Probabilistic micropayments with transferability

Taisei Takahashi and Akira Otsuka

Institute of Information Security
Yokohama, Japan
{dgs194102, otsuka}@iisec.ac.jp

Abstract. Micropayments are one of the challenges in cryptocurrencies. The problems in realizing micropayments in the blockchain are the low throughput and the high blockchain transaction fee. As a solution, decentralized probabilistic micropayment has been proposed. The winning amount is registered in the blockchain, and the tickets are issued to be won with probability p, which allows us to aggregate approximately $1/p$ transactions into one. Unfortunately, existing solutions do not allow for ticket transferability, and the smaller p, the more difficult it is to use them in the real world. We propose a novel decentralized probabilistic micropayment *Transferable Scheme*. It allows tickets to be transferable among users. By allowing tickets to be transferable, we can make p smaller. We also propose a novel *Proportional Fee Scheme*. This is a scheme where each time a ticket is transferred, a portion of the blockchain transaction fee will be charged. With the proportional fee scheme, users will have the advantage of sending money with a smaller fee than they would generally send through the blockchain. For example, sending one dollar requires only ten cents.

Keywords: blockchain · micropayment · transferability · tamper-proof wallet.

1 Introduction

Micropayments are minimal payments, e.g., less than $1, and can be used in a wide range of applications, such as per-page billing in e-book and deliver contents billed per minute. However, it is challenging to realize micropayments in the blockchain.

The problems in realizing micropayments in the blockchain are the low throughput and the high blockchain transaction fee. Since the capacity of each block is fixed, miners give priority to transactions that can generate high fees and put off micropayment transactions with low fees. In addition, the blockchain transaction fees do not depend on the amount of money to be transferred. Thus, the blockchain transaction fees can be relatively small for high-value transfers but high for micropayments.

The above problems can be solved by Layer-two [6]. Instead of registering all transactions in the blockchain, Layer-two aggregates small transactions into

a few larger ones, which can increase transaction throughput and reduce transaction fees. Decentralized probabilistic micropayments have been proposed as one of the methods for Layer-two. It is a lottery-based scheme, the amount of required payments is locked in an escrow, and micropayments are issued as lottery tickets. Let the winning amount be β, and the winning probability is p, the expected value per lottery ticket is $p \cdot \beta$, and the ticket is used as currency. Probabilistic micropayments allow us to aggregate the entire transactions by approximately p. As an example, if $10,000$ transactions are to be processed by a probabilistic micropayments scheme, only $10,000 \cdot p$ will be registered in the blockchain.

Almashaqbeh et al. have proposed MicroCash [2] which is a lightweight protocol for non-interactive and sequential payments. The disadvantage of MicroCash is that the game theory guarantees safety against double-spending attacks. Thus, the penalty escrow, which is confiscated with the double-spending attack is discovered, is expensive. As an example, when $m = 5$ and $B_{\text{escrow}} = 2000$, the penalty escrow is $B_{\text{penalty}} = 477.6$. In addition, tickets can only be sent once by the ticket issuer; in other words, the tickets can not be *transferable*.

As MicroCash, when safety is constructed using an only game-theoretic approach, considering penalty escrow, the number of beneficiary users who can receive the ticket, u, is realistically constrained to about 5. If we make u large, we need to make the penalty escrow large in proportion to u. As an alternative plan, if we assume the situation that the users can not commit malicious activity, such as tamper-resistant assumption, u can be large without penalty escrow. However, the smaller p is, the higher the gambling potential becomes and the less the payee can use it for actual economic transactions. If many tickets with a minimal winning probability are sent and not winning, the beneficiary merchants can not make any income. This is because if the ticket can not be transferable, the payee will not earn any income unless the ticket they received wins. The smaller p, the more the opportunity to get an income is lost.

If the ticket is *transferable*, p can be reduced. The payees does not lose anything since the ticket can be used to pay others even if the ticket is not won. However, it is challenging to achieve transferability with existing solutions. Since if the ticket is transferable, the double-spending attacks can be performed by the issuer and all users. Requiring game-theoretically guaranteed penalty escrow for all users is practically undesirable because of high collateral costs. Suppose the ticket transfer is limited to a tamper-proof device, malicious activities that deviate from the protocol can be prevented, and transferability can be achieved without the need for high penalty escrow.

1.1 Contribution

We propose a novel decentralized probabilistic micropayments, *Transferable Scheme*, which allows tickets to be transferable among users. Instead of a game-theoretic approach, we introduce a tamper-proof assumption, which states that all users can only issue, send, and receive tickets through tamper-proof wallets created by trusted manufactures.

Theoretically, users are not able to perform double-spending attacks through tamper-proof wallets. However, it is not possible to assume tamper-proof completely. In reality, a tamper-proof device can be broken, and a double-spending attack can be performed. For this reason, in this study, instead of assuming a tamper-proof wallet and eliminating the need for penalty escrow, we force adversaries to weigh the cost of breaking a tamper-proof wallet against the maximum expected value that can be obtained from the attack. As long as the expected value does not exceed the cost, there will be no incentives for an adversary to perform the attack. Furthermore, we propose a mechanism to detect the attack with probability $p = 1$, and that the adversary's wallet address is unavailable when the attack is detected. This creates a need for an adversary to weigh the cost of breaking the wallet against the expected utility gain of a single attack.

Furthermore, we propose a novel *Proportional Fee Scheme*. This scheme is where each user who sends and receives a ticket bears a small portion of the blockchain transaction fee required when the ticket is won. This makes it possible for payment with a smaller fee than in the blockchain.

