On the Applicability of Payment Channel Networks for Allocation of Transport Ticket Revenues

Matthias Grundmann, Otto von Zastrow-Marcks, Hannes Hartenstein
KASTEL Security Research Labs
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany

***Abstract*—In many public transport networks, multiple providers cooperate to offer integrated services and, consequently, provide integrated fare collection. Thus, ticket revenues need to be redistributed so that each provider receives its respective share. Typically, the customers' travel behavior is surveyed and the fares paid are aggregated over certain periods of time, and the revenue is periodically allocated based on this information. To avoid a centralized trusted third party or the exchange of sensitive information between providers, we present an approach that integrates revenue allocation directly into the payment process: The proposed approach is based on payment channel networks and utilizes multi-hop payments to perform revenue allocation. We thereby show how to make use of payment channel networks in this setting as well as the corresponding benefits.**

***Index Terms*—Public Transport, Transit Systems, Payment Channel Networks, Revenue Allocation, Fare Clearing**

I. Introduction

Since more than 50 years, public transport systems have been integrated to improve efficiency with respect to cost and travel time [1], [2]. Usually, fare collection is also integrated: customers can buy a ticket from one provider for a route offered by another provider. We expect that this integration further increases with the development of payment convergence for multimodal transport [3], [4]. Integrated fare collection requires that ticket revenues need to be allocated to the providers participating in the integrated transit system. The question how to allocate revenues (also referred to as fare clearing) has been studied by previous work [2], [5]–[7]: These proposals require authentic data on customer behavior to determine a fair allocation and a central party that manages the redistribution of ticket revenues. The central party computes the providers' balances and instructs the providers to perform clearing transactions. Such a central party needs to be trusted as it is in a powerful position to choose which providers to accept, to take fees, and to observe all clearing payments. Collecting data that accurately represents customer behavior is a challenge that has already been studied [8]–[12] but existing approaches can only create approximations.

In this paper, we present an approach for revenue allocation without requiring a trusted central party and without the need to exchange batches of sensitive customer data between providers. Our approach is based on payment channel networks (PCNs) [13] which have been proposed as a means for scaling payments for blockchains. A payment channel between two parties can be used to transfer funds between these two parties without having to interact with the underlying layer, e.g., the blockchain, for each transaction. In a PCN, payment channels are interconnected so that participants of the PCN can perform multi-hop payments with parties with whom they do not have a common payment channel. PCNs have been shown practical for building systems for payments for charging of electrical vehicles [14], [15] and for toll collection [16]. We suppose that the revenue allocation problem could also benefit from an approach based on a PCN built by the mobility providers (and their customers as nodes attached to a mobility provider): The sender of a multi-hop payment in a PCN pays to one party in their payment channel but the receiver can be any other connected party in the PCN. The intermediate parties directly forward the payment and the payment amount is guaranteed to be delivered to the receiver. Using multi-hop payments for tickets, a customer could send a payment to one provider who forwards the payment to another provider. Thereby the revenue is directly allocated with the payment which circumvents the problem of determining a fair allocation later on. Thus, we address the following research question: *Can payment channel networks be used for ticket revenue allocation and what properties are thereby achieved?*

We address this question in a scenario in which a customer's trip is served by only one provider and the customer pays to another provider. The revenue is allocated by forwarding the fare to the provider serving the trip. In Section II, we specify the scenario and the problem and we define the desired properties that a protocol for ticketing should have which includes that details about a planned trip should be private. After providing background on used building blocks in Section III, we present in Section IV a protocol for ticket purchases that uses a PCN for payment. By analyzing the properties achieved by the protocol, we show in Section V that PCNs can be used to build a protocol that achieves the required properties. In Section VI, we show design options available for a PCN-based approach and possible extensions of the protocol. We discuss the properties and tradeoffs that come with PCNs in Section VII and conclude.

Our contributions are a protocol that integrates revenue allocation with ticket payment and provides privacy properties for customers, a discussion on the benefits and tradeoffs of a PCN-based approach as a payment system for public transport, and a novel approach for selling digital signatures for tickets based on adaptor signatures.

