したものであり、「電子現金」と呼ばれた。電子現金は、ユーザがもつスマートフォン等のデバイスに保管され[5]、その送付はインターネットを介して行われるほか、対面であればデバイスの近距離無線通信技術を利用して行うことも想定される（図表 1 参照）。

このように、電子現金では、サービス事業者の関与なしに決済を行いうる。ユーザ登録や電子現金の発行時にはサービス事業者の関与が必要となるが、送信処理は、送金元ユーザと送金先ユーザの二者間でのデータ通信に閉じることができる。そのため、電子現金の送受信処理においては、サーバ等障害耐性（サーバ等による障害の影響を受けにくい）、ネットワーク障害耐性（インターネットへ接続せずに決済できる可能性がある）、決済処理性能（サーバの処理能力や通信速度ではなく、デバイスの処理能力に依存する）、サービス事業者に対する取引内容の秘匿性

5 本稿ではユーザが使用するデバイスとしてスマートフォンを想定するが、電子現金方式は eSE（embedded Secure Element、埋込型セキュア・エレメント）と同様の耐タンパ性をもつ装置を具備した計算機にも実装可能である。SE（セキュア・エレメント）は、外部からの攻撃に対して高い安全性をもつモジュールであり、ハードウェアとソフトウェアを組み合わせることで実現される。スマートフォン内部に組み込まれるタイプは eSE と呼ばれる。なお、耐タンパ性については、脚注 9 を参照。耐タンパ性の必要性については、本節（4）で整理する。

（サービス事業者がリアルタイムで取引内容を知ることはない）を有すると整理することができる。

また、電子現金方式は、複数のサービスを容易に連携可能という特長も有する。例えば、複数のサービス事業者から発行された異なる電子現金を組み合わせた支払いや、異なる電子現金の交換といったニーズがあるとき、台帳を用いた方式であれば、サービス事業者がそれぞれに管理する台帳を API（Application Programming Interface）等により接続することで、これらサービスを連携させる必要が生じる。これに対し、電子現金方式では、そもそも台帳が存在しないことから、API 接続といったサービス事業者間の連携は不要である。なお、どちらの方式においても、ユーザ側では必要なアプリケーションをデバイスにインストールすることで、サービス連携への対応を行うことになる。

このほか、サービス事業者側のシステム改修を必要とせず、ユーザ側におけるアプリケーションによってサービス形態の変更に対応しうるという特徴を活かして、電子現金にプログラマビリティ[6]を与えることが考えられる。ここでのプログラマビリティとは、サービス事業者以外の主体であっても電子現金の使用にかかる独自のルールを適用可能であることを指し、適用範囲内のユーザがそれぞれに同ルールをデバイスにプログラムすることで独自ルールを実行することをいう。また、電子現金の発行時、サービス事業者が電子現金の使用にかかるルールを同電子現金に書き込むことで、プログラマブル・マネーを実現することも考えうる[7]。例えば、電子現金に使用期限を設けたり、電子現金の使用範囲を限定したりすること等が考えられよう。

一方、電子現金方式における留意点としては、電子現金がデバイスのみに保管されることに伴う影響が挙げられる。台帳方式では、すべての資産データをサービス事業者が管理しているため、仮にユーザがデバイスを紛失した場合であっても、ユーザ本人であることを確認さえできれば、当該ユーザのデータへのアクセスを許可することが考えられる。これに対し、電子現金方式では、ユーザのデバイス内に電子現金が格納されるため、デバイスを紛失すれば電子現金も紛失することになる[8]。こうした性質は現金と同様である。また、ユーザが使用するデバイスには、

6 決済システムにおけるプログラマビリティとは、決済システムの運営者のみならず、さまざまな主体が、個別のニーズにあわせて決済機能をプログラムできることをいう（北條・鳩貝［2022］）。

7 プログラマブル・マネーは、資金データという「オブジェクト」に固有の属性情報やプログラムを格納し、個別の振舞いを制御することに着目したコンセプトである（北條・鳩貝［2022］）。電子現金は、シリアル・ナンバに対するサービス事業者のデジタル署名として発行されるものであることから（本節（3）BOX1 参照）、同署名のメッセージ部分に、電子現金の使用にかかるルールを書き込むことが考えられる。

8 デバイスの紛失に備え、サービス事業者がすべての電子現金をバックアップしておくという対応も考えられるが、その場合にはバックアップデータの送信や保全が必要となることから、台帳方式と同様の障害リスクを有することになる。

一定の耐タンパ性[9]が必要となる（本節（4）参照）点にも留意が必要である。

本稿では、電子現金を保管する媒体として、ユーザが所有するスマートフォンのようなデバイスを想定しているが、将来的には、デバイス外のセキュアな領域で電子現金を保管できるようになることも考えられる。第6世代移動通信システム（6G）におけるネットワークの高度化[10]に向けた検討では、モバイル・ネットワーク自体にユーザの情報を強固に守るセキュリティ確保のための高度な暗号化・秘密計算機能をもたせることが検討されており、TEE（Trusted Execution Environment）環境[11]もその対象となっている（日本電信電話株式会社ネットワークサービスシステム研究所［2023］）。こうしたネットワークの高度化が実現した際には、デバイス所有者の電子現金をデバイスに紐付くエッジ（モバイル・ネットワークの入り口となる基地局等）上のTEE環境に格納することで、デバイスの紛失リスクに対応することが考えられる。

（2） 電子現金方式に求められる性質

電子現金に求められる性質として中山ほか［1997］が挙げている、安全性、利便性、現金がもつメリットの継承について、その概要を説明する。さらに、決済方式の提供に当たっては、マネー・ローンダリング・テロ資金供与防止（Anti-Money Laundering/Countering the Financing of Terrorism: AML/CFT）への考慮が必要となることから、本稿では、決済の透明性を電子現金に求められる性質に加えることとする。

（安全性）

① 電子現金の偽造・改ざん、および、複製・二重使用が困難であること

電子現金の安全性については、現金と同様、偽造や改ざんが困難であることが求められる。また、複製データの見分けが難しいという電子データの特性に伴い生じ

9 具体的な機能には、不正アクセスの証拠を残すタンパ・エビデンス、不正アクセスからデータを防護するタンパ・レジスタンス、不正アクセスに対してデータを消去する対抗動作を行うタンパ・レスポンスがある（田村・宇根［2007］）。

10 モバイル・デバイスを支える移動通信システムの高度化に伴い、データ転送だけでなくさまざまな情報処理機能をモバイル・ネットワークの入り口であるエッジ（基地局等）に配備する議論が進んでいる。デバイスはエッジとの間で機能分散を図り、モバイル・ネットワークを介してインターネット上のサービスにアクセスすることになる。

11 TEEは、高いセキュリティが要求される処理のための実行環境を提供する機能である。こうした実行環境（TEE空間）は、通常のアプリケーションが動作する空間（Rich Execution Environment: REE）と分離され、両空間の間の通信は厳格なアクセス制御のもとで実行されるように設計される。

る複製使用や二重使用が困難であることも必要な要件となる。安全性の観点から電子現金に想定される不正行為は、以下のとおりである。

- 電子現金の偽造・改ざん：発行者以外が電子現金を偽造したり、電子現金の金額等を改ざんしたりすること
- 電子現金の複製使用：他人が所有する電子現金を複製して使用すること
- 電子現金の二重使用：自分が所有する電子現金を複製して使用すること

電子現金方式では、デジタル署名と呼ばれる暗号技術を使用することで、電子現金の偽造・改ざんを防止する。また、第三者による複製使用については、電子現金に所有者の情報を付加することで防止する。二重使用については、台帳方式のように、決済サービス事業者がリアルタイムで検知することが困難であることから、デバイスの耐タンパ性によって防止する（本節（4）参照）とともに、耐タンパ性が低下した場合であっても事後検知できるよう、サービス事業者がシリアル・ナンバによって管理を行う[12]。

（利便性）

② 電子現金を任意の単位に分割して利用できること
③ 対面および非対面での決済が可能であること
④ 電子現金の発行・管理にかかる処理コストが高くないこと

電子現金方式の利便性とは、ユーザおよびサービス事業者における便利さの程度が現金決済におけるそれより高いことをいう。ユーザの視点では、電子現金を任意の単位に分割して利用できる性質や、対面・非対面決済のどちらにも利用できるといった性質へのニーズが高いと考えられる（性質②、③）。また、サービス事業者の利便性としては、発行・管理にかかる処理コストが現金のそれより低いことが挙げられる（性質④）。

ただし、本節（3）で紹介する基本方式、および、3 節で行う実機検証では、電子現金の単位を固定することとし、性質②を充足可能な方式は次の検討課題とすることとした。なお、電子現金を任意の単位に分割・集約を可能とする方式については、4 節（3）で整理している。

12 電子現金方式においては、複製・二重使用の事後検知のために、シリアル・ナンバのデータベース（DB）を必要とするが、決済を記録するための台帳とは異なる。

（現金がもつメリットの継承[13]）

⑤ 電子現金から過去の取引ユーザの特定が困難であること（匿名性[14]）
⑥ 同一ユーザが使用した電子現金の相互の関連付けが困難であること（関連付け不能性）
⑦ インターネットから遮断された環境でも決済が可能であること
⑧ 受領した電子現金を別の決済に使用できること
⑨ 電子決済に必要なデバイスを容易にもち運びできること

現金が有する性質の 1 つに匿名性（Anonymity）がある。現金の匿名性とは、現金を受け取った際、当該現金から過去の取引に関する情報を知ることができないことをいい、小売店や金融機関等が結託しても情報が露呈しないことをいう（古市［1995］）。電子現金についても、プライバシ保護の観点から同様の性質を有することが望ましく、過去に電子現金を使用したユーザの特定が困難といった性質が求められる（性質⑤）。また、大量のデータを収集し解析しうる状況においても、複数の電子現金から過去の取引が関連付けできないことが望ましい。具体的には複数の電子現金があるとき、それらが以前に同一ユーザの取引に用いられたものであっても、その事実が露呈しない性質は関連付け不能性（Unlinkability）と呼ばれる（性質⑥）。

現在普及しているキャッシュレス決済の多くは、インターネット通信の利用が前提となっている。しかし、自然災害等によるネットワーク障害の発生時においても決済できるようにしておくというニーズも少なくないことから、インターネットから遮断された環境でも決済できることが望ましい（性質⑦、中田［2021］）。また、ユーザビリティの観点からは、受領した電子現金をサービス事業者に還流させることなく、そのまま別の決済に使用できる性質（転々流通性[15]）が求められるほか、いつでもどこでも決済を可能とするよう、デバイスのポータブル性も必要と考えられる（性質⑧、⑨）。

（透明性）

⑩ 必要に応じて、サービス事業者は電子現金を追跡できること

13 中山ほか［1997］では、⑤と⑥の性質に加えて、複数金融機関対応という性質が挙げられている。中山ほか［1997］は、電子現金が預金を見合いとして発行され、預金に戻されることが想定されていることから、発行した金融機関以外への預入れを可能とする性質が望ましいとされていた。しかし、本稿では、決済サービスの提供者が電子現金を発行する想定としており、金融機関の関与を必須としないことから、複数金融機関対応という性質は含めないこととする。
14 中山ほか［1997］では、ユーザの「追跡不能性」と説明されている。
15 本性質を有する方式はオープンループ型と呼ばれ、そうでない方式はクローズドループ型と呼ばれる。

決済サービスには、AML/CFT の観点から、必要に応じて資金の流れを追跡可能とする決済の透明性が求められる。資金の流れを追跡するには、ユーザ情報の把握が必須であり、ユーザから提示された本人情報（氏名、住所等）に誤りがないことを確認する本人確認が必要となる。

ただし、サービス事業者がユーザに関するすべての情報を管理すれば、還収した電子現金の履歴からユーザを特定することが可能となりうる。そのため、ユーザの本人確認はサービス事業者とは独立した認証機関が行うこととする[16]。なお、ユーザのプライバシ保護と透明性については、本節（5）で整理する。

(3) 基本方式

ここでは、上記性質を有するよう設計された電子現金方式（中山ほか［1997］）を紹介する。電子現金方式を構成する主たるエンティティは、サービス事業者、認証機関[17]、ユーザ（電子現金の送受信者、店舗等を含む）、ユーザのデバイス（電子現金の送受信や保管を行うデバイス）である。基本となる電子現金方式は、ユーザ登録と証明書の発行[18]、電子現金の発行、電子現金の送信、電子現金の還収、電子現金の二重使用チェックのフェーズで構成される[19]（図表 2 参照）。具体的な流れは以下のとおり。なお、詳細な暗号処理の方法等については、BOX1 を参照されたい。

イ．ユーザ登録と証明書の発行

電子現金の利用に当たり、ユーザは暗号処理に必要な鍵ペア[20]の生成を行い、認証機関から証明書の発行を受ける。

① ユーザは、証明書発行に必要な本人情報を認証機関に提出する。

16 ユーザのプライバシ保護の観点から、中山ほか［1997］と同様、ユーザの情報を管理する機関を独立に設けることとした。もっとも、本稿では現行法制度に照らした考察は検討の対象外としていることから、法制度との関係については別途の検討が必要である。

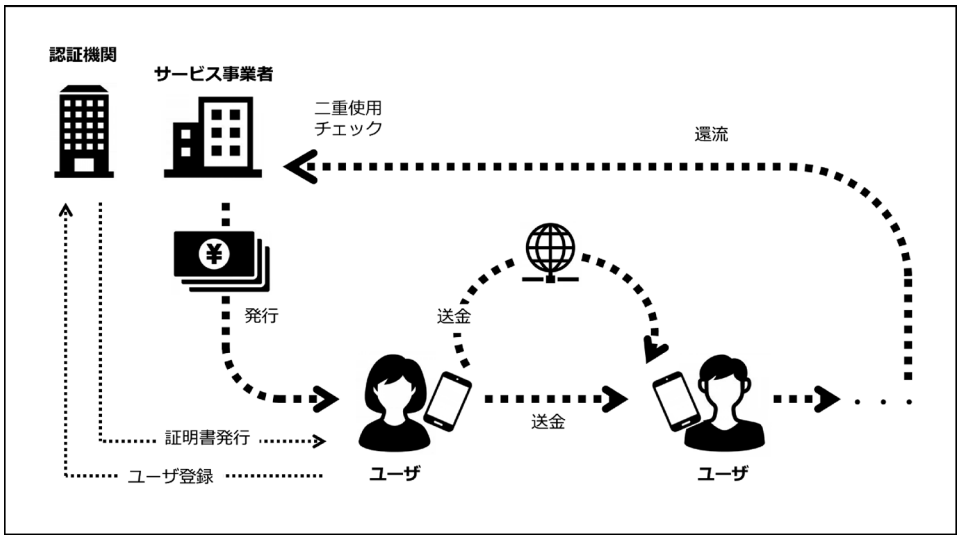
17 認証機関は、ユーザの本人確認を行ったうえで、ユーザの公開鍵に対する証明書を発行する。複数のサービス事業者が 1 つの認証機関と連携することも考えられる。

18 認証機関によって発行される証明書は、暗号処理に使用された鍵ペアが認証機関に登録された正しいものであることを第三者が確認するのに用いられる。

19 特定の管理者をもたない暗号資産では、一般に、ユーザ登録や証明書の発行といった手続きはない。そのため、暗号資産ユーザが特定されることはないほか、鍵が盗取された場合であっても同鍵を無効化することができない。もっとも、暗号資産取引に暗号資産交換業者を利用する場合には、暗号資産交換業者等による本人確認が義務付けられているため、この限りではない。また、暗号資産は台帳ですべて管理されているため、還収処理も発生しない。

20 暗号処理には、秘密鍵・公開鍵と呼ばれる鍵ペアが用いられる。秘密鍵はユーザのみが知る情報として管理する必要があるのに対し、公開鍵は他ユーザへの公開が可能な情報である。デジタル署名方式では、公開鍵を用いた処理によって、署名対象であるメッセージが秘密鍵の所有者によって作成されたものであること、および、事後的に改ざんされていないことを確認できる。

図表２　電子現金スキームの基本的構成



② 認証機関は、ユーザの本人確認を行い、ユーザ情報を登録する。
③ ユーザは、秘密鍵と公開鍵の鍵ペアを生成し、公開鍵を認証機関に送信する。
④ 認証機関は、公開鍵をユーザ情報に対応付けて管理するとともに、公開鍵に対する証明書をユーザに発行する。

ロ．電子現金の発行

ユーザの依頼に応じて、サービス事業者はユーザに電子現金を発行する。電子現金には、偽造・改ざん防止対策としてサービス事業者のデジタル署名が付与されるほか、所有者を指定して発行することで複製・二重使用への対策を講じる。そのため、電子現金は、その所有者情報を記載した移転履歴と呼ばれるデータとセットで取り扱う。詳細については BOX2 を参照されたい。

① サービス事業者は、電子現金の発行を希望するユーザに証明書を要求し、その正当性を検証する。証明書の有効性については、認証機関による証明書失効リスト（CRL: Certificate Revocation List）で確認を行う。
② サービス事業者は、ユーザが希望する金額分の電子現金を発行する[21]。電子現金にはシリアル・ナンバを割り当て、発行した電子現金のシリアル・ナンバを「発行済電子現金 DB（データベース）」に追加する。さらに、それぞれの電子現金に対して、サービス事業者からユーザに発行（移転）されたことを示す

21 本電子現金方式では、発行される電子現金の額面を固定とする。現金と同じであれば、10 種類（10,000 円、5,000 円、2,000 円、1,000 円、500 円、100 円、50 円、10 円、5 円、1 円）となる。

データとなる移転履歴を付与する。

③ ユーザは、サービス事業者の証明書を検証するとともに、電子現金とその移転履歴が正しく生成されていること（電子現金が偽造・改ざんされたものでないこと、電子現金と移転履歴が同じシリアル・ナンバに対応していること、自分が送金先として指定されていること）を確認し、デバイス内に保存する。

ハ. 電子現金の送信

ユーザは、電子現金を任意のユーザや店舗に送付することができる。その際、送付先となるユーザの情報を移転履歴に追加することで、複製・二重使用への対策を講じる（詳細は BOX2 を参照）。