1.2 Organization of This Paper

This paper is structured as follows: Section 2 surveys the works related to our proposed scheme. Section 3 outlines our payment scheme, and Section 4 presents the ticket winning condition. In Section 5 we introduce our new payment fee scheme, "Proportional Fee Scheme," and Section 5 presents the security design.

2 Background

Payment Channels and Networks The payment channel establishes a private, peer-to-peer transmission protocol. Based on pre-defined rules, two parties can agree to update their state and transfer money by exchanging authenticated state transitions in a so-called 'off-chain' fashion.

In order to conduct a transaction on *Payment Channel*, two parties must first register a shared 2-of-2 multi-sig escrow fund in the blockchain and establish the channel. The payment channel enables the two parties to perform transactions through private communications. After the sending and receiving are completed in the channel, the final fixed value is registered in the blockchain. Only two transactions are registered in the blockchain per channel, escrow fund transaction, and final fixed value. A payer can send money to a user who has not established a channel with the payer through the *Payment Network* between users who have established a channel. For example, suppose Alice sends 0.1 coins to Charlie, who has not established a channel with Alice. First, Alice sends 0.1 coins to Bob, whom Alice has a channel. Next, Bob sends 0.1 coins to Charlie, whom Bob has a channel.

Unfortunately, the payment channel and the network have the disadvantage of high collateral cost [8]. Each time a channel is established, escrow is required

between two parties. Also, the longer the payment network path, the more reserves are required and locked. Since the reserves can not be used during the locktime periods, the reserves represent a lost opportunity. Furthermore, in a payment network, a fee is charged for each pass through the nodes. It is impractical to adopt a payment network for micropayments since it is undesirable to incur the cost for each node.

Probabilistic micropayment The idea of probabilistic micropayments has been proposed by Wheeler [12] and Rivest [10]. Since small payments would be costly if settled each time, they proposed a lottery-style protocol where the ticket issuer deposit a large amount of money in the bank, and the winner could receive the money if they won. The lottery tickets can be used as currency, and the value per ticket is regarded as the expected value of the ticket. In this scheme, the existence of a bank is mandatory, and participants are limited to people who have a relationship with the bank.

MICROPAY [9] and DAM [4] have been proposed as decentralized probabilistic micropayments using blockchain. Since both have a large overhead of supporting sequential micropayments, Almashaqbeh et al. have proposed MicroCash which is a light-weight protocol for non-interactive and sequential payments.

The drawback of existing solutions is that the lottery ticket can only be sent from the ticket issuer to the recipient. Also, because of game-theoretic suppression of double-spending attacks, the penalty escrow increases proportionally with the number of recipients. Furthermore, the smaller the probability of winning p, the higher the gambling potentials becomes, and the more unstable the income-earning opportunity for the recipient.

Secure offline payment The double-spending attack on fast payment is one of the fatal architectural problems in cryptocurrencies [7]. Dmitrienko et al. proposed an offline fast payment scheme that relies on tamper-proof wallets produced by trustworthy manufacturers. However, their scheme requires a trusted online time-stamp server. Takahashi et al. [11] overcome this drawback and proposed a protocol that allows secure offline payment using tamper-proof device wallet.

3 Ticket Transfer Overview

This section presents the design of our transferable scheme. We start with an outline of the lottery ticket transaction, followed by a detailed description of each part.

3.1 Outline

The outline of the system is shown in Figure 1.

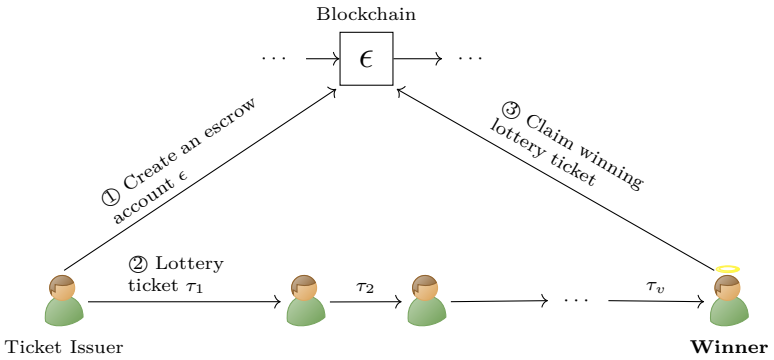


Fig. 1: Overall Design

Step 1, The issuer issues a smart contract escrow account ϵ and registers and confirms that ϵ has been registered in the blockchain. **Step 2**, The issuer issues the ticket τ for probabilistic micropayments and sends it to a user. The payee verifies that the ticket came from a legitimate wallet and that the escrow account is properly registered in the blockchain. If there is no problem, the user receives the ticket and returns the service or product to the payer. Then, the payee signs the ticket with his wallet and sends it to another user. **Step 3**, If the ticket received meets the requirements for winning, the ticket is sent to the escrow account ϵ.

The sequence of procedures in this scheme, such as ticket issuance and payment with the ticket, is done using a tamper-proof wallet.

Tamper-proof wallet The premise is that all users participating in the transferable scheme have tamper-proof hardware wallets.

The wallet consists of a tamper-proof device manufactured by a trusted manufacturer. It does not accept any unauthorized operation that deviates from the protocol, such as double-spent tickets.

There are two keys in the wallet. One is a key for personal use key pairs (sk^{Wx}, PK^{Wx}) for sending and receiving the ticket, we denote the hash value of PK^{Wx} be the "address" associated with the wallet owner. The other is a secret key sk^T used to prove that the ticket was created and sent from a legitimate wallet. Additionally, the wallet owner possesses a certificate $cert_T$ corresponding to the private key sk_T.