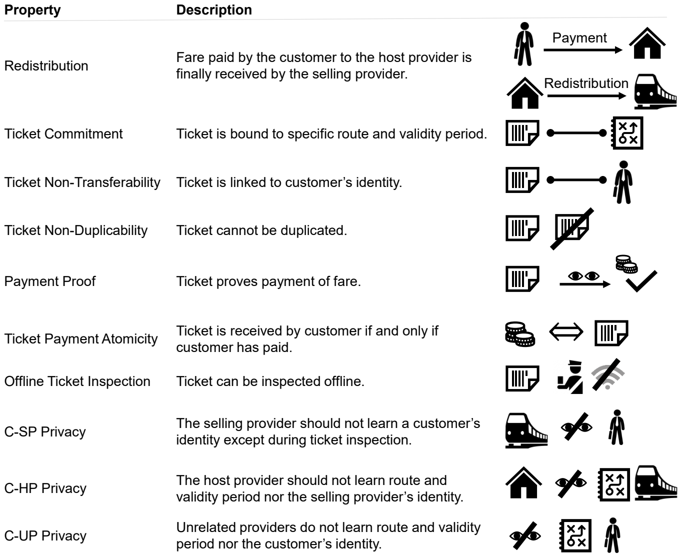
II. Scenario and Problem Statement

In our scenario, there are multiple mobility *providers* each providing mobility services on specific routes for public transport. *Customers* use the public transport to travel. To simplify our scenario, we assume that a customer uses for each trip only the means of transport offered by exactly one provider. We leave the development of a protocol for trips using multiple providers for future work (see Section VI). A customer needs to buy a ticket before traveling to pay the fare for the trip. The offered transport routes and the availability of different fares is learned by a customer through an abstract third party that we call the *query service*. This can, for example, be a platform for open data [17], [18], an interface offered by some public transport providers themselves, an off-line database on a customer's device, or a combination of multiple query services.

We assume that customers travel regularly with different providers. To pay for the ticket, a customer has to be registered and needs to have configured a means of payment. To improve usability and bootstrapping for new providers, customers should not be required to register at each provider they want to travel with. Instead, a customer who is registered at one provider, the customer's *host provider*, can buy tickets from a different *selling provider* serving the trip's route.

For the ticket purchase and fare allocation, the following properties should hold (see Fig. 1): The customer pays to the host provider and, following the ticket sale, the payment needs to be redistributed between the providers. A fare paid by the customer to the host provider must finally be received by the selling provider (*Redistribution*). A customer has to buy a ticket before the customer starts to travel. A ticket is bound to a specific route and validity period (*Ticket Commitment*). A ticket should not be transferable between different customers and, thus, a ticket is linked to one customer's identity (*Ticket Non-Transferability*). A ticket serves as proof that the customer has paid the ticket's price to the selling provider (*Payment Proof*). A ticket cannot be duplicated (*Ticket Non-Duplicability*). In conjunction with the previous property, this means that in exchange for one payment only one ticket can be obtained. To protect both parties during a ticket sale from (accidental or deliberate) faults that result in the customer receiving a ticket without having paid or the selling provider receiving a payment while the customer does not receive a ticket, the ticket payment should be atomic. Hence, a customer receives a ticket if and only if the customer pays the ticket's price to the selling provider (*Ticket Payment Atomicity*).

A customer starts the trip by entering a vehicle. An inspector employed by the selling provider might verify the existence and the validity of a traveling customer's ticket. This ticket inspection should only require communication between the traveling customer and the inspector, i.e. the ticket inspection should not require the inspector to communicate with a back-end to verify the validity of tickets (*Offline Ticket Inspection*). Instead of personnel on a vehicle, ticket inspection could also be performed by a security gate at a station.