① ユーザは、送金先ユーザの証明書を要求し、その正当性を検証する[22]。

② ユーザは、電子現金の移転履歴に送信先ユーザの情報を追加し、電子現金、移転履歴、自身の証明書を送付する。

③ 送金先ユーザは、送金元ユーザの証明書を検証するとともに、受け取った電子現金と移転履歴が正しく送付されたものであること（電子現金が偽造・改ざんされたものでないこと、電子現金と移転履歴が同じシリアル・ナンバに対応していること、他人の電子現金を複製したものでないこと、自分が送金先として指定されていること）を確認する。その際、サービス事業者による発行以降のすべての移転履歴が正しく生成されていることの確認も行う[23]。その後、電子現金と移転履歴をデバイス内に保存する。

ニ. 電子現金の還収

事後的な電子現金の二重使用チェック、および、電子現金の再発行のため、一定

22 証明書の検証によって、相手ユーザが認証機関に登録された正しいユーザであること、および、証明書が有効であることを確認する。後者を確認するには、認証機関による CRL を確認する必要がある。インターネットに接続できる状況であれば、最新の CRL を確認できるが、そうでない状況に備えて、インターネットに接続したタイミングで速やかに CRL をデバイスに自動的にダウンロードする仕様としておくことが必要である。

23 データに改ざんや不整合がないことを確認するために行う。改ざんや不整合があった場合、送金先ユーザは電子現金の受取りを拒否する。なお、移転履歴にあるその他のユーザの証明書は確認しないため、移転履歴の中に、サービス事業者へのユーザ登録を行っていない（証明書をもたない）ユーザや他人から盗取した秘密鍵（CRL に掲載されている公開鍵に対応する秘密鍵）を使用したユーザが紛れ込んでいたとしてもこれを検知することはできない。こうした不正なユーザが二重使用していたとしても、サービス事業者は同ユーザの情報をもたないことから、同ユーザを特定することができない。しかしながら、正しいアプリケーションを使用しているユーザであれば、証明書をもたないユーザとは取引できない（アプリケーションが取引を拒否する）はずであることから、不正なユーザに電子現金を送信したユーザ、および、不正なユーザから電子現金を受け取ったユーザは共犯者である可能性が高いといえる。このため、サービス事業者は、不正にかかわったユーザを特定することで、二重使用者を追跡できる可能性があると考えられる。

の頻度で電子現金をサービス事業者に還流させる必要がある[24]。サービス事業者による還収は、ユーザによる電子現金の払戻し（換金）のタイミングのほか、移転回数が規定を超えたタイミングで強制的に行うことが考えられる[25]。

① ユーザは、サービス事業者に電子現金と移転履歴のセットを送信する。
② サービス事業者は、受け取った電子現金と移転履歴のセットが正しく送付されたものであることを確認する。その際、移転履歴については、発行以降のすべての移転履歴が正しく生成されていることを確認するとともに、移転履歴にある公開鍵の有効性を CRL の参照によって確認する。
③ サービス事業者は、電子現金の二重使用チェックを行う（後述）。
④ サービス事業者は、ユーザから電子現金の再発行依頼があれば、同金額分の電子現金を改めて発行する。

ホ．電子現金の二重使用チェック

サービス事業者は、還収した電子現金のシリアル・ナンバを確認することで、二重使用チェックを行う。

① サービス事業者は、発行済電子現金 DB を参照し、還収した電子現金のシリアル・ナンバが DB にあるか否かを確認する。
② 発行済電子現金 DB に同シリアル・ナンバがあれば、DB から同シリアル・ナンバを削除して、還流してきた電子現金と移転履歴を「還収済電子現金 DB」に格納する。
③ 発行済電子現金 DB に同シリアル・ナンバがなかった場合には、二重使用されたものと判断し、還収済電子現金 DB から同じシリアル・ナンバをもつ電子現金の移転履歴を取り出し、両移転履歴から二重使用したユーザの公開鍵を特定する。
④ サービス事業者は、二重使用したユーザの公開鍵を認証機関に送信し、公開鍵に対応するユーザを特定する[26]。

24 電子現金に付与される移転履歴のサイズは、移転回数に比例して増加するため、移転回数が多くなった電子現金については還収のうえ再発行することが望ましい。
25 中山ほか［1997］では、電子現金の発行機関が金融機関であるため、口座への預入れという形で還流してきた電子現金を二重使用チェックすることができる。本方式においても、サービス事業者が金融機関と連携することで、金融機関に預け入れられた電子現金をサービス事業者に還流させて二重使用チェックを行うことも考えられる。
26 こうした対応には、サービス事業者と認証機関による別途の契約とユーザの事前許諾が必要である。

BOX 1　電子現金方式を構成する 6 つのフェーズ

（初期設定）

① サービス事業者 I は、額面ごとに秘密鍵と公開鍵のペアを生成し、認証機関から発行された証明書とともに公開鍵を公開する。ここで、Y 円用の鍵ペアを $(sk_{I(Y)}, pk_{I(Y)})$ とする。
② 認証機関 C は、秘密鍵と公開鍵のペアを生成し、自己証明書とともに公開鍵を公開する。認証機関 C の鍵ペアを (sk_C, pk_C) とする。

（ユーザ登録と証明書の発行）

① ユーザ U は、証明書発行に必要な本人情報を認証機関 C に提出する。
② 認証機関 C は、ユーザ U の本人確認を行い、ユーザ情報を登録する。
③ ユーザ U は、秘密鍵と公開鍵のペア (sk_U, pk_U) を生成し、公開鍵 pk_U を認証機関 C に送信する。
④ 認証機関 C は、公開鍵 pk_U をユーザ情報に対応付けて管理するとともに、pk_U に対する証明書 $Cer_U \leftarrow Sign_{sk_C}(pk_U)$ をユーザ U に発行する。$Sign$ は署名生成関数であり、$Sign_{sk}(m)$ はメッセージ m に対する秘密鍵 sk によるデジタル署名を表す。証明書は認証機関によるデジタル署名となっていることから、その正当性はデジタル署名の検証処理 $1/0 \leftarrow Verify_{pk_C}(pk_U, Cert_U)$ によって確認される。$Verify_{pk}(m, \sigma)$ は、メッセージ m に対するデジタル署名 σ の検証式を表し、公開鍵 pk を用いることで、デジタル署名 σ が pk に対応する秘密鍵を用いて生成されたこと、および、メッセージ m が改ざんされていないことを確認できる。$Verify$ は、σ が正しい署名であれば 1、そうでなければ 0 を出力する。

（電子現金の発行）
・サービス事業者 I がユーザ U に Y 円分の電子現金を発行する

① サービス事業者 I は、電子現金の発行を希望するユーザ U に公開鍵 pk_U とその証明書 Cer_U を要求し、Cer_U の正当性を $Verify_{pk_C}(pk_U, Cer_U)$ によって検証する。また、認証機関による CRL を参照し、pk_U の有効性を確認する。
② サービス事業者 I は、ユーザ U が希望する金額分の電子現金を発行する。SN_i をシリアル・ナンバとするとき[27]、Y 円分の電子現金は、SN_i に対するサ

27 添字の i は、複数の電子現金を扱う場合の通番である。

ービス事業者 I のデジタル署名 $T_i \leftarrow Sign_{sk_{I(Y)}}(SN_i)$ として表される。また、T_i の移転履歴として、$\sigma_{i(0)} \leftarrow Sign_{sk_{I(Y)}}(SN_i \| pk_U)$ を生成し、T_i と $\sigma_{i(0)}$ をユーザ U に送信する。その際、$(T_i, \sigma_{i(0)})$ の検証に必要となるシリアル・ナンバ SN_i もあわせて送付する。ここで、$\|$ はデータの連結を表し、$\sigma_{i(n)}$ は T_i の n 回目の移転履歴を表す（発行を 0 回目と数える）。その後、サービス事業者は、シリアル・ナンバ SN_i を発行済電子現金 DB に追加する。

③ ユーザ U は、サービス事業者の証明書を検証するとともに、電子現金と移転履歴 $(T_i, \sigma_{i(0)})$ が正しく発行されていることを $Verify_{pk_{I(Y)}}(SN_i, T_i), Verify_{pk_{I(Y)}}(SN_i \| pk_U, \sigma_{i(0)})$ によって確認し、いずれの出力も 1 であれば、デバイス内に保存する。

（電子現金の送信）

・ユーザ U がユーザ V に m 枚の電子現金を送信[28]

① ユーザ U は、送金先ユーザ V に公開鍵 pk_V とその証明書 Cer_V を要求し、その正当性を $Verify_{pk_C}(pk_V, Cer_V)$ によって検証する。

② ユーザ U は、電子現金 T_i の移転履歴 $\sigma_{i(n)}$ を更新するに当たり、デジタル署名 $\sigma_{i(n+1)} \leftarrow Sign_{sk_U}(\sigma_{i(n)} \| pk_V)$ を生成する。$\sigma_{i(n+1)}$ のメッセージ部分には、$\sigma_{i(n)}$ とユーザ V の公開鍵 pk_V が含まれる。ユーザ U は、電子現金とその移転履歴 $(T_i, \{\sigma_{i(\ell)}\}_{0 \le \ell \le n+1})$ をユーザ V に送付する。その際、これらの検証に必要となるシリアル・ナンバ SN_i、移転履歴にあるユーザの公開鍵、自身の公開鍵と証明書もあわせて送付する。m 枚の電子現金を送付するに当たり、これを m セット送付する。

③ ユーザ V は、ユーザ U の証明書 Cer_U を $Verify_{pk_C}(pk_U, Cer_U)$ によって検証するとともに、受け取った電子現金と移転履歴が正しいことを確認し、デバイス内に保存する。電子現金 T_i とその移転履歴 $\{\sigma_{i(\ell)}\}_{0 \le \ell \le n+1}$ の正しさとは、(1) T_i と $\sigma_{i(0)}$ がいずれも SN_i に対するサービス事業者の署名であること、(2) $0 \le \ell \le n-1$ について、$\sigma_{i(\ell+1)}$ が $\sigma_{i(\ell)}$ とユーザの公開鍵に対する署名であり、その署名者が $\sigma_{i(\ell)}$ のメッセージ部分に含まれる公開鍵に対応するユーザであること、(3) $\sigma_{i(n+1)}$ が $\sigma_{i(n)}$ と自身の公開鍵に対する送金元ユーザ U の署名となっていることをいう。

なお、(1)は、あるユーザの公開鍵 pk に対して、$1 \leftarrow Verify_{pk_{I(Y)}}(SN_i, T_i)$ と $1 \leftarrow Verify_{pk_{I(Y)}}(SN_i \| pk, \sigma_{i(0)})$、(2)は、$0 \le \ell \le n-1$ に対し、あるユーザの公開鍵

28 電子現金は額面単位で発行されるため（脚注 21）、送信の際は枚数で数えることとする。

pk について、$1 \leftarrow Verify_{pk}(\sigma_{i(\ell+1)} \| pk', \sigma_{i(\ell+2)})$ と $1 \leftarrow Verify_{\tilde{pk}}(\sigma_{i(\ell)} \| pk, \sigma_{i(\ell+1)})$ が成り立つことをいう。このとき、pk' は公開鍵を pk とするユーザが T_i を送付したユーザの公開鍵であり、$\tilde{pk}$ は公開鍵を pk とするユーザに T_i を送付したユーザの公開鍵である。また、(3) は、$1 \leftarrow Verify_{pk_U}(\sigma_{i(n)} \| pk_V, \sigma_{i(n+1)})$ が成り立つことを示す。

（電子現金の還収）

① ユーザ V は、サービス事業者に電子現金とその移転履歴 $(T_i, \{\sigma_{i(\ell)}\}_{0 \leq \ell \leq n+1})$、シリアル・ナンバ SN_i、移転履歴にあるユーザの公開鍵、自身の公開鍵と証明書を送信する。
② サービス事業者 I は、受け取った $(T_i, \{\sigma_{i(\ell)}\}_{0 \leq \ell \leq n+1})$ が正しいものであることを確認する。移転履歴にある公開鍵の有効性については、認証機関による CRL への照会によって確認する。

（電子現金の二重使用チェック）

① サービス事業者 I は、発行済電子現金 DB を参照し、還収した電子現金 T_i のシリアル・ナンバ SN_i が DB にあるか否かを確認する。
② 発行済電子現金 DB に SN_i があれば、DB から SN_i を削除して、還流してきた電子現金と移転履歴を還収済電子現金 DB に格納する。
③ DB にシリアル・ナンバ SN_i がなかった場合には、二重使用されたものと判断し、還収済電子現金 DB から SN_i をシリアル・ナンバとする電子現金と移転履歴を取り出し、両移転履歴から二重使用したユーザの公開鍵を特定する。例えば、同じシリアル・ナンバをもつ 2 つの移転履歴について、公開鍵 pk_{U_1} に対応するユーザの送信先が、一方が pk_{U_2}、もう一方が pk_{U_3} となっていれば、同ユーザが電子現金を二重使用したと特定できる。
④ サービス事業者は、二重使用したユーザの公開鍵を認証機関に送信し、公開鍵に対応するユーザを特定する。

BOX 2　電子現金の偽造・改ざん対策と複製・二重使用対策

- 電子現金の偽造・改ざん対策（発行機関以外による電子現金の偽造、発行機関以外による電子現金の金額や所有者情報等の改ざんへの対策）：電子

現金 T は、発行機関 I によるシリアル・ナンバ SN に対するデジタル署名 $T \leftarrow Sign_{sk_I}(SN)$ として表される。デジタル署名技術により、第三者による電子現金の偽造・改ざんを防止することができる。

- 第三者による電子現金の複製対策（他人がもつ電子現金を複製して使用する不正への対策）：電子現金と所有者の対応付けにより、第三者が電子現金を複製して使用する不正を防止する。つまり、電子現金とセットとなる移転履歴にユーザの公開鍵を改ざん困難な形式で追加することで、同ユーザだけが当該電子現金を使用できるようになっている[29]。

- 複数ユーザへの電子現金の二重使用対策（正規所有者が電子現金を複製して使用＜受領した電子現金を複製して、複数ユーザにそれぞれ送信＞する不正への対策）：電子現金を送信する際には、秘密鍵による署名生成が必要であることから、デバイスの耐タンパ性によって不正を防止することが可能である。もっとも、技術進展によりデバイスの耐タンパ性は低下するおそれがあり、そうした事態への備えは必須である（本節（4）を参照）。そのため、サービス事業者が事後的に不正者を特定できるよう、電子現金には、その移転履歴を改ざん困難な形式で付与する。
  - ➢ 例えば、同じシリアル・ナンバ SN_i をもつ電子現金が 2 つ確認され、それぞれが下記であったとき、$\sigma_{i(\ell-1)}(= Sign_{sk_{U_1}}(\sigma_{i(\ell-2)} \| pk))$ のメッセージ部分に含まれる公開鍵 pk について、一方が pk_{U_2}、もう一方が pk_{U_3} となっていれば、$\sigma_{i(\ell-1)}$ の署名者（公開鍵 pk_{U_1} に対応するユーザ）が 2 人のユーザに同じ電子現金を送付したことがわかる（pk_{U_4} と pk_{U_5} は、その後の移転先となったユーザの公開鍵）。

$$\sigma_{i(\ell)} = Sign_{sk_{U_2}}(Sign_{sk_{U_1}}(\sigma_{i(\ell-2)} \| pk_{U_2}) \| pk_{U_4}). \quad (1)$$

$$\sigma_{i(\ell)} = Sign_{sk_{U_3}}(Sign_{sk_{U_1}}(\sigma_{i(\ell-2)} \| pk_{U_3}) \| pk_{U_5}). \quad (2)$$

- 同一ユーザへの電子現金の複製・二重使用対策（正規所有者あるいは第三者が電子現金を複製して使用＜あるユーザに送信した電子現金を複製して、当該ユーザに再送信＞する不正への対策）：移転履歴にワンタイム性を付与する（時刻情報を入れる等）、あるいは、電子現金の送信処理をチャレンジ・レスポンス方式[30] とすることで対応が可能である。

29 攻撃者が電子現金を複製する方法としては、ユーザ間の通信路を盗聴する、あるいは、デバイスから不正に読みだすことが考えられる。

30 検証者が毎回ランダムに生成する値を用いてリプレイ攻撃への対策を講じるもの。リプレイ攻撃と

(4) セキュリティに関する考察

イ. デバイスの耐タンパ性

電子現金方式の安全性は、デジタル署名の安全性に依拠する部分が大きいことから、デジタル署名に使用する秘密鍵を安全に管理できるデバイスの利用が前提となる。耐タンパ性を有するデバイスであれば、デバイスの正規所有者であっても内部の秘密鍵を不正に読みだすことが困難であるほか、アプリケーションの改ざんも難しくできることから、電子現金の二重使用を防止することができる。もっとも、悪意のあるユーザが、意図的に耐タンパ性をもたないデバイスを利用して不正を行うことも考えられるため、サービス事業者にはユーザが使用するデバイスの耐タンパ性確認が必要となろう。例えば、ユーザがもつデバイスの耐タンパ性が確認できたときに限り、ユーザ登録を許可するといった運用とすることで、不正を試みるユーザを排除することが考えられる。

もっとも、デバイスのセキュリティ機構に対する攻撃手法の進展に伴い、耐タンパ性は低下する。そのような状況になれば、上記対策を講じていたとしても、電子現金の二重使用を防止できるとは限らない。また、サービス事業者は、デバイスに標準装備されたセキュリティ機能を前提にアプリケーションを構成するが、こうしたセキュリティ機能が期待通りに動作しないケース等においては、アプリケーションのリスク評価と管理を適切に行うことが困難との指摘もある（磯部・宇根［2021］）。このため、電子現金方式では、デバイスの耐タンパ性低下等に備え、電子現金の移転履歴を改ざん困難な形式で電子現金に付与することとし、還流してきた電子現金のシリアル・ナンバをチェックすることによって、ユーザによる二重使用を事後的に検知できるようにしている[31]。

ロ. ユーザとデバイスと鍵の関係

電子現金は、他人による複製使用といった不正を防止するため、電子現金に紐付けられた「秘密鍵」を用いてしか使用（他ユーザへの送信）できないように設計されている。このとき、「ユーザ（人）」と秘密鍵を保管する「デバイス」の関係には以下のパターンが考えられる。