3.2 Escrow Setup

The flow diagram is shown in Figure 2. The issuer X requests a new account w_X from the wallet (Step 1), then create the escrow transaction τ_l transferring β coins from the account x to the wallet address w_X and commit it to the networks (Step 2). As soon as τ_l is verified and integrated into the Blockchain network in

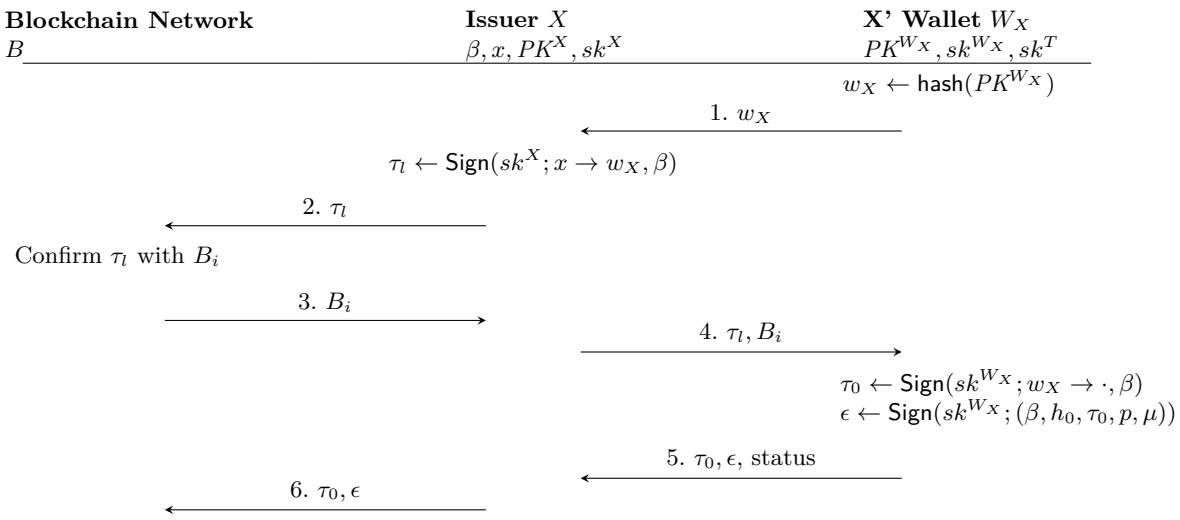


Fig. 2: Escrow Setup

a block, say B_i, X takes B_i (Step 3), and provides τ_l and B_i to W_X (Step 4). W create the escrow account ϵ. Then, sends it to X with status (Step 5). Finally, X sends τ_0 and ϵ to the Blockchain network. [1]

3.3 Payment with Lottery Ticket

The flow diagram is shown in Figure 3. In the payment with lottery ticket phase, the payee Y sends PK^{W_Y} (Step 1). The wallet W_X creates a ticket τ_1 and signs it with the private key sk^{W_X}, and signs the ticket τ_1 with the wallet manufacturer's private key sk^T. The wallet W_X sends ticket τ_1, $proof_1$, and $cert^T$ to the payee' wallet W_Y (Step 2). If all checks succeed, Y stores τ_1, $proof_1$, and replies to W_X with the status (Step 3). If the payee Y wants to send the received ticket to another user, the same procedure is followed from Step 1.

3.4 Ticket Winning and Revocation

The flow diagram is shown in 4. If $\tau \in \mathsf{win}$, Y sends τ and *proof* to the contract account ϵ (Step 1). If τ is both valid and $\mathsf{eligible}$, the escrow account ϵ signs the escrow transaction τ_0 with w_Y as the destination. The payee Y observes the blockchain network and periodically updates its local chain and confirms τ_0 is valid (Step 2).

[1] The wallet does not check the validity of the escrow transaction τ_0 and ϵ. Payees will reject the ticket which is not transferred from ϵ.