Payment
Redistribution

Property	Description
Redistribution	Fare paid by the customer to the host provider is finally received by the selling provider.
Ticket Commitment	Ticket is bound to specific route and validity period.
Ticket Non-Transferability	Ticket is linked to customer's identity.
Ticket Non-Duplicability	Ticket cannot be duplicated.
Payment Proof	Ticket proves payment of fare.
Ticket Payment Atomicity	Ticket is received by customer if and only if customer has paid.
Offline Ticket Inspection	Ticket can be inspected offline.
C-SP Privacy	The selling provider should not learn a customer's identity except during ticket inspection.
C-HP Privacy	The host provider should not learn route and validity period nor the selling provider's identity.
C-UP Privacy	Unrelated providers do not learn route and validity period nor the customer's identity.

Fig. 1. Overview of desired properties regarding a ticket, payment and redistribution, and privacy.

To protect a customer's privacy, the selling provider should not learn any information that identifies the customer except if the selling provider is the customer's host provider or during ticket inspection (Customer – Service Provider Privacy, *C-SP Privacy*). While a customer's host provider knows a customer's identity, the host provider should only learn purchase time and price about a ticket a customer buys but no other details such as the selling provider's identity or the booked route and validity period (Customer – Host Provider Privacy, *C-HP Privacy*). Any other provider should neither learn a ticket's details nor a customer's identity (Customer – Unrelated Provider Privacy, *C-UP Privacy*). While previous work [19]–[21] has proposed ticketing systems that consider privacy more comprehensively, our problem statement includes ticket sale *as well as* revenue allocation, and the PCN-based approach presented below is decentralized.

III. Background on Building Blocks

A. Payment Channel Networks

Two parties create a payment channel by depositing funds into a shared account. Both store the allocation of who owns which share of the funds. The payment channel protocol ensures that each party can close the shared account at any time independent from the other party and both parties receive their correct share of the funds. The two parties in a payment channel can perform transactions by updating the allocation of funds in the channel, i.e., transactions inside a payment channel require only interaction between the two parties and no other third party. Such payment channels can be built over different payment layers such as cryptocurrencies or central bank digital currencies (CBDC) [22]. Implementations of payment channels are for example the Lightning Network [13] on top of Bitcoin and the Raiden Network on top of Ethereum. Multiple payment channels can be linked to create a payment channel network (PCN). In a PCN, two parties can perform transactions even if they do not have a common channel. For such a multi-hop transaction, the participants of the PCN forward a payment from the sender to the final receiver while the PCN's protocol must ensure that each intermediary neither looses nor steals funds, i.e., each intermediary receives funds from the previous hop if and only if the intermediary sends the funds to the next hop on the payment route.

A multi-hop payment is secured using Hash Timelocked Contracts (HTLCs) [13]. An HTLC is a contract between two parties that one party pays the other party a specified amount if and only if the receiving party presents a preimage to a hash value before a given time. In a PCN, HTLCs are chained over multiple hops so that the first hop effectively pays the last hop if and only if the last hop reveals a preimage to the given hash value. Each intermediate hop forwards the payment as well as the preimage and the setup of HTLCs ensures that an intermediate hop can neither steal nor loose funds.

B. Adaptor Signatures

While payment channel networks offer with HTLCs a way to 'buy a preimage', we need in our protocol the feature that a customer buys a signature for a certain message from a seller. To implement this feature using PCNs, we use adaptor signatures [23] as a building block for the protocol. Adaptor signatures are constructed from a digital signature scheme and add additional algorithms to create a pre-signature for a message which can later be adapted using a secret that is linked to the pre-signature. More precisley, an adaptor signature consists of the following algorithms: $\mathrm{pSign}(m, Y, sk)$ creates a pre-signature $\tilde{\sigma}$ for message m and hash Y and private key sk. $\mathrm{pVrfy}(\tilde{\sigma}, m, Y, pk)$ verifies that $\tilde{\sigma}$ is a valid pre-signature for m for public key pk which can be adapted to a signature using the preimage y of $Y = H(y)$ with a hash function H. $\mathrm{Adapt}(\tilde{\sigma}, y)$ creates a signature σ from the pre-signature $\tilde{\sigma}$. While the definition of adaptor signatures also contains an extract algorithm, we omit the definition because we do not use this algorithm in our protocol. An adaptor signature provides the security guarantee that an adversary knowing Y and $\tilde{\sigma}$ for a message m can create a valid signature σ for m only with negligible probability (*Existential Unforgeability*). Constructions of adaptor signatures exist based on ECDSA signatures and based on Schnorr signatures [23].