は、通信路を盗聴して得たデータをそのまま利用してなりすましを行う攻撃である。検証者が送信するランダムな値をチャレンジ、それに対する証明者の回答をレスポンスと呼ぶ。

31 電子的に決済を行う方法には、電子マネーに採用されているような、他ユーザから受領した金銭的価値を合算してデバイス内に保管・管理して決済を行う方式（残高管理型と呼ばれる）もある。残高管理型においても、仮に耐タンパ性が低下してしまった場合には、デバイス内のデータを不正に操作することで、金銭的価値を無制限に増やして使用することができてしまう。こうした不正を事後的に検知するには、すべての取引をサーバ側の台帳で管理することが必要となる（祖山［2020］）。

- ケース A：デバイスをもっている者であれば、誰でもデバイス内の電子現金を使用できる（*n* 対 1 対応）。
- ケース B：デバイスの正規所有者のみが、デバイス内の電子現金を使用できる（1 対 1 対応）。

ケース A は現金に近い使用方法であり、ユーザビリティも高いように考えられるが、事後的な不正検知、および、電子現金の透明性確保のため、電子現金については、ケース B での使用形態を想定する。ケース B は電子現金を「ユーザ（人）」に紐付けるものであり、当該ユーザだけが同電子現金を使用できるようにするものである。ケース B を実現する方法としては、デバイスの使用者を制限する機能を付す、あるいは、本人だけが秘密鍵を生成できる方法を採用する[32] こと等が考えられる。

(5) プライバシ保護と透明性に関する考察

イ. プライバシ保護と透明性のバランス

決済サービスでは、ユーザのプライバシへの配慮が必要との意見がある。例えば、金融調査研究会［2018］は、新たな決済手段が受け入れられる条件として、現金と同等の安心・安全な決済手段であることを挙げており、サービス事業者はプライバシおよび個人情報の保護に努めるべきと提言している。

一方、AML/CFT の観点からは、事後的に資金の流れを探る必要性が生じるケースがあり、そうした場合に備えて追跡可能性を有することが決定的に重要とされている（野田［2022］）。現金は追跡可能性をもたないが、現金の保管・流通に物理的制約があることから、マネー・ローンダリング等の媒体になる機会が限定されていた。しかし、電子的な送信・決済手段についてはマネー・ローンダリング等の媒体になるリスクがあることから、事後的な追跡ができるような手当が必要となる。

このように、事業者が決済サービスを提供するに当たっては、プライバシ保護と透明性とのバランスが重要となる。そこで、以下では、他ユーザに対するプライバシ保護とサービス事業者に対するプライバシ保護のそれぞれについて考察を行う。

ロ. 他ユーザに対するプライバシ保護

本節（3）で整理したとおり、電子現金の送受信時には、送金元ユーザと送金先ユーザ間で互いに証明書の検証を行う。そのため、個人を特定可能な情報（氏名、住所等）が証明書に含まれれば、それにより、相手ユーザの特定が可能となってし

32 例えば、PBI（Public Biometric Infrastructure）（高橋［2021］）によって、ユーザの生体情報から、都度、秘密鍵を生成することが考えられる。

まう[33]。店舗での支払いでは、相手ユーザの特定が不要であることから、個人を特定しうる情報は証明書に記載しないことが求められよう。

また、電子現金は移転履歴とセットで送受信され、移転履歴には電子現金の所有者であったユーザの公開鍵が含まれる。公開鍵（あるいは証明書）にユーザを特定可能な情報が含まれていなくとも、公開鍵そのものがユーザの ID となって、さまざまな情報を関連付けできてしまう可能性がある。例えば、別途のチャネル（例えば、対面決済）から取引相手ユーザに関する情報を入手できれば、ユーザと公開鍵との対応関係を知ることができ、ユーザが使用した電子現金の相互の関係性を特定できる可能性が生じる。そのため、電子現金の匿名性と関連付け不能性を満たすには、決済の都度、公開鍵を更新することが必要となる（中山ほか［1997］）[34]。そのほか、ゼロ知識証明と呼ばれる暗号技術を使用することで、証明書そのものを開示することなく、自身が電子現金の正当な所有者であることを証明することも考えられる。ゼロ知識証明を使用したプロトコルについては、4 節で説明する。

ハ．サービス事業者に対するプライバシ保護と透明性

サービス事業者は電子現金の発行時、発行先となるユーザの証明書を検証するが、証明書に本人を特定しうる情報を記載しないといった対応により、認証機関と結託しない限り、サービス事業者に対するプライバシは確保される。

また、電子現金の取引は、ユーザ間に閉じて実施されることから、サービス事業者がリアルタイムで取引内容を知りうることはない。また、サービス事業者は還収した電子現金の移転履歴を確認しうるが、移転履歴から入手できるのはユーザの公開鍵だけであり、認証機関と結託しない限り、ユーザを特定することはできず、電子現金の匿名性は満たされる。

さらには、認証機関における不正対策を強化する方法として、ユーザ情報の管理を行う機能と証明書の管理・発行を行う機能を分けることが考えられる。これらの機能を分割した場合の証明書発行方式については、補論 1. を参照されたい。なお、還収した電子現金から二重使用したユーザを特定できる性質を維持しつつ、それ以外のユーザに関する情報は何も得られないようにすることは技術的に可能である。こうしたサービス事業者に対するプライバシを強化した方式については、4 節で考察を行う。

そのほか、本節（5）ロ．で整理したとおり、ユーザの取引相手においては、当

33 ユーザには受信したデータの内容を開示しないようアプリケーションを構成することも考えられる。しかし、多少のスキルがあれば、スマートフォン間で送受信されるデータを複製することは可能であることから、送受信されるデータがユーザのプライバシに与える影響について検討しておくことは重要である。

34 電子現金の使用を 1 回とすることでユーザのプライバシ保護を図る方法（例えば、Chaum [1983]）も別途存在するが、電子現金に求める転々流通性を満たさない。

該ユーザと公開鍵との対応関係を知りうる可能性がある。そのため、対応関係を知る取引相手と結託することで、サービス事業者は還収した電子現金から当該ユーザの取引を特定することが可能となる。こうした結託が行われた場合においても、電子現金の匿名性と関連付け不能性を満たすには、他ユーザへのプライバシ保護対策と同様、決済の都度、公開鍵を更新することが必要となる。

電子現金に求められる透明性の観点からは、定期的に電子現金の移転履歴を確認することが望ましいことから、移転回数が x 回を超えた場合や発行後 y 期間が経過した場合等にはサービス事業者に還流するよう設計しておくといったことが一案となろう。こうした対応により、サービス事業者は、電子現金の不審な動向を補足できるようになると考えられる。さらに、高額送金のケースのように、必要な情報をリアルタイムで収集する必要がある場合には、インターネットに接続可能な場合に限り高額送金を許可したうえで、電子現金取引の写しをアプリケーションから自動的にサービス事業者に送信するといった対応も考えられよう。そのほか、電子現金のブロック・リストを作成することで、不審な履歴をもつ電子現金の流通を制限することも考えられる。これらは電子現金が有するプログラマビリティによって実現することができる。

3. 電子現金方式の実機検証

(1) 過去の実証実験

電子現金方式をベースとした実証実験は、これまでに複数回行われている。いずれも接触型 IC カードに電子現金を格納し、インターネットを経由して店舗や他ユーザに送信する形態の実証実験であった。電子現金は、預金口座からの引落しという形で金融機関から発行され、それをインターネット上の店舗（バーチャル店舗）や実店舗で使用することができるものであった。当時の実証実験は、こうした電子現金が現金に代わる新たな決済手段として機能しうるかを検証したものであり、実サービスの提供に向けた課題抽出を目的として行われた。

• インターネットキャッシュ（1998 年）

「インターネットキャッシュ」の実証実験（1998 年 9 月～2000 年 2 月）は、4 つの金融機関と約 1 万人の一般参加者の協力を得て行われた。IC カードに格納されたインターネットキャッシュは、パソコンに接続されたリーダ・ライタを介して、バーチャル店舗への支払いや他のユーザへの送信に使用可能であった。また、ユー

ザ間では、インターネットキャッシュの転々流通も可能であった。金融機関から発行されたインターネットキャッシュは、預金口座への払戻しも可能とされた（サイバービジネス協議会［2000］）。

• スーパーキャッシュ（1999 年）

「スーパーキャッシュ」の実証実験（1999 年 4 月～2000 年 5 月）は、24 の金融機関、約 1,000 の店舗、約 2.2 万人の参加者による協力のもと行われた。本実証実験は、市中での利用可能性に焦点を当てたものであったことから、バーチャル店舗に加えて実店舗での支払いも可能とされたほか、IC カードへのチャージには金融機関に設置されたチャージ機や公衆電話を利用することができた。もっとも、店舗で使用されたスーパーキャッシュは、データ・センターに伝送される仕様となっており、転々流通性は検証の対象外とされた（エヌ・ティ・ティ・コミュニケーションズ株式会社［2000］）。

インターネットキャッシュとスーパーキャッシュについては、当時行われた実証実験の概要が公表されているものの、電子現金の送受信にかかった時間については記載されていない。しかし、当時の IC カード仕様や通信環境を考慮すれば、ユーザビリティの確保が難しかったと推測される。

(2) 実機検証の概要

上記実証実験から約 25 年が経過し、われわれを取り巻く技術環境は大きく変化した[35]。そこで、現在の技術レベルを前提に、改めて電子現金のユーザビリティについて検証を行うこととする。今次実機検証では、電子現金方式の実装にかかる検

35 2000 年におけるインターネット利用率は 37.1％であり、電話回線によるダイヤルアップ接続が主流であった。2000 年に商用提供が開始された非対称デジタル加入者線（Asymmetric Digital Subscriber Line: ADSL）の最大通信速度は、下りが 50 メガビット毎秒、上りが 5 メガビット毎秒であった。これに対し、2023 年におけるインターネット利用率は 84.9％となったほか、2020 年に提供が開始された第 5 世代移動通信システム（5G）の最大通信速度は、下りが 4.2 ギガビット毎秒、上りが 218 メガビット毎秒となっており、それぞれ ADSL の 84 倍、約 43 倍となっている。また、1999 年頃に日本電信電話株式会社が開発した高機能用途 IC カードは、中央演算処理装置（Central Processing Unit: CPU）クロック周波数が 15 メガヘルツ、随時書込み読出しメモリ（Random Access Memory: RAM）容量が 2 キロバイト、フラッシュ・メモリ容量が 512 キロバイトであった。一方、現在の一般的なスマートフォンは、CPU クロック周波数が数ギガヘルツのマルチコア構造であり、RAM 容量が数ギガバイト、フラッシュ・メモリ容量が数百ギガバイトとなっているうえ、これらのデバイスの多くには IC カードと同様に耐タンパ性をもつ SE が内蔵されている。2022 年発売のスマートフォン（Google Pixel 7）に採用された eSE である ST54K（ST マイクロエレクトロニクス社製）は、CPU クロック周波数が 100 メガヘルツ、RAM 容量が 64 キロバイト、フラッシュ・メモリ容量が 2,048 キロバイトとなっている。なお、Google Pixel は、Google LLC の商標または登録商標である。

討の端緒として、電子現金の送信処理に焦点を当てる。また、総合的なユーザビリティの評価には、アプリケーションの操作性等も含まれるが、今次検証では、電子現金の送信にかかる時間によってユーザビリティを測ることとする。

イ．使用する機器のスペック等

電子現金の実装には、秘密鍵等の重要な情報を格納するための耐タンパ・デバイスのほか、電子現金を送受信するための通信機能が必要となる。スマートフォンには、耐タンパ・デバイスである eSE が搭載されているほか[36]、モバイル・データ通信、Wi-Fi[37]、Bluetooth[38]、NFC（Near Field Communication）といった通信機能も具備されていることから、今次検証では、スマートフォンの利用を想定することとした。なお、国内におけるスマートフォンの普及率は 2024 年時点で 97％となっており（NTT ドコモモバイル社会研究所［2024］）、一般に広く普及したデバイスであるといえる。また、電子現金方式で使用する暗号アルゴリズムは、電子政府推奨暗号である ECDSA（Certicom Research [2000]）を採用した。

ロ．実機検証の前提条件

（イ） 取引レスポンス時間

電子現金の送信処理にかかる時間（ターンアラウンド時間）とは、立ち上げたアプリケーションのメニュー画面から送信先を選択して送金額を入力し、送金が終了するまでの時間をいう。このうち、「送金」ボタンを押下してから、送金完了画面に推移するまでの時間は、送金元ユーザと送金先ユーザのスマートフォンでの処理時間とデータ通信にかかる時間の和（取引レスポンス時間）となる（図表 3）。具体的には、①移転履歴の更新（署名生成）にかかる時間、②電子現金の通信にかかる時間、③電子現金・移転履歴の検証（署名検証）等にかかる時間の和となる。

（ロ） 署名生成と検証にかかる処理

移転履歴の更新にかかる処理（①）では、まず、送金先ユーザの証明書を検証した後、デジタル署名の生成を行う。送信する電子現金が m 枚であるとき[39]、送付元ユーザは移転履歴の更新に伴い、m 回の署名生成処理を実行する必要がある。証明書検証と署名生成にかかる 1 回当たりの処理時間をそれぞれ t_{vs}、t_s（秒）とすれば、送信時の処理にかかる時間は $t_{vs} + t_s \cdot m$（秒）となる。また、電子現金・移転履歴の検証にかかる処理（③）では、送金元ユーザの証明書の検証と移転履歴の検証（デ

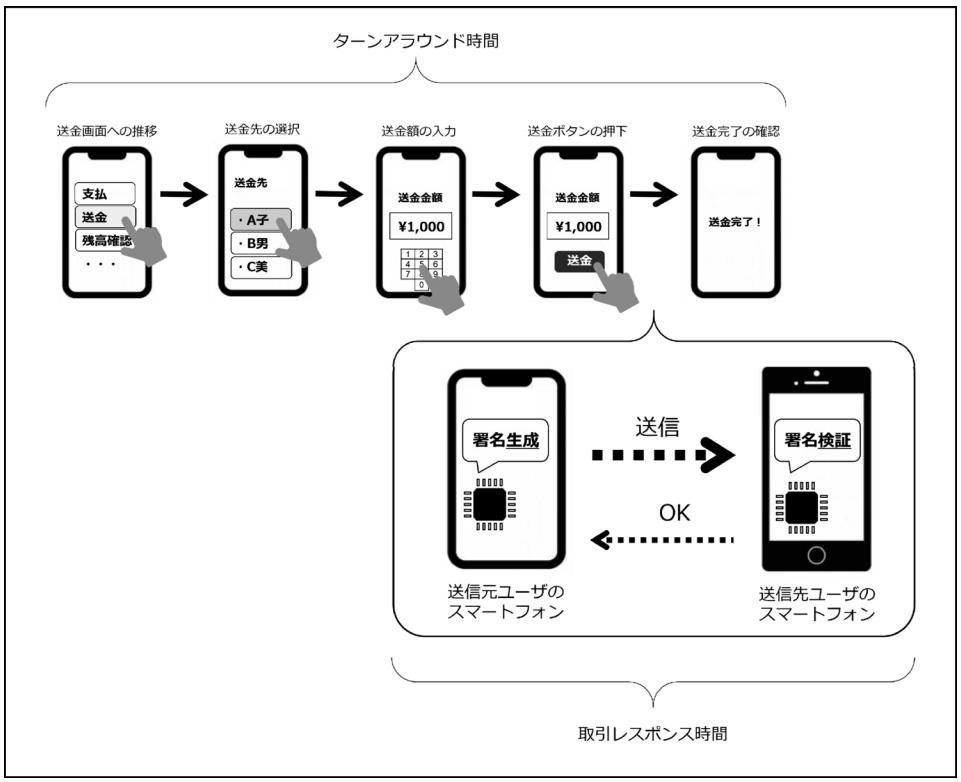
36 2024 年時点では、日本国内で販売されている Android スマートフォンの約 9 割に GP-SE が搭載されている。GP-SE とは、IC カードにかかる技術の標準化を推進する Global Platform による標準仕様に準拠した SE を指す。

37 Wi-Fi は、Wi-Fi Alliance の登録商標である。

38 Bluetooth は、Bluetooth, SIG Inc. の登録商標である。

39 送金金額に応じて、最小枚数で送金するようプログラムされていることを前提とする。

図表 3　電子現金処理にかかる操作



ジタル署名の検証）を行うため、m 枚の電子現金における発行後の移転回数がそれぞれ n_m であるとき、署名検証にかかる 1 回当たりの処理時間を t_v（秒）とすれば、受信処理にかかる時間は、$t_{vs} + t_v \cdot \sum_m (n_m + 1)$（秒）となる。このため、1 取引にかかる①と③の合計時間は、$2t_{vs} + t_s \cdot m + t_v \cdot \sum_m (n_m + 1)$（秒）となる。

電子現金方式の実装においては、秘密鍵を必要とする署名生成処理のみを SE 等の耐タンパ・デバイス上で行い、秘密鍵を必要としない署名検証処理等は TEE 等の隔離実行環境で行うといった、ハイブリッドのアーキテクチャとすることが想定される。各実行領域の処理能力を比較すると、隔離実行環境は通常の環境と同等の計算資源が利用できるのに対し、耐タンパ・デバイス（特にスマートフォンに搭載されている eSE）については、利用できる計算資源が少ない[40] ことに加え、上位レイヤからの呼出しのオーバーヘッドもかかることから、単位時間当たりの演算能力は低い。

40　2024 年時点で市場に流通している eSE のスペック・シートによれば、CPU クロック周波数は 10〜100 メガヘルツ・オーダー（シングル・コア）、RAM は 10〜100 キロバイト・オーダーであり、通常の環境や隔離実行環境と比較すると演算能力が低い。

デジタル署名については、署名検証に比べて署名生成にかかる処理量が大きいことから、耐タンパ・デバイスでは署名生成時間が性能のボトルネックとなる可能性が高い。そのため、以下では、署名生成にかかる時間（t_s）に焦点を当てて検討を行うこととする。