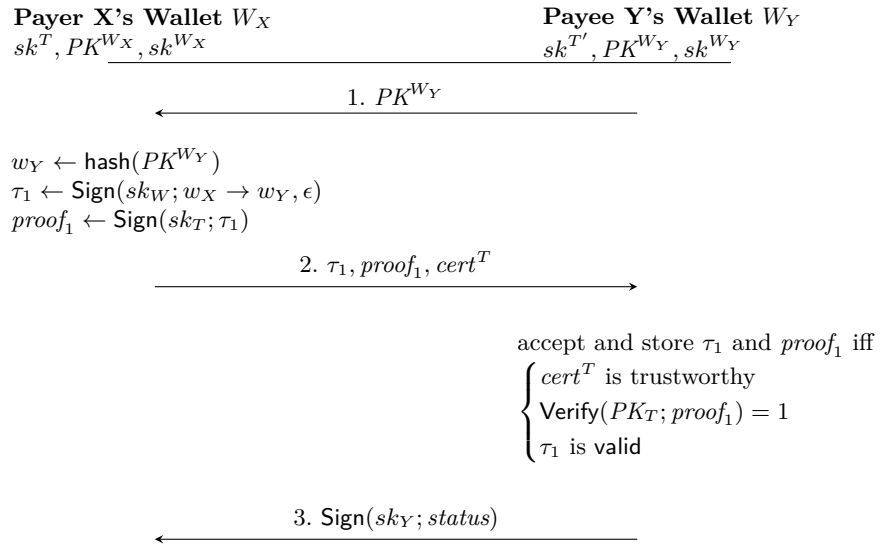


Fig. 3: Payment with Lottery Tickets

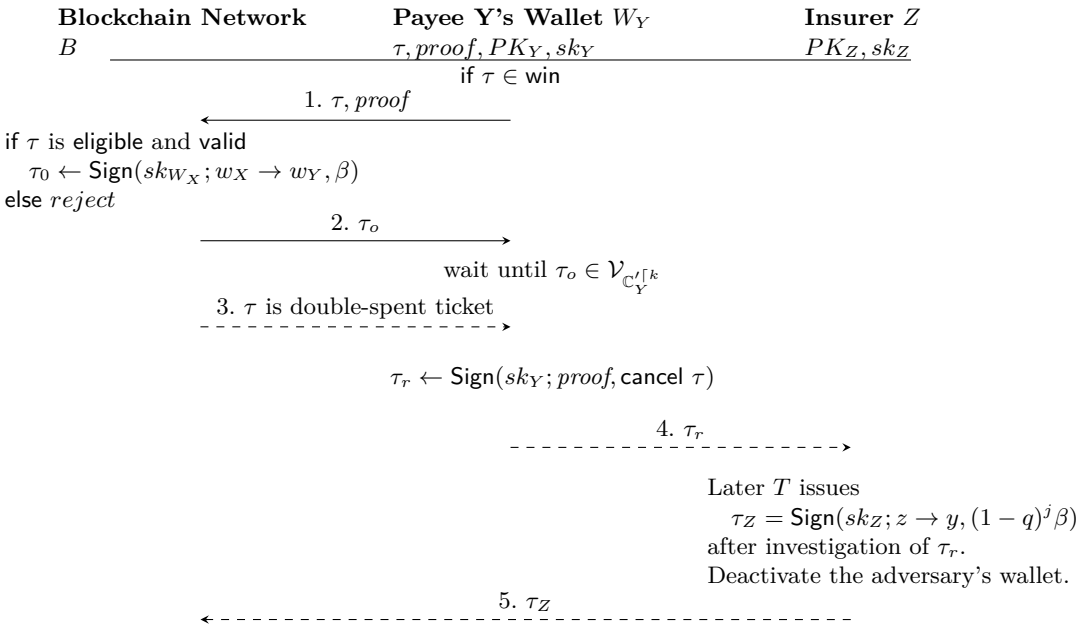


Fig. 4: Ticket redemption and double-spending wallet revocation protocol

If τ is one of the double-spent tickets created by a double-spending attack, the contract account ϵ shows that τ is double-spend one (Step 3). [2] Y initiates revocation by creating a revocation transaction $\tau_r = \mathsf{Sign}(sk_Y; \mathit{proof}, \mathsf{cancel}\ \tau)$ and send it to Insurer Z (Step 4). Z investigate τ_r and in order to compensate Y for the damage, issues τ_Z then committed to the Bitcoin network (Step 5). [3]

4 Ticket Winning Condition

- $\mathbb{C}$: a blockchain.
- $\mathcal{U}$: a set of users.
- $X, Y \in \mathcal{U}$: (typically, X as a payer, Y as a payee).
- l: the number of double-spent (duplicated) tickets by an adversary
- ϵ: escrow account which has several fields: $(\beta, h_0, \tau_0, p, \mu)$
  - β: the lottery winning amount
  - h_0: the block height containing the escrow account
  - τ_0: escrow creation transaction
  - p: the probability for determining of winning a ticket
  - μ: the fixed number to calculate the winning ticket($\in \mathbb{N}$)
- τ: a lottery ticket which has several fields : $(A, B, \tau_{pre}, \sigma)$
  - A: a sender
  - B: a receiver
  - τ_{pre}: a reference to a previous ticket or to an escrow account ϵ
  - σ: signature by a sender
- Φ: the cost of breaking a tamper-proof hardware wallet
- γ: the blockchain transaction fee

4.1 Structure of the ticket

This section describes the structure of the lottery ticket and the design of the ticket winning method. If the ticket is transferable, a blockchain transaction fee is charged when the ticket is won and registered in the blockchain. We introduce a scheme where users who send and receive the ticket share the blockchain transaction fee little by little.

Definition 1. *A lottery tickets τ consists of a fivefold:*

$$\left(A, B, \tau_{pre}, \sigma^W, \sigma^T, \mathit{cert}^T\right) \tag{1}$$

where A and B are accounts of a sender and a receiver, respectively. τ_{pre} is a reference to a previous ticket or to an escrow account ϵ. A pair of signatures, σ^W and σ^T, is a multi-signature, where σ^W is signed with a signing key tied with a sender's account and σ^T is signed with a tamper-proof device's signing key to

[2] Double-spending attacks can be perfectly detected, and the adversary's address is discovered. See section 6.

[3] The compensated amount is the same as the return when received the ticket. See section 5 for the value of a ticket when it is in transfer.

prove that the signing device is trusted verifiable with a certificate $cert^T$ issued by a trusted manufacturer. We denote by σ_A to denote a signature signed by A. The escrow account ϵ further contains $(\beta, h_0, \tau_0, p, \mu)$ to specify the parameters of the transferable transaction, where β is the ticket winning amount, and h_0 is the block height to specify particular VDF values. τ_0 is the escrow creation transaction. p is the probability for determining of winning a ticket. μ is a fixed value used to determine the winning ticket.