C. Commitment Scheme

A cryptographic commitment scheme is a two party protocol used for one party to commit to a value x using the function $\mathrm{Com}(x)$ and later reveal the committed value with the following properties: The other party is not able to extract information about the value from the commitment (*Hiding*). The committing party can only reveal the value that the party committed to (*Binding*). Various practical commitment schemes have been proposed based on different assumptions such as Collision-Free Hash Functions [24].

IV. Protocol for Payment and Clearing

For the problem statement described in Section II, we present a protocol that is based on PCNs. The protocol assumes that the public transport providers are part of a PCN which customers use to make payments to their host providers and payments are redistributed to the selling provider. To create the PCN, the providers open payment channels to other providers by locking funds on an underlying layer, e.g., a blockchain. The providers exchange information about the topology of the PCN with the query service so that each provider as well as the query service knows the network's topology. Customers become part of the network by opening a payment channel to their host provider during registration.

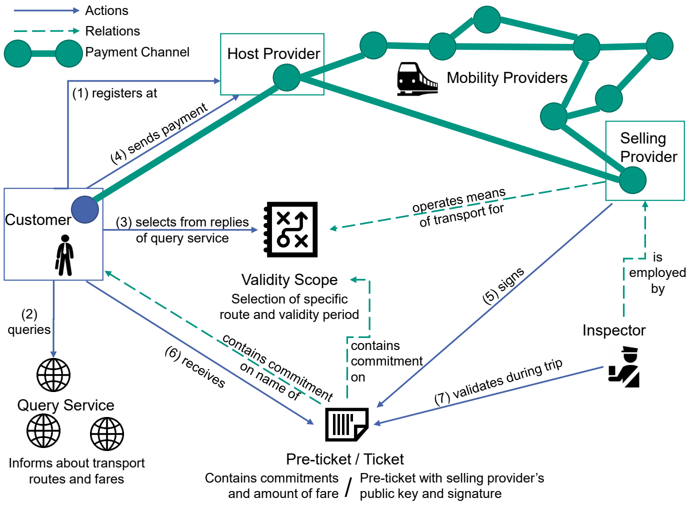

Actions
Relations
Payment Channel
Host Provider
Mobility Providers
Selling Provider
(1) registers at
(4) sends payment
Customer
(3) selects from replies of query service
operates means of transport for
Validity Scope
Selection of specific route and validity period
is employed by
Inspector
(5) signs
(2) queries
contains commitment on name of
(6) receives
contains commitment on
(7) validates during trip
Query Service
Informs about transport routes and fares
Pre-ticket / Ticket
Contains commitments and amount of fare
Pre-ticket with selling provider's public key and signature

Fig. 2. Protocol overview. A customer queries available offers for a route from a query service. The customer selects an offer to create a validity scope for the ticket that contains the selected route and time of travel. The means of transport for this trip are operated by the selling provider. The customer creates a pre-ticket for the trip that becomes a valid ticket once it is signed by the selling provider. The (pre)ticket contains commitments on the customer's name and on the validity scope and the amount of the fare. The customer pays the fare to the host provider that the customer is registered at. During the trip, an inspector validates the customer's ticket by verifying the commitments as well as their contents and the matching fare.