（ハ） 電子現金の送信にかかる処理

スマートフォンに搭載されている通信機能のうち、近距離通信であれば、災害等によって通信障害が発生したような状況であっても使用することができる。インターネットから遮断された環境でも決済が可能であること（2 節（2）性質⑦）は、電子現金に求められる性質の 1 つでもあることから、今次実機検証では、近距離無線通信技術を使用して電子現金を送信する。なお、対面決済でのユーザビリティが確認できれば、より高速なインターネット通信による決済でのユーザビリティも確保できると考えられる。

現行のスマートフォンには、一般に、Bluetooth、Wi-Fi Direct[41]、NFC が標準搭載されている（図表 4 参照）。Bluetooth は、10 メートル程度の近距離の通信規格であり、主にスマートフォンの周辺機器を無線で接続するのに使用されている。Bluetooth には、大容量のデータを高速伝送できる Bluetooth Classic と、低速・超低消費電力を特徴とする Bluetooth LE（Low Energy）がある。Wi-Fi Direct は、無線 LAN ルーターを使うことなく、Wi-Fi 機能が搭載されているパソコンやスマートフォン等の機器同士を無線で直接つなぐための規格である。また、NFC は、非接触 IC カードの通信および機器間相互通信を可能とする、通信距離 10 センチメートル程度の近距離無線通信技術である。

これらを比較すると、NFC は電子マネー決済等に広く使用されている通信技術ではあるが、Bluetooth や Wi-Fi Direct と比較して通信速度が遅い[42]。また、Bluetooth Classic については、iOS と Android で対応プロファイルが異なり、本稿執筆時点（2024 年 10 月）においてはクロスプラットフォーム[43] でのアプリケーション間通信が不可となっていることから、異なる機種のスマートフォン間でも電子現金の送受信を可能とするには、Bluetooth Low Energy と Wi-Fi Direct が候補となりうる。

41 Wi-Fi Direct は、Wi-Fi Alliance の商標または登録商標である。

42 電子現金 1 枚当たりのデータ・サイズを約 4 キロバイトとしたとき、50 枚を送付する際のデータ・サイズは約 200 キロバイト（=1.6 メガビット）となる。NFC の通信帯域は理論上最大でも 424 キロビット毎秒であることから、50 枚の送付では約 3.8 秒かかることになる。

43 異なる複数のオペレーティングシステム（Operating System: OS）上で同じ仕様のアプリケーションを動作させるプログラム。

図表 4　近距離無線通信技術の比較

	近距離無線通信技術			
	Bluetooth Classic	Bluetooth Low Energy	Wi-Fi Direct	NFC
最大通信速度規格仕様（毎秒）	3 メガビット（EDR PHY（8DPSK））	2 メガビット（LE 2M PHY）	9.6 ギガビット（Wi-Fi 6）	106キロビット（Type-A/B） 424キロビット（Type-F）
iOS[*1]・Android[*2]間接続	不可[*3]	可能	可能[*4]	可能
接続認証	必須	必須ではない	必須	必須ではない
通信データの暗号化	あり	あり（接続認証した場合）[*5]	あり	なし[*6]

備考：*1　iOS は、米国およびその他の国における Cisco Systems, Inc. の商標または登録商標である。
*2　Android は、Google LLC の商標である。
*3　2024 年 6 月時点。
*4　Android 側がグループ・オーナとなるような設計が必要。
*5　接続認証しない場合であっても、アプリケーション層で暗号化する設計は可能。
*6　アプリケーション層で暗号化する設計は可能。

図表 5　署名生成にかかる時間の比較（50 回実施した平均時間）

	eSEの種類		
	A社製	B社製（1）	B社製（2）
署名生成時間	112ミリ秒	102ミリ秒	35ミリ秒

(3)　実機検証結果

イ．署名生成にかかる時間

市場に流通している 3 種類の eSE[44] を対象に、ECDSA による 1 回のデジタル署名生成にかかる時間を評価ボードで計測したところ、A 社製は 112 ミリ秒、B 社製（1）は 102 ミリ秒、B 社製（2）は 35 ミリ秒であり、チップ・ベンダやチップ・バージョンによって数値が大きく異なることがわかった（図表 5）。最も高速に署名生成できる B 社製（2）の eSE を用いた場合、28 枚であれば、理論上 1.0 秒以内にすべての署名生成（移転履歴の更新）を実行できる計算となる。

44 検証に用いた eSE は、主要なチップ・ベンダ 2 社（A 社、B 社）の製品である（B 社製（2）は B 社製（1）の後継品）。

図表 6　電子現金の送付にかかる時間の比較（50 回実施した平均時間）

		送付サイズ（バイト）*	通信時間（ミリ秒）（接続確立の時間は含まない）	
			Bluetooth Low Energy	Wi-Fi Direct
枚数	1	3,380	590	2
	50	169,000	5,580	62
	100	338,000	10,700	160

備考：*発行されてからの移転回数を 5 と想定した場合の電子現金と移転履歴のサイズ。電子現金（サービス事業者による署名）のサイズは 580 バイトであり、移転履歴は 560 バイトであることから、合計は 580+560*5=3,380 バイトとなる。なお、実機検証に当たっては以下のようにサイズを見積もり、テスト・データを作成した。電子現金に関するデータ：580 バイト＝署名サイズ 96 バイト＋メッセージ・サイズ 484 バイト（公開鍵、シリアル・ナンバ等）、移転履歴に関するデータ：560 バイト＝署名サイズ 96 バイト＋メッセージ・サイズ 464 バイト（公開鍵等）。

一方、より多くの電子現金を一度に送付するニーズもありうる。4 節（1）で整理する方式は、署名対象となるメッセージを葉としたマークル木を作成し、そのルート値（ルート・ハッシュ）に対して署名を付与するものであり、送付枚数が何枚であっても、移転履歴の更新にかかる署名生成の回数を 1 回にすることができる。B 社製（2）の eSE の場合、送付する電子現金の枚数に関係なく、その署名生成時間は 35 ミリ秒となる。

ロ．電子現金送信にかかる時間

実機検証では、Android スマートフォンを使用して、Bluetooth Low Energy と Wi-Fi Direct それぞれについて電子現金の送付にかかる時間の測定を行った。具体的には、送信する電子現金の枚数を 1 枚、50 枚、100 枚としたときの通信時間（接続確立にかかる時間は含まない）を計測した。

測定の結果、100 枚を送信するケースでは、Bluetooth Low Energy では 10.7 秒（10,700 ミリ秒）かかったのに対し、Wi-Fi Direct ではわずか 0.16 秒（160 ミリ秒）で送付でき、Wi-Fi Direct の方が高速に電子現金を送付できることが確認できた。このため、電子現金を送付する際は Wi-Fi Direct の方が効率的であると評価できる（図表 6 参照）。

ハ．取引レスポンス時間

B 社製（2）の eSE を用いた場合、マークル木を用いた効率化を図ることで、電子現金を 100 枚送付する場合であっても、署名生成にかかる処理時間を 35 ミリ秒とすることができる。仮に、送金先における署名検証に同程度の時間がかかったとしても、送金元と送金先におけるアプリケーションの総処理時間は約 70 ミリ秒に留まる。送金元から送金先へは、Wi-Fi Direct であれば 160 ミリ秒で送信可能であることから、電子現金 100 枚の送信にかかる取引レスポンス時間は約 230 ミリ秒

（0.23 秒）と概算できる。

取引レスポンス時間については、1.0 秒以内が望ましいとする研究成果があるほか（Nielsen [2010]）、クレジットカードのタッチ決済（0.5 秒、Visa [2014]）や交通系電子マネー決済（0.2 秒、大槻［2011］）と比べても、0.23 秒であれば、十分なユーザビリティを確保可能と評価することができよう。もっとも、タッチ決済等とは異なり、電子現金の場合にはスマートフォン操作が必要であるほか、Wi-Fi Direct では都度の接続認証と通信データの暗号化が行われるため、それらにかかる時間について別途検討が必要である。

4. 電子現金方式の効率化に向けた検討

本節では、電子現金の送受信時に行われる処理と電子現金の還収における処理の効率化に向けた検討を行う。具体的には、複数枚の電子現金を送信する際に移転履歴の更新を一度にまとめて処理する方法と、還収時にサービス事業者が実施する電子現金と移転履歴の検証処理について、同じ署名者によるデジタル署名をまとめて効率的に検証する方法である。

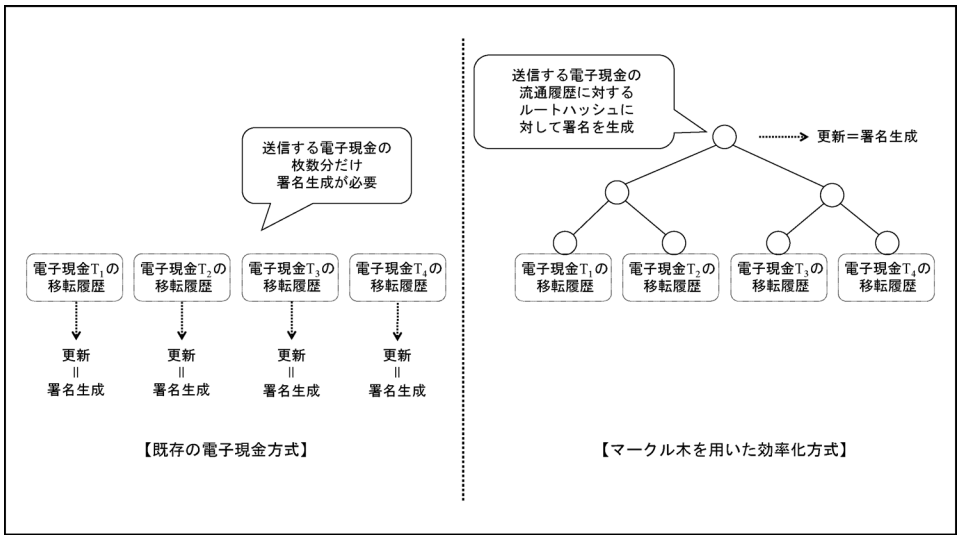
また、2 節で整理した電子現金方式では、送金額と同額の電子現金をもち合わせていない場合、釣銭を受け取る必要があった。つまり、固定額面の電子現金方式では、1 回の決済に 2 回の送金処理が必要となるケースが多い。そこで、本節では、効率化手法の 1 つとして、所有する電子現金を柔軟に分割・集約して送信することが可能な変動額面方式について考察を行う（2 節（2）性質②）。なお、変動額面方式の実現方法については、これまでにも研究されてきているが（例えば、Okamoto and Ohta [1992]）、ここでの変動額面方式は、既存方法とは異なり、デジタル署名アルゴリズム（例えば、ECDSA や EdDSA[45]）を部品として使用可能な構成となっている。そのため、任意の署名アルゴリズムを使用でき、例えば、暗号移行問題にも比較的柔軟に対応できる可能性がある[46]。

さらに、2 節の電子現金方式では、関連付け不能性を満たすために、ユーザは、取引の都度、公開鍵を更新する必要があった。本節では、こうした公開鍵の更新にかかる処理の効率化を企図して、ゼロ知識証明を利用することで、電子現金の関連付け不能性を充足する方法の提案を行う。

45 電子政府推奨暗号の 1 つ。Edward 曲線を用いたデジタル署名方式である。

46 もっとも、耐量子計算機暗号のように、署名サイズが大幅に変更される場合については、通信プロトコルにおけるメッセージ・サイズの変更等が必要となることには留意が必要である。金融分野では、耐量子計算機暗号への移行に関する検討も進んでおり（宇根［2023］）、将来の暗号移行への対応も意識した構成としておくことは重要であると考えられる。

図表 7　マークル木を用いた効率化方式



(1)　電子現金の送受信にかかる効率化

まずは、ユーザビリティ向上の観点から、電子現金の送信にかかる処理の効率化について検討を行う。2 節の電子現金方式では、送付する電子現金の枚数分だけ移転履歴の更新が必要であった。移転履歴には改ざん困難となるようにデジタル署名を付与する必要があるため、電子現金の枚数に比例して電子現金の送付にかかる処理時間が長くなる。

そこで、以下では、マークル木を用いることでデジタル署名生成を効率化するプロトコルを紹介する。マークル木とは、すべての葉ノードにデータのハッシュ値がラベル付けされ、内部ノードに、その子ノードのラベルのハッシュ値がラベル付けされている木構造をいう。マークル木を用いた効率化手法では、送付する電子現金の移転履歴からなるマークル木のルート値（ルート・ハッシュ）に対して署名を付与させる。そのため、送付枚数が何枚であっても、署名生成の回数を 1 回にすることができる（図表 7 参照）。

ユーザ U からユーザ V への送信時における処理の概要は以下のとおりである。

① ユーザ U は、送付先ユーザ V に証明書 Cer_V を要求し、その正当性を検証する。

② ユーザ U は、シリアル・ナンバを SN_i とする m 枚の電子現金 $\{T_i\}_{1\le i\le m}$ の送信に当たり、移転履歴 $\{\sigma_{i(\ell)}\}_{1\le i\le m,0\le \ell\le n_i}$ について、$h_i \leftarrow H(\sigma_{i(n_i)})\,(1 \le i \le m)$ を求める。n_i は電子現金 T_i が移転してきた回数であり、$\sigma_{i(\ell)}$ は T_i の ℓ 回目の移転履歴を表す。なお、T_i の発行時に付与される移転履歴は $\sigma_{i(0)}$ である。さらに、$\{h_i\}_{1\le i\le m}$

を葉ノードとしたマークル木 L とルート値 h を求める（マークル木の作成方法については補論 2.（1）を参照）。

③ ユーザ U は、電子現金の移転履歴を更新するに当たり、h とユーザ V の公開鍵 pk_V に対してデジタル署名 $\sigma \leftarrow Sign_{sk_U}(h \| pk_V)$ を生成する。なお、sk_U は、ユーザ U の秘密鍵である。

④ ユーザ U は、ユーザ V に移転履歴を付与した電子現金を送付する。送付するデータは、$\{SN_i, T_i, \sigma, \sigma_{i(\ell)}\}_{1 \le i \le m, 0 \le \ell \le n_i}$ と移転履歴にあるユーザの公開鍵となる。

⑤ ユーザ V は、受け取った電子現金が正しく送付されたものであることをデジタル署名の検証によって確認し、デバイス内に保存する。

これにより、電子現金を受領したユーザ V においても、ユーザ U が更新した移転履歴の検証に当たり、これまで電子現金の枚数分だけ必要であった署名検証処理を 1 回にすることができる。ただし、それ以前の移転履歴 $\sigma_{i(\ell)}(0 \le \ell \le n_i)$ については、従来と同様の検証が必要である。本プロトコルの詳細については補論 3. を参照されたい。

(2) 電子現金の還収にかかる効率化

還収時、サービス事業者は、すべての電子現金について、発行以降のすべての移転履歴が正しく生成されていることを確認する。つまり、発行以降に移転した回数が n 回であった場合、移転履歴には $n+1$ 個のデジタル署名が連鎖しているため、それらすべての検証を行う必要がある。還収される電子現金の枚数が多くなれば、サービス事業者における処理量もそれに比例して増加する。

署名方式 EdDSA では、同じ署名者による複数の署名検証式を線形結合することで署名検証時に必要となる楕円曲線上の掛け算の回数を減らすことができる（Bernstein *et al*. [2012]、藤崎［2020］）。還収した移転履歴について、金融機関や小売店といった大口ユーザの署名をまとめることができれば、移転履歴の検証にかかる処理量の削減が期待できる[47]。EdDSA の詳しいアルゴリズムについては、補論 2.（2）を参照されたい。

47 一般的なノート PC を利用して、同じ署名者による複数の署名にバッチ処理を適用した署名検証処理とバッチ処理を適用しなかった署名検証処理を比較したところ、バッチ処理を適用した場合の実行時間は、適用しなかった場合の約 45％となった（100 万回実施したときの平均値）。

(3) 変動額面方式に関する考察

2節で整理した電子現金方式は額面が設定されているため、電子現金を分割して支払うということができない。それに対して、以下では、最小単位（例えば、1円）を葉にもつ木構造で任意の金額を表すことができるようにするほか、柔軟な金額の分割も可能とする方法について考察する。具体的には、1円を葉に見立てた N 分木 ($N \geq 2$) を作成し、サービス事業者は N 分木（のルート値）へのデジタル署名として電子現金を発行する。その後、ユーザが電子現金の一部を送金したい場合には、送金したい金額に該当する部分木を作成し、その部分木（のルート値）への署名を電子現金として送金する[48]。このように、本方式では、移転履歴だけではなく、電子現金についても送受信の都度更新されることになる。なお、本節（1）で紹介したマークル木を用いた方法によって、分割された電子現金を再度集約して送信することも可能である。

以下では、$N = 2$ とした場合について、具体的な事例を用いて電子現金の発行と電子現金の送信にかかるプロトコルを紹介する[49]。

イ. 電子現金の発行

サービス事業者がユーザ U に6円分の電子現金を発行する際の手順は以下のとおりである。

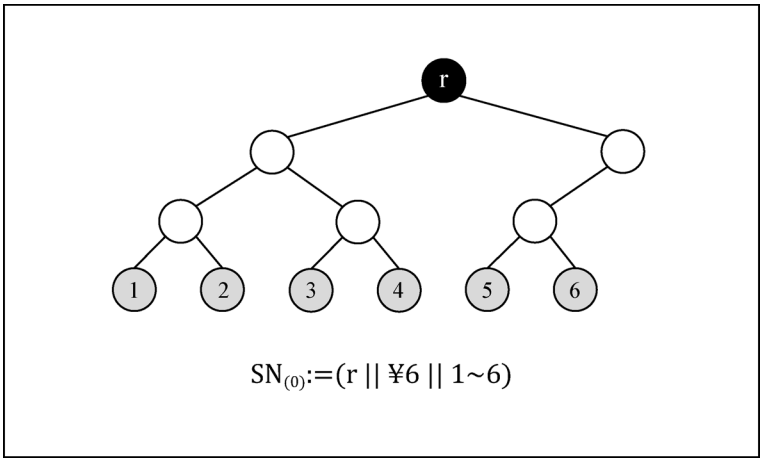
① サービス事業者は、秘密鍵と公開鍵のペア (sk_I, pk_I) を生成し、認証機関から発行された証明書とともに公開鍵を公開する。
② サービス事業者は、6円を表すために深さ $3 = (\lceil \log_2 6 \rceil)$ の2分木を生成し、葉にラベルを振る。ラベルは2分木における葉の位置を示すものであり、左から順番に $1, 2, 3, \cdots$ とする。
③ サービス事業者は、乱数 r を生成し、2分木のルートに値 r を割り当てる。
④ サービス事業者は、6円を示す電子現金のシリアル・ナンバ $SN_{(0)} := (r \| ¥6 \| 1 \sim 6)$ を生成する[50]。シリアル・ナンバ $SN_{(n)} := (r \| amount \| left \sim right)$ は、電子現金の木構造を一意に特定できるデータとなっており、ルートに割り当てた乱数 r、