For readability, we write a ticket τ as:

$$\tau = (A \to B, \tau_{\text{pre}})_X \,. \tag{2}$$

We define $|\tau|$ the "number of generations" of τ, which is the length of the sequence from ϵ to τ. For example, $|\tau| = n$ if there exists a sequence $\tau_1, \ldots, \tau_{n-1}$ such that $\epsilon \prec \tau_1 \prec \tau_2 \prec \cdots \prec \tau_{n-1} \prec \tau$. We define $|\tau| = \infty$ if no such sequence exists [4]. To write compactly, we denote by τ_i the i-th generation of τ.

Definition 2 (Transferred transaction). *Two tickets $\tau_i = (A \to B, \tau_{pre})_X$ and $\tau_{i+1} = \left(A' \to B', \tau'_{pre}\right)_{X'}$ are said to be* transferred *if and only if following properties satisfies:*

$$\begin{cases} H(\tau_i) = \tau'_{pre} \\ A = X,\ B = A' = X' \\ cert^T_{X'} \text{ is trustworthy} \\ \text{multi-signature } \sigma^W_{X'} \text{ and } \sigma^T_{X'} \text{ are valid} \end{cases} \tag{3}$$

Then, we write $\tau_i \prec \tau_{i+1}$.

We write $\tau_i \ll \tau_{i+n}$ if there exists a sequence of ordered lottery tickets $\tau'_1 \prec \ldots \prec \tau'_n$ for $n \geq 1$ and they satisfy $\tau_i \prec \tau'_1$ and $\tau'_n \prec \tau_{i+n}$. In the case where τ has no previous lottery tickets, the ticket is called a 'genesis' ticket. For the genesis tickets τ_1 tied to an escrow account ϵ, we specially denote by $\epsilon \prec \tau_1$ so that a lottery tickets are simply written as:

$$\epsilon \prec \tau_1 \prec \tau_2 \prec \ldots \prec \tau_n. \tag{4}$$

Definition 3. *A lottery tickets τ is said to be* valid *with respect to a blockchain $\mathbb{C}$ for some security parameter k if and only if there exists an escrow account ϵ and a sequence of transactions $\tau_{i,1}, \ldots, \tau_{i,n}$ such that*

$$\epsilon \in \mathbb{C}^{\lceil k} \quad \textit{and} \quad \epsilon \prec \tau_1 \prec \ldots \prec \tau_n \prec \tau. \tag{5}$$

$\mathbb{C}^{\lceil k}$ denotes the set of blocks that are k or more blocks before the beginning of the blockchain. This notion is borrowed from Garay et al [5].

[4] For practical purposes, we assume that the height of τ can only be measured when all tickets in the sequence from ϵ to τ are given. Even if such sequence exists, the height of τ is considered to be ∞ unless the entire sequence is specifically presented.

4.2 Ticket Winning Condition

This section describes the design of the ticket winnings.

Definition 4. *$\tau_{i,v}$ is said to be* win *if and only if the following properties satisfies:*

$$\mathsf{win} = \left\{ \tau_v \;\middle|\; p : H\left(\mathrm{VDF}(h_0 + v \cdot \mu)\right) < D \;\; \textit{for all } v \in \mathbb{N} \right\} \tag{6}$$

where v is the number of generations of τ and μ is the fixed number specified in the escrow account ϵ.

h_0 is registered in ϵ, which specifies the block height at which ϵ would be registered. The probability p is calculated using a simple Verifiable Delay Function (VDF) [3]. The calculation can be done after a certain period of time has elapsed from when the ticket is transferred according to the number of generations. For example, if a ticket with $h_0 = 100$, $\mu = 5$, and $v = 3$ is received, the VDF value will be known when the block height of 115 is confirmed.

As described in the next subsection 5, even though the ticket meets the requirements win, the ticket may be used as payment instead of getting the winning amount β. If a ticket $\tau \in \mathsf{win}$ has already been transferred, the user with the most recent ownership can get the winning amount β.

Definition 5. *τ_v is said to be* eligible *if and only if the following properties satisfies:*

$$\mathsf{eligible} = \left\{ \tau_{v'} \;\middle|\; \tau_{v'} = \max(\{\tau_{v''} | v'' \geq v\}) \right\} \tag{7}$$

eligible ticket will be considered as the final winning ticket. Thus, the user who has the eligible ticket can get β from the escrow account ϵ.

5 Proportional Fee Scheme

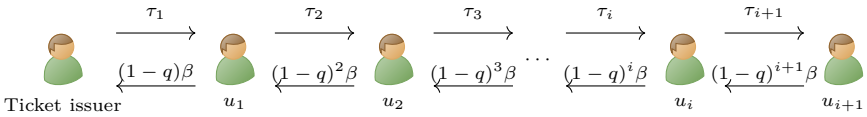


Fig. 5: Proportional Fee Scheme

In this section, we consider the blockchain transaction fee to transfer the winning amount to the winner's address and the value of the ticket in the transfer process.

In our transferable scheme, it is not beneficial for the issuer to bear the blockchain transaction fee. Since when the issuer bears the blockchain transaction fee, the amount available for payment is $\beta - \gamma$, which does not provide any advantage for the issuer to use the transferable scheme.

We propose a novel *Proportional Fee Scheme*. The process is depicted in Figure 5. This scheme is where each time a payer transfers a ticket, the payer borne the fee based on the number of generations of the ticket. When a payer sends τ_j to the payee, in return, the payee gives goods or services worth $(1-q)^j\beta$.