A. Protocol Overview

Before initiating a purchase, the customer registers at a host provider. The host provider takes the role of a payment provider for the customer. From a query service, the customer selects a ticket offer (see Fig. 2). Based on the ticket offer, the customer generates a *pre-ticket* containing commitments on the ticket's validity scope and the customer's identity. For the pre-ticket to become a valid ticket, the customer needs the selling provider's signature on the pre-ticket. We implement an atomic exchange of payment and the selling provider's signature on the pre-ticket by using adaptor signatures and HTLCs: The customer sends the pre-ticket to the seller. The seller draws a random secret y and creates a pre-signature that can be used to adapt the pre-signature to a valid signature. The selling provider sends the pre-signature on the pre-ticket to the customer. The customer obtains the secret y by performing a payment through the PCN in exchange for the secret required to adapt the pre-signature. The customer uses the secret y to create a signature from the pre-signature. The signature completes the pre-ticket so that the pre-ticket becomes a ticket. During the trip, the customer presents the ticket to the inspector and opens the commitments for the inspector. The inspector verifies the commitments and the seller's signature on the ticket.

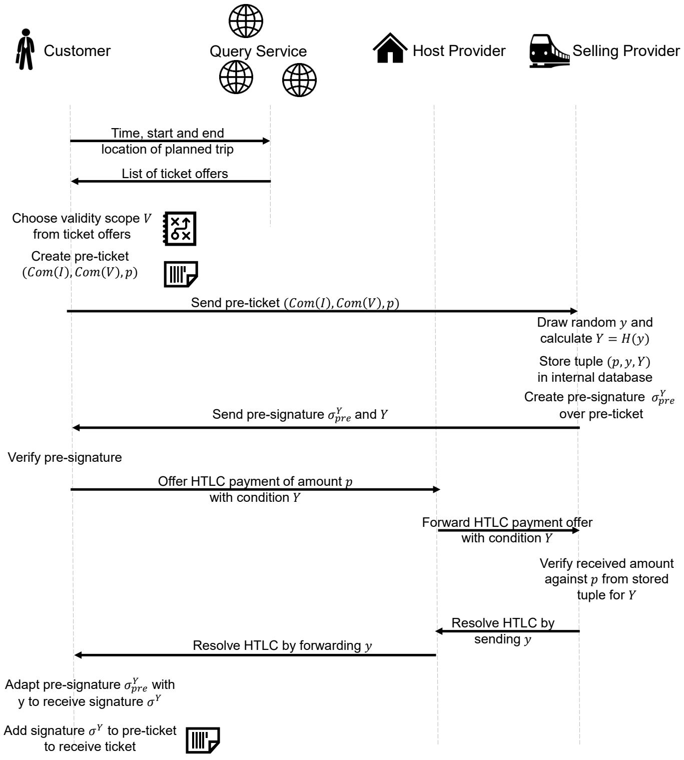
B. Registration

Each provider who is willing to accept new registrations announces this publicly and accepts new incoming payment channels. A customer registers at a host provider by opening a payment channel to the provider. The funds initially deposited into the payment channel can be used to pay for tickets.

C. Ticket Purchase Protocol

The steps needed to purchase a ticket can be seen in Fig. 3. To purchase a ticket, the customer first retrieves routing information and ticket offerings from the query service. We consider the implementation of the query service out-of-scope and specify only the interface. The customer sends the time, starting location, and ending location of the planned trip to the query service. The query service returns a list of ticket offers each containing the seller's identity S, the ticket's price p, and the validity scope V containing the trip's route and the time window of ticket validity. The customer chooses an offer from the list of available ticket offers. In addition to ticket data, the query service also communicates the routing information for the PCN to the customer.