48 変動額面方式を実現する方法としては、ビットコインにおけるトランザクションのように、署名のメッセージ部分に金額情報を付与することが考えられる。これに対し、本プロトコルは金額数字ではなく、1円単位の電子現金を区別できる形で送付するものであることから、二重使用された電子現金が混在してしまった場合であっても、どの電子現金が二重使用されたものかを特定することが可能という特徴をもつ。なお、実際のビットコインは台帳方式であり、二重使用チェックが速やかに実施されることから、二重使用されたビットコインが移転先で混在する可能性は低い。

49 サービス事業者による電子現金の二重使用検知プロトコルについては割愛しているが、既存方式と同様、サービス事業者は還収した電子現金をすべて保存しておき、重複がないかの確認を行う。

50 一般化する場合には、乱数 r_i に対して、シリアル・ナンバ $SN_{i(0)} := (r_i \| ¥6 \| 1 \sim 6)$ となる。

図表 8　変動額面方式における電子現金のシリアル・ナンバの作り方（2 分木で 6 円を構成する場合）



金額（*amount*）、6 円の葉の場所を指定する情報（最も左にある葉のラベル<*left*>から最も右にある葉のラベル<*right*>）で構成される（図表 8 参照）。また、*n* は発行後に電子現金がユーザ間を移転した回数である。

⑤ サービス事業者は、6 円を表す電子現金 $T_{(0)} \leftarrow Sign_{sk_I}(SN_{(0)} \| ¥6)$ とその移転履歴 $\sigma_{(0)} \leftarrow Sign_{sk_I}(SN_{(0)} \| ¥6 \| pk_U)$ を生成し、$(SN_{(0)}, T_{(0)}, \sigma_{(0)})$ をユーザ U に送信する。電子現金は、シリアル・ナンバ（$SN_{(0)}$）と金額（¥6）に対するデジタル署名であり、移転履歴は、それらに発行先ユーザの公開鍵（pk_U）を加えて生成される。

⑥ サービス事業者は、発行した電子現金（$T_{(0)}$）を発行済電子現金 DB に登録する。その際、*r* をインデックスとして $T_{(0)}$ を検索できるようにしておく。

⑦ ユーザ U は、受領した $(T_{(0)}, \sigma_{(0)})$ の正当性について、下記を検証する。

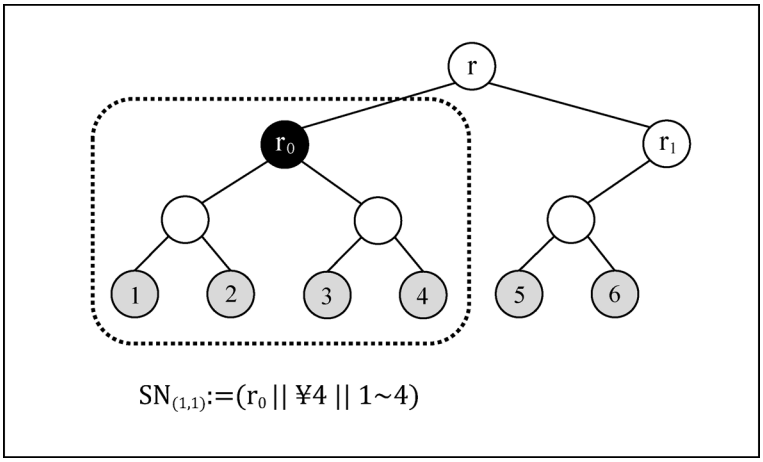
- $T_{(0)}$ と $\sigma_{(0)}$ がどちらも同じシリアル・ナンバ（$SN_{(0)}$）に対するサービス事業者の署名であること、
- $SN_{(0)}$ が発行金額に対して正しい構成となっていること、
- $\sigma_{(0)}$ が自身の公開鍵 pk_U に対するサービス事業者の署名であること。

ロ．電子現金の送信

ユーザ U がサービス事業者から発行された 6 円の電子現金のうち 4 円をユーザ V に送信する際の手順は以下のとおりである[51]。

51 6 円のうち 5 円を送る場合等、送信する金額分の葉で構成される木が木全体となるケースもある。そうした場合には、G による疑似生成乱数は不要であり、ステップ②で生成するシリアル・ナンバの第 1 要素は木のルート値 *r* となる。

図表 9　変動額面方式における電子現金の分割方法（6 円のうち 4 円を分割して送金）



① ユーザ U は、$T_{(0)}$ のシリアル・ナンバ $SN_{(0)}$ が示す 2 分木を作成し、ユーザ V に送信する 4 円分の電子現金（ラベルは 1 ~ 4）を葉とする部分木のルートに対し、$r_0 \| r_1 \leftarrow G(r)$ によって計算した r_0 を割り当てる[52]。ここで、G は m ビットから $2m$ ビットの疑似乱数を生成するシステムに共通の関数とする。

② ユーザ U は、送信する電子現金のシリアル・ナンバ $SN_{(1,1)} \leftarrow (r_0 \| ¥4 \| 1 \sim 4)$ を生成する。このとき、$SN_{(n,k)}$ は、$T_{(n-1,x)}$ $(x \geqq k)$ を分割して生成した k 個目の電子現金に付与されるシリアル・ナンバである（図表 9 参照）。さらに、電子現金と移転履歴をそれぞれ更新するため、送金先ユーザの公開鍵 pk_V に対し、$T_{(1,1)} \leftarrow Sign_{sk_U}(T_{(0)} \| SN_{(1,1)} \| ¥4)$ と $\sigma_{(1,1)} \leftarrow Sign_{sk_U}(\sigma_{(0)} \| SN_{(1,1)} \| ¥4 \| pk_V)$ を生成し、$(SN_{(1,1)}, T_{(1,1)}, \sigma_{(1,1)}, SN_{(0)}, T_{(0)}, \sigma_{(0)})$ を公開鍵 pk_U とその証明書とともにユーザ V に送信する。ユーザによる署名のメッセージ部分には、分割前の電子現金 $T_{(0)}$ と移転履歴 $\sigma_{(0)}$ がそれぞれ含まれ、移転とともに署名が連鎖する構成となっている。また、同一ユーザへの電子現金の複製対策として、移転履歴にワンタイム性を付与する、あるいは、チャレンジ・レスポンス方式を採用するものとする。

③ ユーザ U は、$T_{(0)}$ のうちラベル 5 ~ 6 が未使用である情報とともに $(T_{(0)}, \sigma_{(0)})$ をデバイスに保存する。

- 残りの 2 円のすべて、あるいはその一部を送信する際には、$(T_{(0)}, \sigma_{(0)})$ を元に送信する電子現金とその移転履歴をステップ②と同様の手順で作成する。

④ ユーザ V は、受領した $(SN_{(1,1)}, T_{(1,1)}, \sigma_{(1,1)}, SN_{(0)}, T_{(0)}, \sigma_{(0)})$ の正当性について、下

52 $G(r)$ によって生成した r_1 は、ラベル 5〜6 を葉とする部分木を送付する際に使用する。

記を確認する。

- $SN_{(1,1)}$ が正しく生成されていること、
  - ➢ $SN_{(0)}$ の第 1 要素である r に対して、$x \leftarrow G(r)$ としたとき、x の上位 m ビットが $SN_{(1,1)}$ の要素となっていること、
- $T_{(1,1)}$ と $\sigma_{(1,1)}$ が正しく生成されていること、
  - ➢ $T_{(1,1)}$ と $\sigma_{(1,1)}$ が送金元ユーザ U の署名であること、
  - ➢ $\sigma_{(1,1)}$ が自身の公開鍵 pk_V に対する署名となっていること、
- $T_{(0)}$ と $\sigma_{(0)}$ が正しく生成されていること、
  - ➢ どちらも同じシリアル・ナンバ（$SN_{(0)}$）に対するサービス事業者の署名であること、
- $\sigma_{(0)}$ のメッセージに（$\sigma_{(1,1)}$ の署名者の）公開鍵 pk_U が含まれていること。

(4) プライバシを強化した電子現金方式

2 節で整理したとおり、電子現金における関連付け不能性を充足させるには、電子現金の送受信の都度、公開鍵を変更することが必要となる。ただし、高頻度で公開鍵を更新すれば、認証機関側における証明書の発行・管理コストが膨大となるほか、それに伴う処理の遅延等によりユーザビリティも低下する。そこで、以下では、認証機関におけるユーザの証明書発行・管理コストを増加させることなくユーザのプライバシを強化可能とする方法について考察を行う[53]。ただし、本方式は、ゼロ知識証明を使用することから、電子現金の送受信にかかる処理・通信コストが相対的に大きくなる。本方式を実行するには、ユーザが使用するデバイスの性能および通信速度の向上が要件となることには留意が必要である。

プライバシを強化した電子現金方式について、プライバシ保護の関連から考慮すべき要件は以下の 3 つである。

要件①：同一ユーザが使用した電子現金の相互の関係性が特定困難であること（関連付け不能性）。

- 複数の電子現金の移転履歴に同一の公開鍵が含まれていた場合、同公開鍵に該当するユーザの取引に関する情報（例えば、購買履歴）が漏洩する可能性がある。

要件②：電子現金の匿名性と関連付け不能性が充足した状況においても、電子現金の受領者は、同電子現金取引が認証機関に登録された正当なユーザによって行われたことが確認できること。

53 本節で考察を行う方式は、本節（3）における変動額面方式には対応していないため、変動額面方式のプライバシ強化については別途検討が必要である。

- 従来は、認証機関によって発行された証明書の検証によって、同取引が正当なユーザによって行われたことが確認可能であった。プライバシを強化した方式においても、同様の性質が求められる。

要件③：電子現金の匿名性と関連付け不能性が充足した状況においても、サービス事業者は電子現金の二重使用を行ったユーザを特定できること。

- 従来の電子現金方式と同様、不正者を特定できる機能は必須である。

プライバシを強化した方式では、認証機関に登録する公開鍵とは別に取引用公開鍵を用意するほか、取引用公開鍵を取引の都度更新できるようにすることで、サービス事業者や他のユーザに対するプライバシの保護を図る（要件①）。また、認証機関から発行された証明書と電子現金取引に用いる取引用公開鍵を対応付け、それらの関係を証明することで、ユーザが認証機関に登録された正しい鍵を用いて取引を行っていることを確認できるようにする（要件②）。さらに、基本方式と同様、プライバシ保護を強化した方法においても、ゼロ知識証明の性質を利用して、二重使用された電子現金からユーザを特定できるようにする（要件③）。

本方式では、電子現金の送信元ユーザと送信先ユーザの間で、Σ プロトコルと呼ばれるゼロ知識証明（補論 3.（1）参照）を非対話形式としたプロトコルを実行する。Σ プロトコルでは、証明者が秘密にしている情報がある条件を満たしていることについて、同情報を一切明かすことなく検証者に示すことができる。本方式では、要件②を充足するため、電子現金の送信元ユーザを証明者、送金先ユーザを検証者とした Σ プロトコルを用いる。

非対話形式の場合、証明者は、コミットメント、チャレンジ、レスポンスと呼ばれる 3 種類のデータを生成して検証者に送信する。このとき、同一のコミットメントに対して、2 つの異なるチャレンジとレスポンスが存在した場合には、証明者が秘密にしている情報が露呈される性質がある。本方式では、この性質を利用して、電子現金を二重使用したユーザを特定できるようにする。具体的には、ユーザが生成するコミットメントを改ざん困難な形式でデジタル署名（電子現金の移転履歴）に埋め込ませることで、二重使用の場合であっても、コミットメントは同一となるように構成する。そのため、コミットメントについては、電子現金を送信するタイミングではなく、当該電子現金を受領するタイミングで生成させる。なお、本方式においても、プリミティブな電子現金方式と同様、電子現金 T と電子現金の移転履歴 σ がセットとして送受信される。

本方式においては、上述要件①～③を下記の方法で充足する。

- i 回目の取引に使用する取引用公開鍵 $pk_{(i)}$ の生成はユーザが行い、取引の都度、ユーザが取引用公開鍵 $pk_{(i)}$ の更新を行う。
- ユーザは、認証機関から公開鍵 pk に対する証明書 Cer の発行を受ける。また、

取引用公開鍵 $pk_{(i)}$ には秘密鍵 sk を用いて自己署名証明書 $Cer_{(i)}$ を生成する。そのうえで、証明書 Cer と自己証明書 $Cer_{(i)}$ がともに同じ公開鍵に対応するものであることの証明によって、$pk_{(i)}$ による取引が、サービス事業者に登録された正規ユーザによって行われたものであることを示す。また、ゼロ知識証明を利用することで、公開鍵 pk や証明書 Cer を秘匿できるため、サービス事業者に対する匿名性が満たされる。

- サービス事業者は、ゼロ知識証明の特性を利用することで、二重使用された電子現金から、不正を行ったユーザの公開鍵 pk を特定する。

具体的な手順の概要はイ．～ニ．のとおりである。なお、プロトコルの詳細については補論 3.（2）ハ．を参照されたい。

本方式では、二重使用を行わない限り、ユーザの公開鍵がサービス事業者に知られることはない[54]。そのため、AML/CFT の観点からは、例えば、高額取引の場合等においては、アプリケーション上で強制的に電子現金を二重使用した形をとらせることで、サービス事業者がユーザを特定できるようにする方法が考えられる（大塚［2022］）。

イ．初期設定

認証機関とユーザは暗号処理に使用する鍵ペアと証明書を以下の手順で用意する。

① 認証機関は、秘密鍵と公開鍵のペア (sk_C, pk_C) を生成し、自己証明書とともに公開する。
② ユーザ U は、鍵ペア (sk_U, pk_U) を生成し、pk_U を認証機関に送信する。
③ 認証機関は、公開鍵 pk_U に対する証明書 Cer_U をユーザ U に発行する。

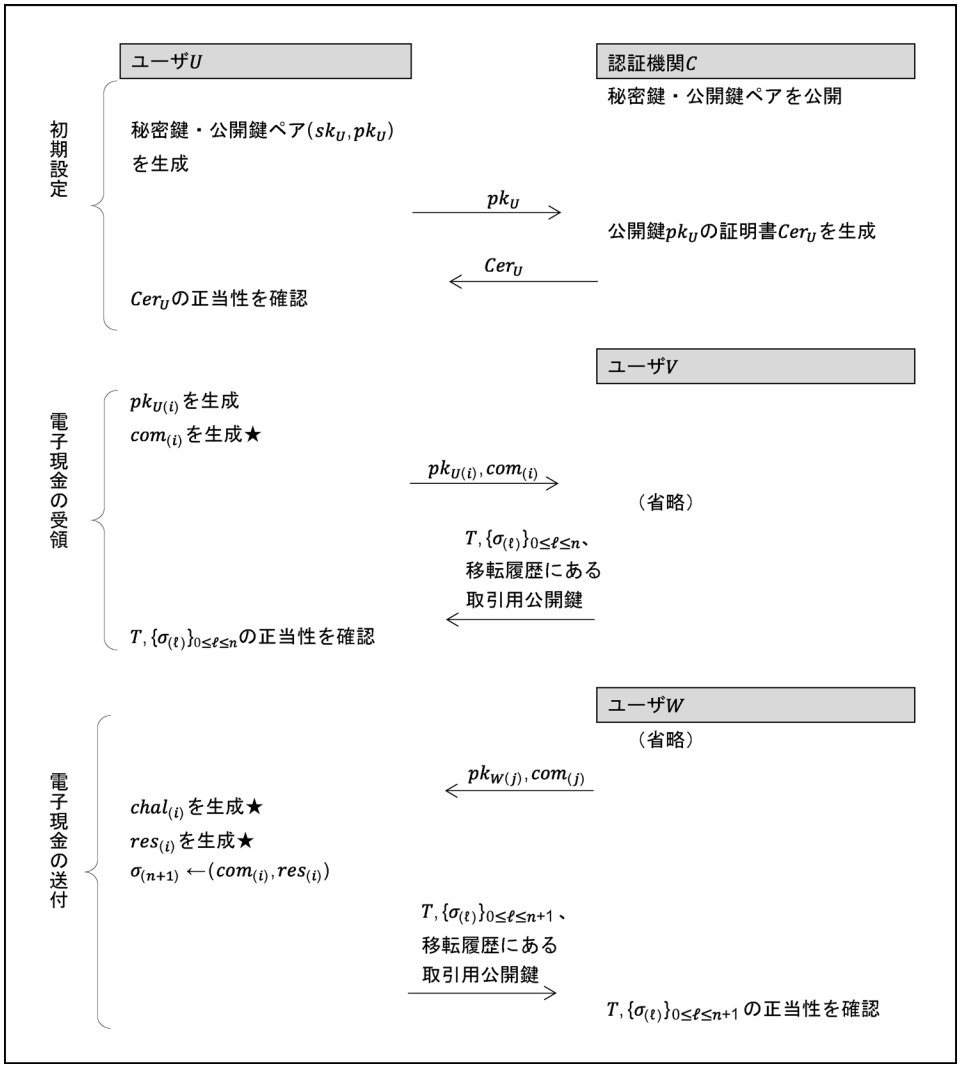
ロ．電子現金の受領

ユーザ U がユーザ V から電子現金 T を受領する際の手順は以下のとおりである。

① ユーザ U は、ランダムに取引用公開鍵 $pk_{U(i)}$ を生成する。i は、ユーザ U の取引 ID である。
② ユーザ U は、sk_U を用いて取引用公開鍵 $pk_{U(i)}$ の自己証明書 $Cer_{U(i)}$ を作成する。
　➢ 公開鍵 $pk_{U(i)}$ は、ユーザ V から受領する電子現金の取引用として生成するものである。
③ ユーザ U は、ユーザ V からの依頼を受け、電子現金 T に使用する取引用公開鍵