Definition 6 (Proportional fee scheme). *Let q be the lottery ticket transaction fee rate. Suppose a payer sends a ticket τ_i, in return the payee gives goods or services worth $(1-q)^i\beta$ to the sender. The fees borne by the payment is $(1-q)^{i-1}q\beta$.*

Specifically, the fee for each payment is $\tau_{i-1}-\tau_i = (1-q)^{i-1}q\beta$, and the profit (income − expenditure) when $\tau_i =$ eligible is $\beta - (\tau_i + \gamma) = \left(1-(1-q)^i\right)\beta - \gamma$ where γ is the blockchain transaction fee.

Suppose the ticket satisfies the win condition before the accumulated fees exceed the blockchain transaction fee γ. In this case, the user may decide whether to send it to the blockchain network and get β or transfer the ticket to another user as payment. Specifically, the user can profit from the eligible ticket by getting the winning amount β under the following condition:

$$\left(1-(1-q)^i\right)\beta > \gamma. \tag{8}$$

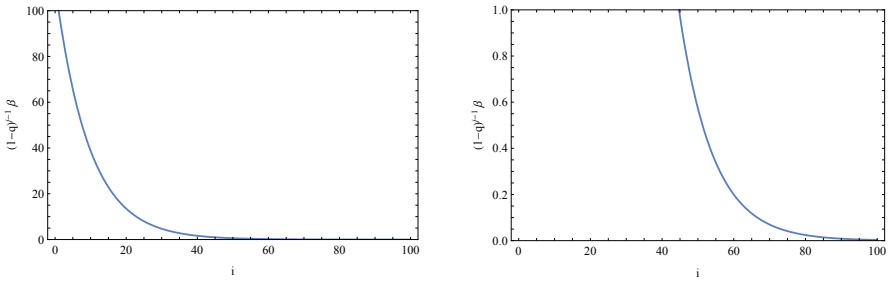
If the ticket satisfies the win condition is transferred to another user, the ticket is distributed as eligible and can be sent to the blockchain in any subsequent generation. Naturally, the ticket will be sent to the blockchain network in the generation that satisfies the equation 8.

This scheme has the advantage that the payment fee can be smaller than the blockchain transaction fee. The average transaction fee for cryptocurrencies, especially Bitcoin, is around $11 to $15 [1].

In our transferable scheme, let $\beta =$ \$100, $p = \frac{1}{100}$ and $q = \frac{1}{10}$, the ticket value per generation is depicted in Figure 6. As we can see from Figure 6b, the value of the ticket falls below \$1 from approximately $i = 50$. Figure 7 shows the frequency of the fee, and we can see that there are more than 50 transactions whose value is less than \$1. Since the fee per payment is roughly $q = \frac{1}{10}$, the fee for a \$1 transfer is about 10 cents.

Both the existing Lottery scheme and our Transferable scheme can aggregate blockchain transactions by the winning probability p. The difference is that our transferable scheme does not increase the gambling potential, even makes the winning probability p smaller. In the existing scheme, the smaller p is, the lower the probability that the payee will win the ticket, which makes the income more unstable for the payees. In our transferable scheme, even if the ticket is not winning, the payee can use it for payment by paying a smaller fee than the blockchain transaction fee.

There is a concern that the sizeable winning amount β decreases the velocity of the ticket. This is because if there is a large gap between the winning amount β and the value of the ticket, the profit of winning $\beta-\gamma$ will be more significant. Therefore, it is best for recipients to decide whether to use the ticket for payment after confirming their winnings, which causes the velocity of the ticket to be slow.



(a) < 100 (b) < 1

Fig. 6: Ticket value per generation

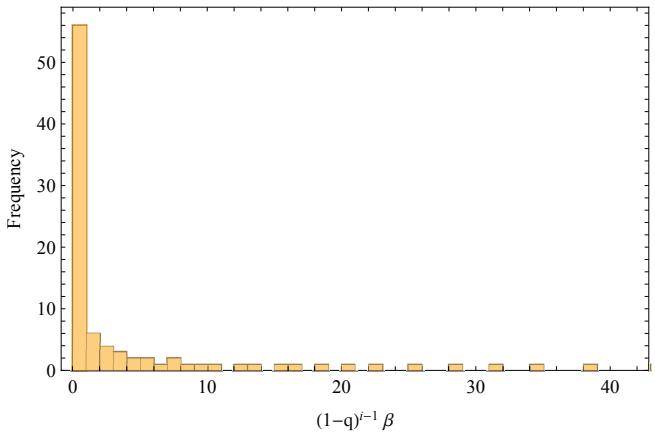


Fig. 7: Frequency of ticket value

The solution is not to make the winning amount β too high. In addition, if we set the winning amount β to a value almost equal to the blockchain transaction fee γ, the profit of winning will be very small; thus, the velocity of the ticket will not be affected.

6 Security Design

As far as the tamper-resistant assumption holds, double-spending attacks can not be performed theoretically. However, in reality, the tamper-proof hardware wallet could be broken, leading to double-spending attacks. Thus, instead of requiring a penalty escrow, we design security from the perspective of whether the utility an adversary can gain from the attack exceeds the cost of breaking the tamper-resistant hardware.

Definition 7 (κ-tamper proof). *A device is called κ-tamper proof if it satisfies the following conditions:*

1. *tamper-proof hardware is the hardware that prevents an adversary from stealing and changing stored data.*
2. *the device is either completely broken/tampered or working perfectly with probability κ and $(1-\kappa)$, respectively* [5].
3. *broken/tampered is a state in which all confidential information inside the device, including the private key, has been leaked to the adversary.*

We assume each device is in a state either completely broken/tampered or working perfectly. They occur with probabilities κ and $(1-\kappa)$, respectively. As long as the behavior is observed from outside, it is not possible to distinguish between a device that is operating correctly and a device that adversary control the correct device who have an access to its internal key.