The customer generates a pre-ticket $(Com(I), Com(V), p)$ containing a commitment to the customer's name I, a commitment to the validity scope V, and the ticket's price p and sends it over an encrypted communication channel to the seller. The seller draws a random value y and calculates $Y = H(y)$ using a hash function H. The seller stores the payment price p, the secret y, and the hash Y that serves as an identifier of the pre-ticket in an internal database. The seller creates a pre-signature σ^Y_{pre} over the pre-ticket $(Com(I), Com(V), p)$ and sends the pre-signature σ^Y_{pre} and Y to the customer. The customer verifies that the pre-signature matches the customer's request $(Com(I), Com(V), p)$. As the customer has knowledge of the PCN's topology, the customer can find a payment route to the selling provider identified by S. The customer offers their host provider an HTLC with the ticket's price and the challenge for a preimage for Y as condition and includes the onion-encrypted information for the remaining hops along the route to the seller. When the seller receives the HTLC from one of their neighbors in the PCN, the seller verifies that the price matches the expected price stored for Y and resolves the HTLC by sending the preimage y to the seller's neighbor who forwards it along the payment route to the customer resolving each HTLC. When the customer has received the preimage y, the customer uses y to adapt the pre-signature σ^Y_{pre} for the pre-ticket and receives a signature σ^Y for $(Com(I), Com(V), p)$. The tuple $(Com(I), Com(V), p)$ together with the seller's public key and the signature σ^Y constitutes a valid ticket.



Customer
Query Service
Host Provider
Selling Provider
Time, start and end location of planned trip
List of ticket offers
Choose validity scope V from ticket offers
Create pre-ticket $(Com(I), Com(V), p)$
Send pre-ticket $(Com(I), Com(V), p)$
Draw random y and calculate $Y = H(y)$
Store tuple (p, y, Y) in internal database
Create pre-signature σ^Y_{pre} over pre-ticket
Send pre-signature σ^Y_{pre} and Y
Verify pre-signature
Offer HTLC payment of amount p with condition Y
Forward HTLC payment offer with condition Y
Verify received amount against p from stored tuple for Y
Resolve HTLC by sending y
Resolve HTLC by forwarding y
Adapt pre-signature σ^Y_{pre} with y to receive signature σ^Y
Add signature σ^Y to pre-ticket to receive ticket

Fig. 3. Protocol for ticket purchase. The customer queries the query service for ticket offers and selects the validity scope from one of the offers. The customer sends the pre-ticket to the selling provider who creates a pre-signature. The customer pays for the ticket over the customer's host provider using the PCN and receives the secret y required to adapt the pre-signature. After adaption, the customer has a valid signature for the pre-ticket which together with the signature represents a valid ticket.

D. Inspection

Ticket inspection can be done either when the customer enters the vehicle or in the course of the trip. The inspector is employed by the selling provider and has the seller's public key. For inspection, the customer sends the ticket $(Com(I), Com(V), p)$ and the signature σ^Y to the inspector and reveals the commitments. Communication can happen, for example, using NFC or visually using a barcode. The inspector verifies the validity of the commitments and verifies with the seller's public key that σ^Y is a valid signature for the ticket. To achieve Ticket Non-Transferability, the customer must reveal their identity and the inspector must verify that the customer can identify as the identity that the ticket is committed to.

V. Evaluation of Security and Privacy Properties

The protocol achieves the required properties as presented in Section II. Table I summarizes our results by showing the main concepts that are used to achieve each property.

To reach the *C-SP Privacy*, we assume that, when the customer exchanges pre-ticket and pre-signature with the service provider, the communication is done over a communication channel that hides the customer's identity. The communication could be performed over the selling provider's website which

TABLE I
SECURITY AND PRIVACY PROPERTIES ACHIEVED BY THE PROTOCOL

Property	Achieved through
Ticket Commitment	Binding property of commitment for validity scope, verified by ticket inspector
Ticket Non-Transferability	Binding property of commitment for customer's identity, verified by ticket inspector
Ticket Non-Duplicability	Combination of Ticket Commitment and Ticket Non-Transferability
Payment Proof	Signature on ticket
Ticket Payment Atomicity	HTLCs and Existential Unforgeability of adaptor signature
Offline Ticket Inspection	Inspection only requires verification of commitment and signature
Redistribution	Multi-hop payment
C-SP Privacy	Anonymous communication channel, Hiding property of commitment for I and V
C-HP Privacy	Host provider does not receive (pre-)ticket
C-UP Privacy	Unrelated providers do not receive (pre-)ticket and do not interact with customer

might in practice preserve the customer's privacy under the assumption that the selling provider cannot identify the customer by the customer's IP address, browser, or another side-channel. An alternative is the use of an anonymous communication system (e.g., Tor [25], Nym [26]).