54 本方式では、サービス事業者であっても還収した電子現金の移転履歴からユーザの公開鍵を知ることは困難となっている。そのため、認証機関ではなく、サービス事業者がユーザ登録業務を行い、ユーザ情報を管理していたとしても、電子現金の匿名性や関連付け不能性を充足することが可能である。

図表 10　プライバシを強化した電子現金の送受信プロトコル



備考：★は、ユーザ U がユーザ V に送信する際に実行する Σ プロトコルで使用するコミットメント、チャレンジ、レスポンスである。

$pk_{U(i)}$ と、コミットメント $com_{(i)}$ を送信する[55]。

➢ ユーザ U が生成する $com_{(i)}$ は、後に電子現金をユーザ W に送信する際に実

55 x のコミットメントとは、ユーザ U からユーザ V にコミットメントを送信したとき、ユーザ V がコミットメントから x の値を知ることが困難であるほか、ユーザ U はコミットメントの送信後に x の値を変更できないように生成されたデータをいう。また、コミットメントを計算するための関数をコミットメント関数と呼ぶ。

行する Σ プロトコル[56] で用いるものであるが、電子現金の二重使用者を特定できるようにするため、このタイミングで生成してユーザ V に送信する。

④ ユーザ U は、電子現金 T とその移転履歴 $\{\sigma_{(\ell)}\}_{0\leq\ell\leq n}$、および、移転履歴にあるユーザの取引用公開鍵をユーザ V から受け取る。

➢ $\sigma_{(n)}$ は、ユーザ V による Σ プロトコル[57] でのコミットメント $com_{(k)}$ とレスポンス $res_{(k)}$ のセットである。n は、電子現金 T が移転した回数である。

⑤ ユーザ U は、T と $\{\sigma_{(\ell)}\}_{0\leq\ell\leq n}$ の正しさを確認できれば電子現金を受領し、そうでなければ拒否する。m 枚の電子現金を送付する場合には、これを m 枚分行う。

ハ. 電子現金の送付

ユーザ U がユーザ V から受領した電子現金 T をユーザ W に送付する際の手順は以下のとおりである。

① ユーザ U は、ユーザ W に対して、取引用公開鍵 $pk_{W(j)}$ とコミットメント $com_{(j)}$ を要求する。j は、ユーザ W の取引 ID である。

② ユーザ U は、$pk_{U(i)}$ を開示し、チャレンジ $chal_{(i)}$ とレスポンス $res_{(i)}$ を生成する。なお、先に生成した $com_{(i)}$ および、$(chal_{(i)}, res_{(i)})$ を用いた以下の証明により、自分が認証機関に登録した正当なユーザであること（認証機関に登録した公開鍵に対応する秘密鍵を保持すること）を示すことができる。このとき、$pk_{U(i)}$ は開示するが、自己証明書 $Cer_{U(i)}$ のほか公開鍵 pk_U や証明書 Cer_U は開示しない。

- $pk_{U(i)}$ に対する自己証明書（$Cer_{U(i)}$）を保持しているほか、自己証明書（$Cer_{U(i)}$）の生成に用いた秘密鍵に対応する公開鍵（pk_U）を保持していること、
- 上記の公開鍵（pk_U）に対して発行された証明書（Cer_U）を保持していること、
- $(pk_{W(j)}, com_{(j)}, com_{(i)}, T)$ が改ざんされていないこと。

③ ユーザ U は、$\sigma_{(n+1)} = (com_{(i)}, res_{(i)})$ を生成し、ユーザ W に $(T, \{\sigma_{(\ell)}\}_{0\leq\ell\leq n+1})$、および、移転履歴にあるユーザの取引用公開鍵を送る。

④ ユーザ W は、T と $\{\sigma_{(\ell)}\}_{0\leq\ell\leq n+1}$ の正しさを確認できれば電子現金を受領し、そうでなければ拒否する。電子現金を m 枚受領した場合には、これを m 枚分行う。

ニ. 二重使用の検知

サービス事業者は、還流してきた電子現金のシリアル・ナンバが、発行済電子現金 DB になければ、二重使用されたものと判断し、Σ プロトコルの性質を利用して、二重使用者の公開鍵を求め、認証機関に照会することで該当するユーザを特定する。

56 証明者をユーザ U、検証者をユーザ W とする Σ プロトコル。
57 証明者をユーザ V、検証者をユーザ U とする Σ プロトコル。

5. おわりに

本稿では、台帳を使用しない決済方式である電子現金に焦点を当て、改めてその特徴を整理するとともに、現在広く普及しているデバイスを前提に実機検証を行った。1990年代に実施された実証実験では、ICカードの性能や通信環境に大きな制約があったことから、ユーザビリティの確保が難しかったと想定される。これに対し、スマートフォンの使用を前提とした今次検証では、100枚の電子現金の送付であっても、理論上、既存の非接触ICカード型決済と同程度の時間で取引が可能であると示すことができた。ただし、今次検証は、取引レスポンス時間の計測を行ったものであり、実際にはアプリケーションの操作やスマートフォン同士の接続にかかる時間の考慮も必要となることには留意が必要である。もっとも、こうした評価は電子現金の送受信処理に限定したものであることから、別途検討すべき課題は少なくない。例えば、今次検討では、サービス事業者に対するプライバシ保護の観点から、ユーザの本人確認および証明書発行業務を担う独立した機関を想定したが、社会実装に向けては、現行法に照らした検討も必要である。また、AML/CFT対応についても、本稿では電子現金の事後的な追跡可能性に焦点を当てて検討を行ったが、そのほかの方法による対応可能性についても検討が望まれる。

本稿では、将来に向けた検討として、電子現金を任意の金額に分割・集約可能な方式についても考察を行った。本方式であれば、釣銭や両替にかかる取引が不要になることから、ユーザビリティの向上が期待できる。また、本稿で提案した方式は、単位金額当たりの電子現金を独立に取り扱う方式であることから、分割・集約時に金額数字を合算するだけの方法とは異なり、仮に二重使用された電子現金が混在してしまった場合でも、二重使用された電子現金の追跡が可能となっている。こうした変動額面の方式については、今後、安全性評価についてより詳細な検討を進めるほか、実機検証も行っていく予定としている。

さらに、プライバシを強化可能とするプロトコルについても考察を行った。本プロトコルでは、サービス事業者であっても、還収した電子現金からユーザの取引に関する情報を入手困難にすることができる。もっとも、ゼロ知識証明による処理は、通常のデジタル署名より計算コストや署名サイズが大きくなることから、その実装にはデバイスの性能や通信速度の向上が必須要件となる。そのため、現時点では机上での検討にとどまるが、将来に向けた検討としては有益であろう。

情報通信技術の発展は著しく、デバイスやネットワーク形態の変化に伴い、われわれの生活スタイルも変化していくことが見込まれる。今後は、実機検証を通して、電子現金の実現可能性に関する評価を継続するとともに、将来の技術進展を見据えた新たな可能性についても検討していきたい。今次検討の結果が、よりよい決

済サービス実現に向けた検討の一助となれば幸いである。

参考文献

磯部光平・宇根正志、「スマートフォン等のスマート・デバイスにおけるセキュリティ：プラットフォーム化によるリスクの現状と展望」、『金融研究』第 40 巻第 3 号、日本銀行金融研究所、2021 年、77～102 頁

宇根正志、「量子コンピュータが暗号に及ぼす影響にどう対処するか：海外における取組み」、金融研究所ディスカッション・ペーパー No. 2023-J-13、日本銀行金融研究所、2023 年

エヌ・ティ・ティ・コミュニケーションズ株式会社、「電子マネー『スーパーキャッシュ』に関する今後の NTT コミュニケーションズの取組みについて」、エヌ・ティ・ティ・コミュニケーションズ株式会社、2000 年

NTT ドコモモバイル社会研究所、「2024 年調査スマートフォン比率 97％：2010 年は約 4％」、NTT ドコモモバイル社会研究所、2024 年（https://www.moba-ken.jp/project/mobile/20240415.html、2024 年 11 月 1 日）

大塚　玲、「耐タンパ性に基づくデジタル通貨ウォレットの研究動向―匿名性と透明性の両立に向けて―」、金融研究所ディスカッション・ペーパー No. 2022-J-9、日本銀行金融研究所、2022 年

大槻知史、「Suica システムの概要」、『電気設備学会誌』第 31 巻第 6 号、2011 年、408～411 頁

岡本龍明・太田和夫、「理想的電子現金方式の一方法」、『電子情報通信学会論文誌』第 76 巻 6 号、1993 年、315～323 頁

金融調査研究会、「キャッシュレス社会の進展と金融制度のあり方」、全国銀行協会、2018 年（https://www.zenginkyo.or.jp/fileadmin/res/news/news300437_1.pdf、2024 年 11 月 1 日）

経済産業省、「2023 年のキャッシュレス決済比率を算出しました～キャッシュレス決済比率は 39.3％、2025 年の目標年に向け堅調に拡大～」、経済産業省、2024 年（https://www.meti.go.jp/press/2023/03/20240329006/20240329006.html、2024 年 11 月 1 日）

経済産業省 商務・サービスグループ消費・流通政策課、「キャッシュレス・ビジョン」、経済産業省、2018 年（https://www.meti.go.jp/policy/mono_info_service/cashless/image_pdf_movie/cl_vision.pdf、2024 年 11 月 1 日）

サイバービジネス協議会、『インターネットキャッシュ検証報告書』、E ジャパン協議会、2000 年

祖山智幸、「Suica におけるデータとサービスの在り方～オンラインとオフラインのハイブリッド～」、決済の未来フォーラムデジタル通貨分科会：ポストコロナのデジタル決済講演資料、日本銀行決済機構局、2020 年（https://www.boj.or.jp/paym/outline/mirai_forum/data/rel200911b8.pdf、2024 年 11 月 1 日）

高橋健太、「デジタル・トラスト時代の生体認証基盤～Public Biometric Infrastructure（PBI）と関連技術～」、第 22 回情報セキュリティ・シンポジウム講演資料、日本銀行金融研究所、2021 年

田村裕子・宇根正志、「IC カードを利用した本人認証システムにおけるセキュリティ対策技術とその検討課題」、『金融研究』第 26 巻別冊第 1 号、日本銀行金融研究所、2007 年、53～100 頁

中田真佐男、「対面決済のキャッシュレス化の進展に伴って検討すべき諸問題とその対応の方向性」、『国民生活研究』第 61 巻第 2 号、2021 年、32～55 頁

中山靖司・森畠秀実・阿部正幸・藤崎英一郎、「電子マネーの一実現方式について―安全性、利便性に配慮した新しい電子マネー実現方式の提案―」、『金融研究』第 16 巻第 2 号、日本銀行金融研究所、1997 年、75～86 頁

日本電信電話株式会社、「電子現金方式の実験システムを開発―暗号技術によって、利用者のプライバシ保護と高い安全性を実現―」、NTT NEWS RELEASE、日本電信電話株式会社、1995 年

――――、「新しい電子マネー実験システムを試作―安全性、信頼性、効率性を一段と高めた新方式を採用―」、NTT NEWS RELEASE、日本電信電話株式会社、1996 年

日本電信電話株式会社ネットワークサービスシステム研究所、「6G/IOWN 時代の融合・協調ネットワーク：インクルーシブコア」、日本電信電話株式会社、2023 年（https://www.rd.ntt/ns/inclusivecore/whitepaper_ver1.html、2024 年 11 月 1 日）

野田恒平、「還流する地下資金―犯罪・テロ・核開発マネーとの闘い―12 終章：デジタル革命と地下資金」、『ファイナンス』令和 4 年 7 月号、財務省、2022 年、40～50 頁

藤崎英一郎、「ディジタル署名 EdDSA の構成の安全性に関する調査および評価」、2020 年度暗号技術関連の調査報告、CRYPTREC、2020 年（https://www.cryptrec.go.jp/exreport/cryptrec-ex-3002-2020.pdf、2024 年 11 月 1 日）

古市峰子、「現金、金銭に関する法的一考察」、『金融研究』第 14 巻第 4 号、日本銀行金融研究所、1995 年、101～152 頁

北條真史・鳩貝淳一郎、「決済システムにおけるプログラマビリティの実現」、日銀レビュー No. 2022-J-12、日本銀行、2022 年

Bernstein, Daniel J., Niels Duif, Tanja Lange, Peter Schwabe, and Bo-Yin Yang, “High-Speed High-Security Signatures,” *Journal of Cryptographic Engineering*, 2012, pp. 77–89.

Certicom Research, “SEC 1: Elliptic Curve Cryptography (Version 1.0),” The Standards for Efficient Cryptography Group, 2000 (available at https://www.secg.org/SEC1-Ver-1.0.pdf、2024 年 11 月 1 日).

Chaum, David, “Blind Signatures for Untraceable Payments,” *Proceedings of CRYPTO ’82*, *Lecture Notes in Computer Science*, 1440, Springer, 1983, pp. 199–203.

Nielsen, Jakob, “Website Response Times,” Nielsen Norman Group, 2010 (available at https://www.nngroup.com/articles/website-response-times/、2024 年 11 月 1 日).

Okamoto, Tatsuaki, and Kazuo Ohta, “Universal Electronic Cash,” *Advances in Cryptology—Proceedings of CRYPTO’91*, *Lecture Notes in Computer Science*, 576, Springer, 1992, pp. 324–337.

Visa, “Contactless Payments,” VISA Canada, 2014 (available at https://www.visa.ca/dam/VCOM/regional/na/canada/merchants/documents/visa-paywave-put-your-customer-in-the-fast-lane-en.pdf、2024 年 11 月 1 日).

補論 1. 認証機関の機能を分割した場合の証明書発行手順

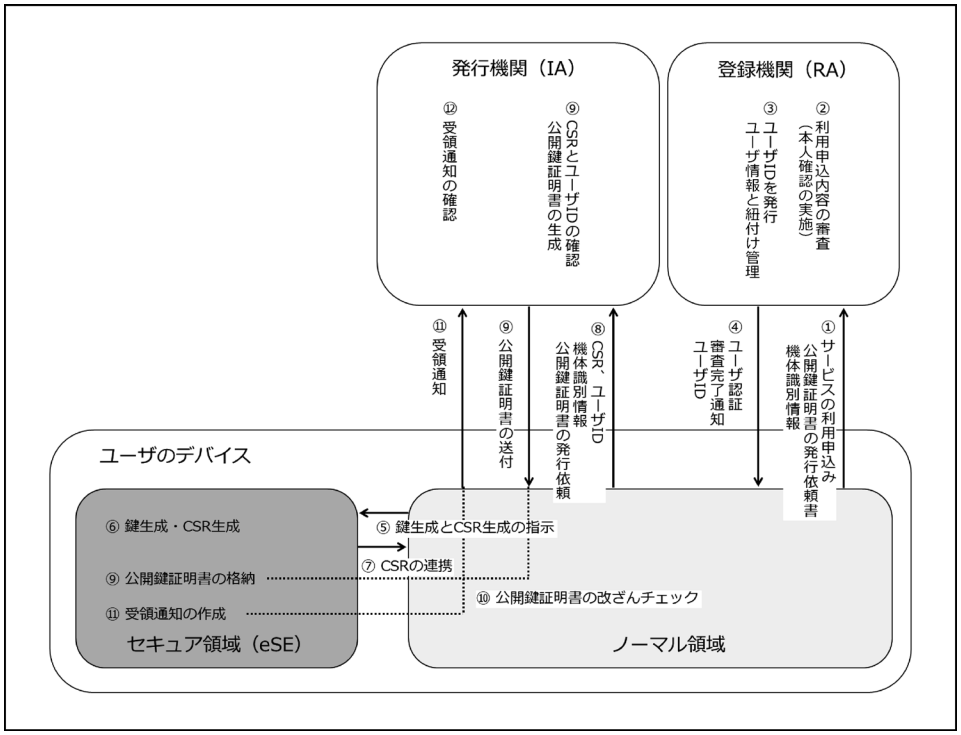
2 節（3）におけるユーザ登録と証明書の発行において、2 つの業務を独立させた場合の手順について整理する（図表 A-1）。ここでは、ユーザ情報の管理を行う機関を登録機関（Registration Authority: RA）、証明書の発行・管理を行う機関を発行機関（Issuing Authority: IA）、この 2 つを合わせたものを認証機関と呼ぶこととする。

① ユーザは、デバイスのノーマル領域にインストールされたアプリケーション（以下、ノーマル領域アプリ）を介して、RA にサービスの利用申込みを行うとともに、証明書の発行依頼書を送付する。その際、RA は、ユーザの機体識別情報[58] をユーザのデバイスから要求する。
② RA は、送付された利用申込書の内容について審査を行う[59]。その際、RA は必要に応じて、本人確認を実施する。
③ RA は、審査の結果に問題がなければ、ノーマル領域アプリに利用者識別符号を発行し、ユーザが提出したユーザ情報と紐付けて管理する。
④ RA は、審査完了通知と利用者識別符号をノーマル領域アプリに送付する。
⑤ ノーマル領域アプリは、審査完了通知を受領した後、セキュア領域にインストールされているアプリケーション（以下、セキュア領域アプリ）に対して、鍵ペアの生成、および、証明書の発行リクエスト（Certificate Signing Request: CSR）の生成を指示する。
⑥ セキュア領域アプリは、鍵ペアの生成、および、CSR の生成を行う。
⑦ セキュア領域アプリは、セキュア領域内で生成された CSR をノーマル領域アプリへ連携する。
⑧ ノーマル領域アプリは、CSR、利用者識別符号、機体識別情報を IA に送信し、証明書の発行を依頼する。
⑨ IA は、受領した CSR と利用者識別符号に問題がないこと（重複がないこと等）を確認のうえ、証明書を生成する。
⑩ セキュア領域アプリは、公開鍵の改ざんチェックを行い、セキュア領域に証明書を格納する。
⑪ セキュア領域アプリは、受領通知をセキュア領域内で作成し、ノーマル領域アプリを介して IA に送付する。
⑫ IA は受領通知を確認する。