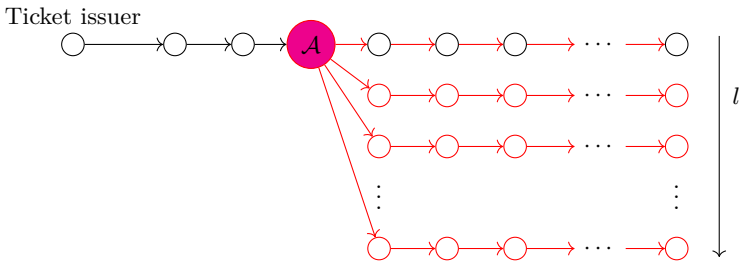


Fig. 8: Double-spending Attack

[5] In reality, the adversaries are biased, but we assume it can not be distinguishable from a legitimate user from outside.

Double-spending Attack Double-spending attacks are an attack that makes a profit by duplicating and double-spending the received ticket. In our transferable scheme, we assume that the adversary breaks the κ-tamper proof and receives or issues tickets then transfers with different addresses (wallets). We call the tickets created by the double-spending attacks "duplicated."

In our transferable scheme, we assume an adversary can gain profit up to $l \cdot \beta$ where l is the number of duplicated tickets. This is illustrated in Figure 8.

6.1 Detection Methods

This section describes how to detect the double-spending attacks and find the adversary's address. We introduce two methods to detect the attack and find the adversary's address perfectly. When the adversary's address is found, we assume that the address is broadcasted to all users, and then the adversary's address is rejected by all users. An adversary can not profit unless the maximum expected utility he can gain from a single attack exceeds the cost of breaking the κ-tamper proof wallet.

Note that the following two detection methods require that the receiver be online at the time of receipt. Conversely, the payer does not need to be online.

Definition 8 (Fork of transferred transactions).
Given two series of transactions initiated with the same escrow account $\epsilon \prec \cdots \prec \tau$ and $\epsilon \prec \cdots \prec \tau'$, the series of transactions are said to be 'fork' if and only if it satisfies both $\tau \not\prec \tau'$ and $\tau' \not\prec \tau$.

Assume the users monitor the blockchain, and after the eligible ticket is registered in the blockchain, the users check the eligible ticket against the ticket they have received.

Theorem 1 (Fork Detection). *Double-spending attacks can be perfectly detected by fork detection described in definition 8.*

Proof. Assume there exists forked two series of transactions τ and $\tilde{\tau}$. Given τ is eligible and registered in the blockchain, the user who has $\tilde{\tau}$ reports double-spending attack. The adversary's address is confirmed from the latest common prefix of τ and $\tilde{\tau}$. □

Definition 9 (Collision of transferred transactions). *Assume that each of the u users has $\alpha\,(\geq 2)$ addresses. If an adversary sends at least two duplicated tickets to any one of u users, the 'collision' occurs.*

We adopt a *round scheme* so that the adversary can not profit when the collision is detected. We divide the ticket sending procedure into three rounds. The procedure is illustrated in Figure 9. **Round** 1) The adversary sends the tickets to the honest payees. The payee checks the received tickets for the collisions. **Round** 2) If the payee find the collision, broadcasts τ and $\tilde{\tau}$ to the honest users. **Round** 3) If the collision is not detected, the payee gives products or services to the payer in return. If the collision is detected, the adversary's address is rejected and will never be accepted by all honest users.

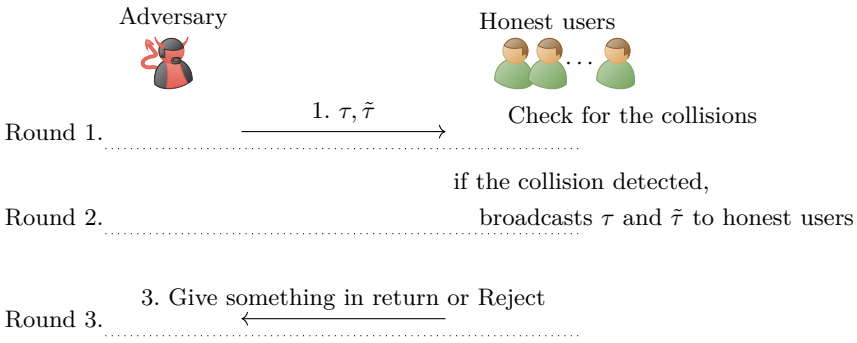


Fig. 9: Collision detection round

Theorem 2 (Collision Detection). *Let u be the number of users who participate in transfer scheme. By collision detection and round scheme, the expected utility of double-spending attack $\mathbb{E}_d$ is upper-bounded by the following inequality:*

$$\mathbb{E}_d \leq \sqrt{\frac{u}{e}}\beta. \tag{9}$$

Proof. As stated in the definition 9, we assume a uniform distribution where each user has α addresses[6]. This must be the case where the adversary choose uniformly l different addresses from the total of αu addresses. By the round scheme, the adversary can not profit if a single user address is chosen more than once.