Depending on the fare system, the host provider or any other provider on the route of the payment in the PCN might be able infer some information from the ticket's price (e.g., the approximate length of the trip), but neither start nor end of the booked route nor the specific time of travel is disclosed. If a specific price was only offered by one specific provider, the price could leak the identity of the selling provider. Thus, this leak of information should be prevented by the fare system.

As the selling provider accepts any pre-tickets and stores them in a database, this opens an attack surface for Denial-of-Service attacks in which an adversary requests many pre-tickets without intending to actually buy a ticket. This problem is enforced by the fact that the selling provider does not know the potential customer's identity. An approach to restrict this attack surface is to introduce a timeout for pre-tickets so that pre-tickets are only stored by the selling provider for a certain time and must be paid during this time window. Also, the selling provider can monitor the number of unresolved tickets and perform rate-limiting based on out-of-protocol knowledge (e.g., IP address, location) of the potential customer.

VI. Design Options and Protocol Extensions

The protocol presented above represents *one* option to build a PCN-based approach for redistribution of payments. Before we draw conclusions on to what extent PCNs are beneficial for such approaches, we discuss in this section some design options and possible extensions to the protocol to give a broader view for the following discussion instead of being focused on just one specific protocol.

Payment Base Layer. A PCN is a second payment layer to a first payment layer which is commonly a blockchain. Implementations of PCNs exist for example for Bitcoin and Ethereum. However, the first layer can also be instantiated differently, for example with banks [22], which allows the parties of the PCN to use Central Bank Digital Currencies (CBDCs) instead of cryptocurrencies. This plays an important role for deployability as providers might want to rely on CBDCs for payments. It is even possible that individual payment channels of the PCN use different base layers, however, if different currencies are used within one payment path, the corresponding parties need to agree on the exchange rates. Two providers in the PCN might maintain multiple payment channels between themselves based on different base layers to provide a wider range of currencies to their customers.

An even further going option would be for two providers to not use a first layer at all. This concept is known from credit networks which, as opposed to PCNs, do not require a first layer but require trust between the network's peers. Two providers might agree on a certain amount they are willing to give credit to the other party which would replace a payment channel by an "I owe you" link of a credit network that can be used for multi-hop payments just like a payment channel.

Payment Channel Network. The PCN used between the providers might be a dedicated PCN used only for this task or an existing PCN could be used. While a dedicated PCN could be customized and independently developed, an existing PCN could provide lower cost during setup and operation.

Customer Attributes. A typical additional requirement for ticket systems is that different fares should be available for certain groups such as students or retirees. This can be achieved by extending the protocol presented above with customer attributes. The host provider can assume the additional role of an attribute provider. During registration, the host verifies attributes of the customer which can be used by the customer to acquire tickets at a reduced price. Then, the pre-ticket would also include the customer's attributes signed by the host provider and the host provider's public key. Upon reception of the pre-ticket, the seller verifies that the customer's attributes are signed by a known host provider. Alternatively, the seller could accept a ticket with any customer attributes and during inspection the traveling customer has to prove that the customer is eligible for using these attributes (e.g., by showing an identity card). Both approaches could be combined so that the inspector is notified during inspection whether attributes have already been verified or whether a verification is required.

Check-In- / Check-Out-Paradigm. The protocol above follows the paradigm that a customer buys a ticket before starting the trip and presents the ticket during the trip to an inspector. A different but also common paradigm is the Check-In- / Check-Out-Paradigm in which a customer checks in at a security gate at one station, travels to another station, and checks out at a gate at the destination station. During check-out, the distance and time used for traveling is calculated and the respective fare deducted from the customer's account. The protocol above can be adjusted to follow this paradigm. The ticket purchase protocol would be performed during check-out with the data for the pre-ticket being determined by the gate of the destination station instead of the customer committing to data returned from the query service. The entering gate needs to be either stored on the customer's device or a mechanism to store it in another decentralized fashion would be required.