58 機体識別情報として使用できる情報は、国際移動体装置識別番号（International Mobile Equipment Identity: IMEI）や各デバイスの eSE に割り振られた ID 等が考えられ、サービスに求められるセキュリティ・ユーザビリティの観点から選択する。

59 サービスを利用させることについて問題がないかを審査。例えば、過去に利用規約違反をしたユーザでないことの確認等が想定される。

図表 A-1　認証機関の機能を分割した場合の証明書発行手順



補論 2. 電子現金方式の効率化

(1) 電子現金の送受信にかかる効率化

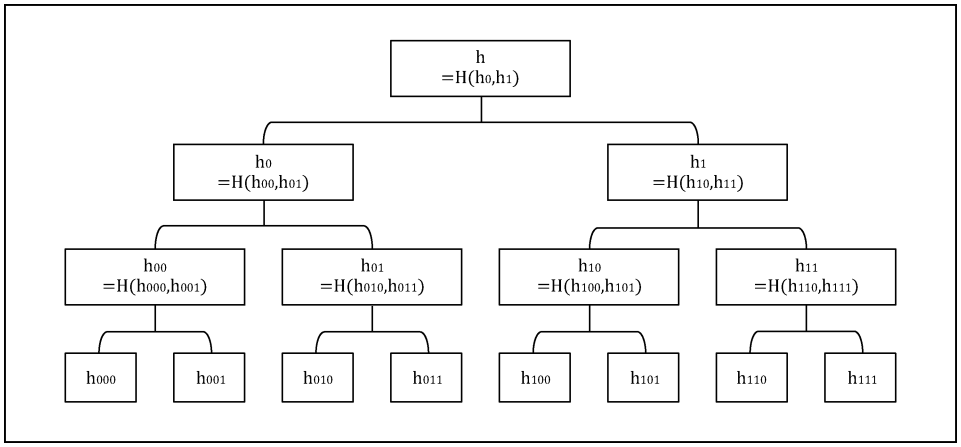
本節では、4 節（1）で整理した電子現金の送受信にかかる効率化について、それを実現するための詳細なプロトコルを述べる。

イ. 初期設定（3 つのアルゴリズム）

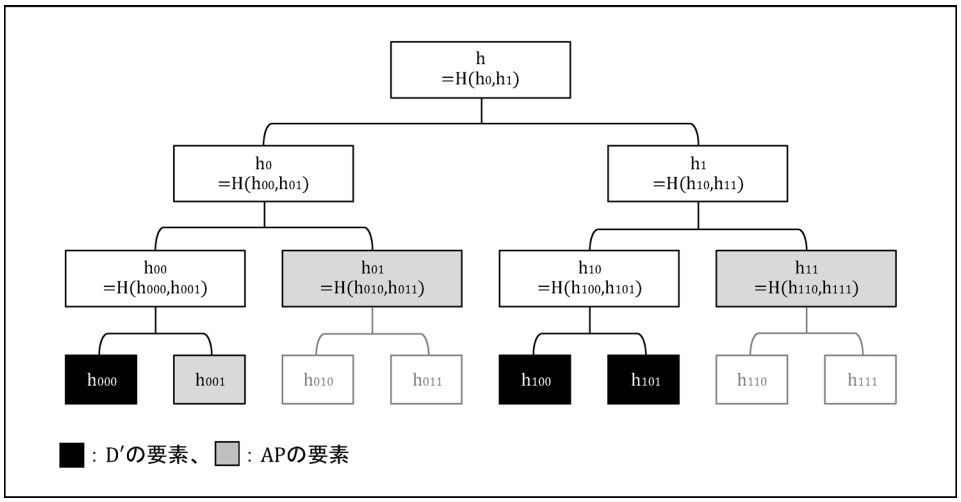
マークル木（図表 A-2）を用いた方式では、以下の 3 つのアルゴリズム MHTree、MHPath、MHVer を使用する。

- $(L, h) \leftarrow \mathrm{MHTree}(D)$
  - ➢ 関数 MHTree は、データセット D の入力に対し、D の要素（$=\{(i, h_i)\}$）を葉ノードとしたマークル木 L とルート値 h を出力する。マークル木とは、2 つの子ノードのハッシュ値を親ノードの値として割り当てる処理を繰り返して生成された木構造である。i はノードの識別子、h_i はノードにラベル付けされた値を示す。例えば、深さ 3 の葉ノードの識別子 i は 3 桁で表され、左から順番に $000, 001, \ldots, 111$ となる。なお、D の要素数が二分木の生成に足りない時は、$h_i =$ “00…” とラベル付けしたノードを補完したマークル木を出力する。そのうえで、H をハッシュ関数とするとき、h_{00} に $H(h_{000}, h_{001})$ を割り当て、h_0 に $H(h_{00}, h_{01})$ を割り当てる作業を繰り返す。図表 A-2 の例では、識別子を i とするノードの値を h_i と表している。
- $(AP, h) \leftarrow \mathrm{MHPath}(D, D')$
  - ➢ MHPath は、データセット D と D の部分集合 $D'(\subset D)$ の入力に対し、D から生成されたマークル木のルート h を D' から計算するうえで必要となるパス AP とルート値 h を出力する。
  - ➢ 例えば、$D' = \{(000, h_{000}), (100, h_{100}), (101, h_{101})\}$ であったとき、MHPath は D' の入力に対して、$AP = \{(001, h_{001}), (01, h_{01}), (11, h_{11})\}$ と h を出力する（図表 A-3 参照）。
- $1/0 \leftarrow \mathrm{MHVer}(D', AP, h)$
  - ➢ MHVer は、データセット D' と AP、および、ルート値 h の入力に対し、$D' \cup \mathrm{AP}$ をノードとしたマークル木のルート値が h と一致するか否かを確認し、一致すれば 1、一致しなければ 0 を返す。

図表 A-2　マークル木の構造



図表 A-3　関数 MHPath の入力（D’）と出力（AP）の関係



ロ．　電子現金の送受信にかかる処理

ユーザ U がユーザ V に電子現金を送付する際の具体的な手順は以下のとおりである。

① ユーザ U は、送付先ユーザ V に証明書 Cer_V を要求し、その正当性を検証する。

② ユーザ U は、送信する m 枚の電子現金 $T_i(1 \le i \le m)$ とその移転履歴 $\sigma_{i(\ell)}(1 \le i \le m, 0 \le \ell \le n_i)$ について、$h_i \leftarrow H(\sigma_{i(n_i)})(1 \le i \le m)$ を求める。n_i は電子現金 T_i が移転してきた回数であり、$\sigma_{i(\ell)}$ は T_i の ℓ 回目の移転履歴を表す。なお、T_i の発行時に付与される移転履歴は $\sigma_{i(0)}$ である。さらに、$\{h_i\}_{1 \le i \le m}$

を葉ノードとしたマークル木 L とルート h を $(L,h) \leftarrow \mathrm{MHTree}(\{(i,h_i)\}_{1\le i\le m})$ によって求める。

③ ユーザ U は、ルート h とユーザ V の公開鍵 pk_V に対するデジタル署名 $\sigma \leftarrow Sign_{sk_U}(h \| pk_V)$ を生成することで移転履歴を更新する。

④ ユーザ U は、ユーザ V に自身の証明書 Cer_U および m 枚の電子現金とその移転履歴を送付する。このとき、送付する m 枚の電子現金とその移転履歴は $(\{T_i\}_{1\le i\le m}, \sigma, \{\sigma_{i(\ell)}\}_{1\le i\le m, 0\le \ell\le n_i})$ となる。さらに、これまでの移転履歴 $\sigma_{i(\ell)}(0 \le \ell \le n_i)$ の検証に必要となる $(AP_j, h_j) \leftarrow \mathrm{MHPath}(\{\sigma_j\}, \sigma_{i(n_i-1)})$ もあわせて送信する。$\{\sigma_j\}$ は、過去に電子現金 T_i をユーザ U_j から受け取った際、それと同時に送られてきた電子現金 $\{T_j\}(\ni T_i)$ の移転履歴を表す。

➢ 更新された移転履歴 σ は、m 枚の電子現金 T_i に共通の移転履歴であり、$\sigma = \sigma_{1(n_1)} = \sigma_{2(n_2)} = \cdots = \sigma_{m(n_m)}$ である。

⑤ ユーザ V は、証明書 Cer_U の正当性確認を行ったうえで、受け取った電子現金が正しく送付されたものであることを以下によって確認し、デバイス内に保存する。

- $(L,h) \leftarrow \mathrm{MHTree}(\{(i,\sigma_{i(n_i)})\}_{1\le i\le m})$ を計算し、σ が h に対するユーザ U の署名であることを確認する。
- $\mathrm{MHVer}(\sigma_{i(n_i)}, AP_j, h_j)$ によって h_j の正当性を確認したうえで、$\sigma_{i(n_i)}$ が h_j に対する正当な署名になっていることを確認する。これをすべての移転履歴 $\sigma_{i(\ell)}(1 \le i \le m, 0 \le \ell \le n_i)$ に対して行う。

(2) 電子現金の還収にかかる効率化

EdDSA は、以下のパラメータ設定のもと、3 つのアルゴリズム（鍵生成、署名生成、署名検証）で構成される（Bernstein *et al.* [2012]、藤崎［2020］）。

【パラメータ設定】

EdDSA で使用されるパラメータは以下のとおりである。

- 奇素数 q、有限体 $\mathbb{F}_q$
- 正整数 b：公開鍵長。なお、$q<2^{b-1}$ を満たす。
- ハッシュ関数 H: $\{0,1\}^* \to \{0,1\}^{2b}$
- エンコーディング関数 : $\mathbb{F}_q \to \{0,1\}^{b-1}$（$\mathbb{F}_q$ の元を $(b-1)$ bit に変換）
- $\mathbb{F}_q$ 上の（Twisted）Edward 曲線
  $E(\mathbb{F}_q)$: $\{(x,y) \in \mathbb{F}_q \times \mathbb{F}_q$: $ax^2+y^2 = 1+dx^2y^2\}$.
  a：$\mathbb{F}_q$ 上の平方剰余、d：$\mathbb{F}_q$ 上の (非ゼロの) 平方非剰余。

なお、$q \equiv 1\,(mod\ 4)$ の時は $a = -1$、$q \equiv 3\,(mod\ 4)$ の時は $a = 1$。
E の位数 $\#E = 2^c * \ell,\ c = 2\ or\ 3,\ 2^c$ を cofactor と呼ぶ。

- ベース・ポイント $B \in E, B \neq (0,1), \ell B = (0,1)$、
  奇素数 ℓ, B の作る巡回群の位数 $\#B = \ell$。

エドワード曲線の加法を下記で定義するとき、E は、$O = (0,1)$ を単位元とする加法群になる。

$$(x_1, y_1) + (x_2, y_2) := \left(\frac{(x_1 y_2 + x_2 y_1)}{1 + d x_1 x_2 y_1 y_2}, \frac{(y_1 y_2 + x_1 x_2)}{1 - d x_1 x_2 y_1 y_2} \right). \tag{A-1}$$

【鍵生成】

ユーザは、秘密鍵 k と公開鍵 A を下記手順で生成する：

- $k \leftarrow \{0,1\}^b$
- $(h_0, h_1, \ldots, h_{2b-1}) \leftarrow H(k)$
- $a \leftarrow 2^{2b-2} + \sum_{3 \le i \le b-3} 2^i h_i \in \{2^{2b-2}, 2^{2b-2}+8, \ldots, 2^{2b-1}-8\}$
- $A \leftarrow aB$

秘密鍵：k
公開鍵：$\underline{A} := aB$（$\underline{A}$は、A をエンコーディングした値）

【署名生成】

署名者は、秘密鍵 k から求めた $(h_b, \ldots, h_{2b-1})\ (=H(k))$ を用いて、メッセージ M に対する署名 $(\underline{R}, \underline{S})$ を以下のとおり生成する：

- $r \leftarrow H(h_b, \ldots, h_{2b-1}, M) \in \{0, 1, \ldots, 2^{2b}-1\}$.
- $R \leftarrow rB$.
- $S \leftarrow (r + H(\underline{R}, \underline{A}, M)\ a) \bmod \ell$.

【署名検証】

検証者は、署名者の公開鍵 $\underline{A}$を用いて、メッセージ M に対する署名 $(\underline{R}, \underline{S})$ の正当性を以下の等式の成立によって確認する：

- $2^c S B = 2^c R + 2^c H(\underline{R}, \underline{A}, M) A$.

【バッチ検証】

公開鍵 A_i によるメッセージ M_i に対する署名を $(\underline{R_i}, \underline{S_i})$ としたとき、これらのバッチ処理は、署名検証式を線形結合することで実行できる。

- 各パラメータと署名数に応じたビットのランダムな整数 z_i を選択し、$H_i=H(R_i, A_i, M_i)$ を計算する。
- $(-\Sigma_i z_i S_i \bmod \ell) B+\Sigma_i z_i R_i+\Sigma_i (z_i H_i \bmod \ell) A_i= 0$ の成立を確認する。

補論 3. プライバシを強化した電子現金プロトコル

ここでは、4 節（4）で提案したプロトコルの詳細に加えて、サービス事業者にユーザの秘密鍵を登録し、不正者の特定はユーザの秘密情報によって行う方式（方式 1）と、サービス事業者にはユーザの公開鍵のみを登録するが、不正者の特定はユーザの秘密情報によって行う方式（方式 2）もあわせて紹介する。これらは、4 節で紹介した方式より運用面での安全性は劣後するが、計算処理コストが低いというメリットがある。

(1) Σ プロトコル

Σ プロトコルは、言語 $L_R = \{x \mid \exists w, (x, w) \in R\}$ に属する x のウィットネス w（x が言語 L に属することの証拠）を知っていることを示すゼロ知識証明プロトコルであり、証明者と検証者間で、「コミットメント」、「チャレンジ」、「レスポンス」と呼ばれるデータを送受信することで実行される。

証明者から検証者に対して最初に送信するコミットメント com は、公開情報 x、ウィットネス w、乱数 r を用いて生成される。次に検証者から証明者に送信されるチャレンジ $chal$ は十分に安全な乱数を選択する関数を用いて生成される。最後に証明者から検証者に送信するレスポンス res は、x のウィットネス w を知っているユーザであれば、それまでに送受信された com と $chal$ に対して、$1 \leftarrow VER(com, chal, res, x)$ となる res を生成することができる関数 RES を用いて生成される。また、x のウィットネス w を知らないユーザが $1 \leftarrow VER(com, chal, res, x)$ となる res を生成できる確率は無視できるほど小さく、圧倒的な確率で $0 \leftarrow VER(com, chal, res, x)$ となる（図表 A-4 参照）。

また、$EXT(x, com, chal, chal', res, res')$ は、2 つのゼロ知識証明において、同じコミットメントを使用した際の $(chal, res)$ と $(chal', res')$ の入力に対して、$com \leftarrow COM(x, w; r)$ をみたす w を返す。

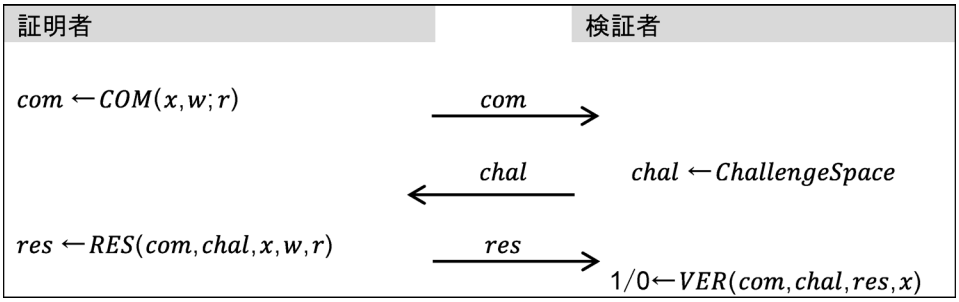
上記 Σ プロトコルは、ハッシュ関数 H を用いることで、非対話証明とすることができる。つまり、$chal \leftarrow H(com)$ と置き換えることで、証明者は検証者と対話することなく res を生成することができる。

プライバシ保護を強化した電子署名方式は、以下の関数で構成される。

- $(sk, pk) \leftarrow Gen(1^{\delta})$

  Gen は、秘密鍵と公開鍵のペア (sk, pk) を生成する関数。δ はセキュリティ・パラ

図表 A-4　Σ プロトコル



メータ。

- $c \leftarrow Commit(x; r)$
  Commit は、乱数 r を用いて x のコミットメント c を生成する関数。
- $\sigma \leftarrow Sign_{sk}(m)$
  Sign は、秘密鍵 sk を用いて、m のデジタル署名 σ を生成する関数。
- $1/0 \leftarrow Verify_{pk}(m, \sigma)$
  Verify は、公開鍵 pk を用いて、メッセージ m に対するデジタル署名 σ の検証を行う関数。σ が正しい署名であれば 1、そうでなければ 0 を出力する。
- $com \leftarrow COM(x, w; r)$
  COM は、x のウィットネス w の保持を証明する Σ プロトコルにおけるコミットメントを生成する関数。r は乱数。
- $chal \leftarrow H(com)$
  H は、ハッシュ関数であり、Σ プロトコルにおけるチャレンジ生成に使用。
- $res \leftarrow RES(com, chal, x, w, r)$
  RES は、x のウィットネス w の保持を証明する Σ プロトコルにおけるレスポンスを生成する関数。
- $1/0 \leftarrow VER(com, chal, res, x)$
  VER は、com と res が、x のウィットネス w の保持を証明する Σ プロトコルでのペアとなっていることの検証を行う関数。(com, res) が正しければ 1、そうでなければ 0 を出力する。
- $w \leftarrow EXT(x, com, chal, chal', res, res')$
  EXT は、2 つのゼロ知識証明において、同じコミットメント com を使用した $(chal, res)$ と $(chal', res')$ の入力に対して、$com \leftarrow COM(x, w; r)$ をみたす w を出力する。

(2) 3つの実現方法

イ. 方式1（秘密鍵を認証機関に登録、二重使用者の特定は秘密鍵で行う）

方式1は、公開鍵 pk ではなく、ユーザの秘密鍵 sk を認証機関に登録し、認証機関から秘密鍵 sk に対する証明書 Cer の発行を受ける方式である。4節（4）で整理した3つの要件については、下記の方法で充足される。