Let $p(l; u)$ be the probability that at least one user address is chosen more than once. This probability is described as follows:

$$p(l; u) \approx 1 - e^{-\frac{l^2}{2u}}. \tag{10}$$

Assume that the adversary double-spent l tickets with a maximum value of β per ticket. The adversary's expected utility value is

$$\mathbb{E}_d < \max_l \left\{ l\beta \cdot (1 - p(l; u)) \right\}. \tag{11}$$

Thus, $\mathbb{E}_d$ is at most $\sqrt{\frac{u}{e}}\beta$ when $l = \sqrt{u}$. □

In our transferable scheme, double-spending attack is perfectly detected and the address used in the attack will be rejected by all users. Therefore, it is not profitable for the adversary unless the cost of breaking a single tamper-proof

[6] In reality, the number of addresses each user has is considered more likely to follow exponential distribution. It is an unfavorable assumption that all user have the same number of addresses α.

wallet exceeds the maximum expected value gained by the attack. Specifically, the adversary can not profit under the following conditions:

$$\sqrt{\frac{u}{e}}\beta < \Phi \tag{12}$$

where Φ is the cost of breaking κ-tamper proof wallet.

As an example, consider the maximum expected utility value $\mathbb{E}_d$ with $u = 1,000,000$ and $\beta = \$100$. Applying the equation 9 produces $\mathbb{E}_d \lessapprox \$60,700$.

7 Conclusions

In this paper, we introduce the first transferable decentralized probabilistic micropayment scheme and the proportional fee scheme. The feature of our scheme is that the ticket is transferable. Therefore, the ticket winning probability can be much smaller than the existing methods. Thus we can aggregate a larger number of transactions into one and can increase the blockchain throughput. Also, the proportional fee scheme can make the transaction fee smaller via the lottery ticket than on the blockchain.

Our scheme only assumes a tamper-proof device, and the ticket transfer protocol is simple, requiring only a digital signature. The tamper-proof assumptions can be achieved by SE (Secure Elements) such as SIM cards, which are widely used in Smartphones. Since the computational resources of SE are limited, the concern arises that it is not impractical to perform all operations in the SE. In our scheme, the operations to be performed in the SE can be limited to the prevention of double-spending. On the other hand, operations that are not related to double-spending prevention can be executed in the regular application area. Therefore, since the use of SE's computational resources can be minimized, it would be feasible to realize our scheme on mobile devices such as smartphones. Specifically, the operations to be performed in the SE are checking whether a ticket is valid, creating a key pair, and signing at the time of money transfer. On the regular application side, the operations are performed to avoid duplicated tickets (e.g., collision and fork detection) and check for winning tickets.

We consider that our transferable scheme is not a singular way of transferable lottery tickets but a system similar to the circulation of paper and coins issued by central banks. We will use blockchain to achieve this. We believe that our scheme can be applied not only to micropayments but also to high-value payment transactions.

References

1. Bitcoin Average Cost Per Transaction (2020), `https://ycharts.com/indicators/bitcoin_average_cost_per_transaction`
2. Almashaqbeh, G., Bishop, A., Cappos, J.: MicroCash: Practical Concurrent Processing of Micropayments. In: Financial Cryptography and Data Security. pp. 227–244. Lecture Notes in Computer Science, Springer International Publishing (2020)

3. Boneh, D., Bonneau, J., Bünz, B., Fisch, B.: Verifiable Delay Functions. In: Advances in Cryptology – CRYPTO 2018. pp. 757–788. Lecture Notes in Computer Science, Springer International Publishing (2018)
4. Chiesa, A., Green, M., Liu, J., Miao, P., Miers, I., Mishra, P.: Decentralized Anonymous Micropayments. In: Advances in Cryptology – EUROCRYPT 2017. pp. 609–642. Lecture Notes in Computer Science, Springer International Publishing (2017)
5. Garay, J., Kiayias, A., Leonardos, N.: The Bitcoin Backbone Protocol: Analysis and Applications. In: Advances in Cryptology - EUROCRYPT 2015. pp. 281–310. Lecture Notes in Computer Science, Springer (2015)
6. Gudgeon, L., Moreno-Sanchez, P., Roos, S., McCorry, P., Gervais, A.: SoK: Layer-Two Blockchain Protocols. In: Financial Cryptography and Data Security. pp. 201–226. Lecture Notes in Computer Science, Springer International Publishing (2020)
7. Karame, G.O., Androulaki, E., Capkun, S.: Double-spending fast payments in bitcoin. In: Proceedings of the 2012 ACM conference on Computer and communications security. pp. 906–917. CCS '12, Association for Computing Machinery (2012)
8. Miller, A., Bentov, I., Bakshi, S., Kumaresan, R., McCorry, P.: Sprites and State Channels: Payment Networks that Go Faster Than Lightning. In: Financial Cryptography and Data Security. pp. 508–526. Lecture Notes in Computer Science, Springer International Publishing (2019)
9. Pass, R., shelat, a.: Micropayments for Decentralized Currencies. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. pp. 207–218. CCS '15, Association for Computing Machinery (2015)
10. Rivest, R.L.: Electronic lottery tickets as micropayments. In: Financial Cryptography. pp. 307–314. Lecture Notes in Computer Science, Springer (1997)
11. Takahashi, T., Otsuka, A.: Short Paper: Secure Offline Payments in Bitcoin. In: Workshop on Trusted Smart Contracts In Association with Financial Cryptography and Data Security. pp. 12–20. Lecture Notes in Computer Science, Springer International Publishing (2020)
12. Wheeler, D.: Transactions using bets. In: Security Protocols. pp. 89–92. Lecture Notes in Computer Science, Springer (1997)