- i 回目の取引に使用する取引用公開鍵 $pk_{(i)}$ の生成はユーザが行い、取引の都度、ユーザが取引用公開鍵 $pk_{(i)}$ の更新を行う。なお、取引用公開鍵 $pk_{(i)}$ は、秘密鍵 sk をもとに生成される。
- ユーザは、認証機関から秘密鍵 sk に対する証明書 Cer の発行を受ける。証明書 Cer と取引用公開鍵 $pk_{(i)}$ がともに同じ秘密鍵に基づくものであることの証明によって、$pk_{(i)}$ による取引が、認証機関に登録された正規ユーザによって行われたものであることを示す。また、ゼロ知識証明を利用することで、秘密鍵 sk や証明書 Cer を秘匿できるため、サービス事業者に対する匿名性が満たされる。
- サービス事業者は、ゼロ知識証明の特性を利用することで、二重使用された電子現金から、不正を行ったユーザの秘密鍵 sk を特定する。

（イ）初期設定

認証機関とユーザは暗号処理に使用する鍵ペアと証明書を以下の手順で用意する。

① 認証機関は、秘密鍵と公開鍵のペア $(sk_C, pk_C) \leftarrow Gen(1^\delta)$ を生成し、公開鍵を自己証明書とともに公開する。
② 認証機関は、ユーザ U の秘密鍵 $sk_U \in \{0,1\}^\lambda$ を生成し、$Cer_U \leftarrow Sign_{sk_C}(sk_U)$ とともにセキュアチャネルを通してユーザ U に発行する。なお、λ はセキュリティ・パラメータである。
③ ユーザ U は、$Verify_{pk_C}(sk_U, Cer_U)$ を計算し、結果が1であれば (sk_U, Cer_U) を受領し、そうでなければ拒否する。

（ロ）電子現金の受領

ユーザ U がユーザ V から電子現金 T を受領する際の手順は以下のとおりである。

① ユーザ U は、乱数 $r_{U(i)} \in \{0,1\}^\lambda$ を選択し、取引用公開鍵 $pk_{U(i)} \leftarrow Commit(sk_U ; r_{U(i)})$ を計算する。

- $pk_{U(i)}$ は、ユーザ U にとって、i 回目の取引に使用する公開鍵である。

② ユーザ U は、乱数 $r_{(i)} \in \{0,1\}^{\lambda}$ を選択し、$com_{(i)} \leftarrow COM((pk_{U(i)}, pk_C), (sk_U, Cer_U, r_{U(i)}); r_{(i)})$ を計算する。

③ ユーザ U は、ユーザ V に $(pk_{U(i)}, com_{(i)})$ を送信し、ユーザ V から $(T, \{\sigma_{(n)} = (com_{(k)}, res_{(k)})\}_{n\geq 0})$ と移転履歴にあるユーザの取引用公開鍵を受領する。T は、ユーザ V から送信された電子現金であり、シリアル・ナンバに対するサービス事業者の署名である。$\sigma_{(n)}$ は電子現金の移転履歴となる（n は、電子現金 T がこれまでに移転した回数）。

- $com_{(k)}$ は、ユーザ V が公開鍵 $pk_{V(k)}$ のもとに他ユーザから電子現金 T を受け取った際に生成したもの（本プロトコルのステップ②でユーザ U が行う計算に相当）。これは、ユーザ V にとって k 回目の取引となる。
- $res_{(k)}$ は、ユーザ V がユーザ U から $(pk_{U(i)}, com_{(i)})$ の送付を受けて生成したもの（電子現金の送付プロトコルのステップ③でユーザ U が行う計算に相当）。

④ ユーザ U は T の検証を行うとともに、$\{\sigma_{(n)} = (com_{(k)}, res_{(k)})\}_{n\geq 0}$ について $VER(com_{(k)}, chal_{(k)}, res_{(k)}, (pk_{V(k)}, pk_C))$ を計算し、すべての結果が 1 であれば電子現金を受領し、そうでなければ拒否する。なお、$chal_{(k)} \leftarrow H(com_{(k)})$ は $com_{(k)}$ のハッシュ値である。

（ハ）電子現金の送付

ユーザ U がユーザ W に電子現金 T を送付する際の手順は以下のとおりである。

① ユーザ U は、ユーザ W から $(pk_{W(j)}, com_{(j)})$ を受領する。

- ユーザ W にとっては、電子現金の受領プロトコルのステップ①～②に相当するものであり、j 回目の取引となる。

② ユーザ U は、$chal_{(i)} \leftarrow H(T, pk_{W(j)}, com_{(j)}, com_{(i)})$ を計算する。

- $com_{(i)}$ は、ユーザ U がユーザ V から電子現金 T を受け取った際に生成したもの（電子現金の受領プロトコル：ステップ②）

③ ユーザ U は、$res_{(i)} \leftarrow RES(com_{(i)}, chal_{(i)}, (pk_{U(i)}, pk_C), (sk_U, Cer_U, r_{U(i)}), r_{(i)})$ を計算する。なお、Σ プロトコルは、$(pk_{U(i)}, pk_C)$ が以下の言語 L に属することを証明するものである。$L := \{(pk_{U(i)}, pk_C) | \exists (sk_U, Cer_U, r_{U(i)}), com(sk_U; r_{U(i)}) = pk_{U(i)} \wedge Verify_{pk_C}(sk_U, Cer_U) = 1\}$.

④ ユーザ U は、ユーザ W に $(T, \{\sigma_{(n+1)} = (com_{(i)}, res_{(i)})\}_{n\geq 0})$ と移転履歴にあるユーザの取引用公開鍵を送信する。

⑤ ユーザ W は、T の検証を行うとともに、$\{\sigma_{(n+1)} = (com_{(i)}, res_{(i)})\}_{n\geq 0}$ について $VER(com_{(i)}, chal_{(i)}, res_{(i)}, (pk_{U(i)}, pk_C))$ を計算し、すべての結果が 1 であれば電子

現金を受領し、そうでなければ拒否する。なお、$chal_{(i)} \leftarrow H(com_{(i)})$ は $com_{(i)}$ のハッシュ値である。

（ニ）二重使用の検知

サービス事業者が電子現金を二重使用したユーザを特定する際の手順は以下のとおりである。

① サービス事業者は、還流してきた電子現金 T のシリアル・ナンバが発行済電子現金 DB になければ、二重使用されたものと判断し、還収済電子現金 DB から同シリアル・ナンバをもつ電子現金とその履歴 σ' を取り出す。
② サービス事業者は、2 つの移転履歴 σ と σ' について、秘密鍵 $sk_U \leftarrow EXT(pk_{U(i)}, com, chal, chal', res, res')$ を計算する。
③ サービス事業者は、認証機関に照会することで、sk_U を秘密鍵とするユーザを特定する。

ロ．方式 2（公開鍵を認証機関に登録、二重使用者の特定は秘密鍵で行う）

方式 2 は、ユーザの秘密鍵ではなく、ユーザの公開鍵に対して証明書の発行を受ける方式である。認証機関・ユーザ間における秘密鍵の送受信がないため、鍵の漏洩リスクが低減するものの、電子現金の送受信にかかる計算コストは方式 1 より増加する。4 節（4）で整理した 3 つの要件については、下記の方法で充足される。

- i 回目の取引に使用する取引用公開鍵 $pk_{(i)}$ の生成はユーザが行い、取引の都度、ユーザが取引用公開鍵 $pk_{(i)}$ の更新を行う。
- ユーザは、認証機関から公開鍵 pk に対する証明書 Cer の発行を受ける。証明書の対象となる公開鍵 pk と取引用公開鍵 $pk_{(i)}$ がともに同じ秘密鍵で生成されたものであることの証明によって、$pk_{(i)}$ による取引が認証機関に登録された正規ユーザによって行われたものであることを示す。
  - ➢ 方式 1 との違いは、認証機関による証明書が公開鍵に対して発行されることである。それに伴い、Σ プロトコルで証明すべき命題が変わるため、コミットメントとレスポンスの生成方法が変わる。
- サービス事業者は、ゼロ知識証明の特性を利用することで、二重使用された電子現金から、不正を行ったユーザの秘密鍵 sk を特定する。

（イ）初期設定

認証機関とユーザは暗号処理に使用する鍵ペアと証明書を以下の手順で用意する。

① 認証機関は、秘密鍵と公開鍵のペア $(sk_C, pk_C) \leftarrow Gen(1^\delta)$ を生成し、公開鍵を自己証明書とともに公開する。
② ユーザ U は、秘密鍵 $sk_U \in \{0,1\}^\lambda$ を生成する。
③ ユーザ U は、乱数 r_U を選択し、公開鍵 $pk_U \leftarrow Commit(sk_U ; r_U)$ を計算する。
④ ユーザ U は、サービス事業者に pk_U を送付し、pk_U に対する証明書 $Cer_U \leftarrow Sign_{sk_C}(pk_U \| *)$ を受け取る。
⑤ ユーザ U は、$Verify_{pk_C}(pk_U, Cer_U)$ を計算し、結果が 1 であれば (pk_U, Cer_U) を受領し、そうでなければ拒否する。

（ロ）電子現金の受領

ユーザ U がユーザ V から電子現金 T を受領する際の手順は以下のとおりである。

① ユーザ U は、乱数 $r_{U(i)} \in \{0,1\}^\lambda$ を選択し、取引用公開鍵 $pk_{U(i)} \leftarrow Commit(sk_U ; r_{U(i)})$ を計算する。
② ユーザ U は、乱数 $r_{(i)} \in \{0,1\}^\lambda$ を選択し、$com_{(i)} \leftarrow COM((pk_{U(i)}, pk_C), (pk_U, sk_U, Cer_U, r_{U(i)}, r_U) ; r_{(i)})$ を計算する。
③ ユーザ U は、ユーザ V に $(pk_{U(i)}, com_{(i)})$ を送信し、ユーザ V から $(T, \{\sigma_{(n)} = (com_{(k)}, res_{(k)})\}_{n \geq 0})$ と移転履歴にあるユーザの取引用公開鍵を受領する。
④ ユーザ U は、T の検証を行うとともに、$\{\sigma_{(n)} = (com_{(k)}, res_{(k)})\}_{n \geq 0}$ について、$VER(com_{(k)}, chal_{(k)}, res_{(k)}, (pk_{V(k)}, pk_C))$ を計算し、すべての結果が 1 であれば電子現金を受領し、そうでなければ拒否する。なお、$chal_{(k)} \leftarrow H(com_{(k)})$ は $com_{(k)}$ のハッシュ値である。

（ハ）電子現金の送付

ユーザ U がユーザ W に電子現金 T を送付する際の手順は以下のとおりである。

① ユーザ U は、ユーザ W から $(pk_{W(j)}, com_{(j)})$ を受領する。
② ユーザ U は、$chal_{(i)} \leftarrow H(T, pk_{W(j)}, com_{(j)}, com_{(i)})$ を計算する。
- $com_{(i)}$ は、ユーザ U がユーザ V から電子現金 T を受け取った際に生成したもの（電子現金の受領プロトコル：ステップ②）

③ ユーザ U は、$res_{(i)} \leftarrow RES(com_{(i)}, chal_{(i)}, (pk_{U(i)}, pk_C), (pk_U, sk_U, Cer_U, r_{U(i)}, r_U), r_{(i)})$ を計算する。なお、Σ プロトコルは、$(pk_{U(i)}, pk_C)$ が以下の言語 L に属することを証明するものである。$L := \{(pk_{U(i)}, pk_C) | \exists (pk_U, sk_U, Cer_U, r_{U(i)}, r_U), Commit(sk_U ; r_{U(i)}) = pk_{U(i)} \wedge Commit(sk_U ; r_U) = pk_U \wedge Verify_{pk_C}(pk_U, Cer_U) = 1\}$.
④ ユーザ U はユーザ W に $(T, \{\sigma_{(n+1)} = (com_{(i)}, res_{(i)})\}_{n \geq 0})$ と移転履歴にあるユーザの取引用公開鍵を送信する。

⑤ ユーザ W は、T の検証を行うとともに、$\{\sigma_{(n)}=(com_{(k)}, res_{(k)})\}_{n\geq 0}$ について $VER(com_{(i)}, chal_{(i)}, res_{(i)}, (pk_{U(i)}, pk_C))$ を計算し、すべての結果が 1 であれば電子現金を受領し、そうでなければ拒否する。なお、$chal_{(i)} \leftarrow H(com_{(i)})$ は $com_{(i)}$ のハッシュ値である

（ニ）二重使用の検知

サービス事業者が電子現金を二重使用したユーザを特定する際の手順は以下のとおりである。

① サービス事業者は、還流してきた電子現金 T のシリアル・ナンバが発行済電子現金 DB になければ、二重使用されたものと判断し、還収済電子現金 DB から同シリアル・ナンバをもつ電子現金 T とその流通履歴 σ' を取り出す。
② サービス事業者は、2 つの移転履歴 σ と σ' について、$sk_U \leftarrow EXT(pk_{U(i)}, com, chal, chal', res, res')$ を計算する。
③ サービス事業者は、認証機関に照会することで、sk_U を秘密鍵とするユーザを特定する。

ハ．4 節で紹介した方式（公開鍵を認証機関に登録、二重使用者の特定は公開鍵で行う）

方式 1 と 2 における二重使用の検知は、不正に使用された秘密鍵の暴露によってユーザを特定する。ただし、二重使用に使用された 2 つの電子現金が手元にあれば、誰でも秘密鍵を暴露することができるため、同秘密鍵を得たユーザによってさらなる不正が行われるリスクがある。これに対して、本方式は、公開情報のみから不正者の特定を可能とする特徴を有する。

（イ）初期設定

認証機関とユーザは暗号処理に使用する鍵ペアと証明書を以下の手順で用意する。

① 認証機関は、秘密鍵と公開鍵のペア $(sk_C, pk_C) \leftarrow Gen(1^\delta)$ を生成し、自己証明書とともに公開する。
② ユーザ U は、秘密鍵と公開鍵のペア $(sk_U, pk_U) \leftarrow Gen(1^\delta)$ を生成する。
③ ユーザ U は、認証機関に公開鍵 pk_U を送付し、pk_U に対する証明書 $Cer_U \leftarrow Sign_{sk_C}(pk_U \| *)$ を受け取る。
④ ユーザ U は、$Verify_{pk_C}(pk_U, Cer_U)$ を計算し、結果が 1 であれば (pk_U, Cer_U) を受領し、そうでなければ拒否する。

（ロ）電子現金の受領

ユーザ U がユーザ V から電子現金 T を受領する際の手順は以下のとおりである。

① ユーザ U は、ランダムに取引用公開鍵 $pk_{U(i)} \in \{0,1\}^{\lambda}$ を選択する。
② ユーザ U は、$Cer_{U(i)} \leftarrow Sign_{sk_U}(pk_{U(i)})$ を計算する。
③ ユーザ U は、乱数 $r_{(i)} \in \{0,1\}^{\lambda}$ を選択し、$com_{(i)} \leftarrow COM((pk_{U(i)}, pk_C), (pk_U, Cer_{U(i)}, Cer_U); r_{(i)})$ を計算する。
④ ユーザ U は、ユーザ V に $(pk_{U(i)}, com_{(i)})$ を送信し、ユーザ V から $(T, \{\sigma_{(n)} = (com_{(k)}, res_{(k)})\}_{n\geq 0})$ と移転履歴にあるユーザの取引用公開鍵を受領する。
⑤ ユーザ U は、T の検証を行うとともに、$\{\sigma_{(n)} = (com_{(k)}, res_{(k)})\}_{n\geq 0}$ について、$VER(com_{(k)}, chal_{(k)}, res_{(k)}, (pk_{V(k)}, pk_C))$ を計算し、すべての結果が 1 であれば電子現金を受領し、そうでなければ拒否する。

（ハ）電子現金の送付

ユーザ U がユーザ W に電子現金 T を送付する際の手順は以下のとおりである。

① ユーザ U は、ユーザ W から $(pk_{W(j)}, com_{(j)})$ を受領する。
② ユーザ U は、$chal_{(i)} \leftarrow H(T, pk_{W(j)}, com_{(j)}, com_{(i)})$ を計算する。
- $com_{(i)}$ は、ユーザ U がユーザ V から電子現金 T を受け取った際に生成したもの（電子現金の受領プロトコル：ステップ③）

③ ユーザ U は、$res_{(i)} \leftarrow RES(com_{(i)}, chal_{(i)}, (pk_{U(i)}, pk_C), (pk_U, Cer_{U(i)}, Cer_U))$ を計算する。なお、Σ プロトコルは、$(pk_{U(i)}, pk_C)$ が以下の言語 L に属することを証明するものである。$L := \{(pk_{U(i)}, pk_C) | \exists (pk_U, Cer_{U(i)}, Cer_U), Verify_{pk_U}(pk_{U(i)}, Cer_{U(i)}) = 1 \wedge Verify_{pk_C}(pk_U, Cer_U) = 1\}$.
④ ユーザ U はユーザ W に $(T, \{\sigma_{(n+1)} = (com_{(i)}, res_{(i)})\}_{n\geq 0})$ と移転履歴にあるユーザの取引用公開鍵を送信する。
⑤ ユーザ W は、T の検証を行うとともに、$\{\sigma_{(n)} = (com_{(k)}, res_{(k)})\}_{n\geq 0}$ について、$VER(com_{(i)}, chal_{(i)}, res_{(i)}, (pk_{U(i)}, pk_C))$ を計算し、すべての結果が 1 であれば電子現金を受領し、そうでなければ拒否する。

（ニ）二重使用の検知

サービス事業者が電子現金を二重使用したユーザを特定する際の手順は以下のとおりである。

① サービス事業者は、還流してきた電子現金 T のシリアル・ナンバが発行済電子現金 DB になければ、二重使用されたものと判断し、還収済電子現金 DB から同シリアル・ナンバをもつ電子現金 T とその流通履歴 σ' を取り出す。

② サービス事業者は、2つの移転履歴 σ と σ' について、$pk_U \leftarrow EXT(pk_{U(i)}, com, chal, chal', res, res')$ を計算する。
③ サービス事業者は、認証機関に照会することで、pk_U を公開鍵とするユーザを特定する。