Validity Scope. While in public transport a ticket's scope is usually specified by the booked route and the validity period, this is different for an application in rental of e-scooters or bikes. In such a use-case, the validity scope could also include a specific vehicle identifier that identifies the scooter or bike that a customer has booked.

Tradeoff between Privacy and Ticket Non-Transferability. The desired property of Ticket Non-Transferability requires tickets to be linked to a customer's identity and, thus, this link to be verified by the inspector. Hence, this property conflicts with a stricter privacy property that would hide the customer's identity completely from the inspector. If a ticket was allowed to be transferable, the ticket could be independent from a customer's identity and the inspection would not reveal the customer's identity.

Multi-Provider Tickets. A typical use case in integrated public transport is that public transport providers offer joint tickets with other providers so that a customer can buy one ticket for a route that is partially offered by one provider and partially by another provider. As the ticket is sold by only one provider, the ticket revenue needs to be split between the providers according to the customer's route. A PCN-based approach might offer the ability to buy a ticket from one provider but pay the ticket's price partially to multiple provider which are on the same payment route. We leave it to future work to design and specify a system for this use case.

PTLC instead of HTLC. For buying a ticket, the above protocol requires the exchange of a payment and a signature. However, an HTLC is a contract for an atomic exchange of a payment and a preimage. To build an atomic exchange of a payment and a signature with HTLCs, we use adaptor signatures. By the activation of the Taproot update, it has recently become possible to implement PTLCs (Point Timelocked Contracts) which can be used to directly sell signatures[1]. While the implementation of approaches for PTLCs is still under discussion, the combination of HTLCs and adaptor signatures can be used with deployed PCNs. Once PTLCs are deployed, the above protocol can be adjusted to work with PTLCs.

VII. Benefits and Limitations

Decentralization. A benefit of a PCN-based approach as presented above is that it does not require a central clearing party. This decentralization increases the providers' independence and saves them the operational cost for the clearing party. However, the decentralization comes with a higher cost of communication and agreement with the other providers. While each pair of providers is independent in the decision of how payment channels are managed, all providers need to maintain a common protocol for multi-hop payments.

Transaction Fees. PCNs are a second layer payment service that processes transactions without requiring communication with the first payment layer. Hence, a payment service based on PCNs requires fewer transactions at the first layer which reduces transaction fees compared to an approach that requires a transaction for each ticket purchase and for redistribution.

Atomicity. PCN protocols implement atomicity for multi-hop payments. As shown by the protocol above, this atomicity can be extended to reach atomicity for the exchange of payment and ticket to fulfill the *Ticket Payment Atomicity* property which reduces the required trust between customer and selling provider.

Payment Finality. Payments over a PCN have instant finality by design. Once a payment has been performed, the payment cannot be reverted. This property reduces the risk for providers because it ensures that a provider will keep the payment for a ticket sold. On the other side, PCNs do not offer a native way to handle ticket refunds. A refund for a ticket can be handled like a regular payment in reversed direction; however, the customer's identity must be disclosed to the selling provider to conduct the refund.

Implicit Fare Clearing. The protocol presented above shows that a benefit of a PCN-based approach is that the redistribution of ticket revenues does not need to be handled explicitly. Instead, each payment is directly forwarded to the selling provider and no clearing is required at a later time. This reduces the trust required between providers because they do not depend on other providers' future ability to pay.

Privacy. The privacy analysis shows that the protocol can achieve privacy properties even for the payment. This characteristic is achieved by uncoupling the payment from the ticket issuance. The selling provider signs a ticket but payment is performed through a PCN without direct communication between the customer and the selling provider. During payment, the protocol hides information about the booked route because the payment is only linked to the identifier Y.

Locked Funds. The use of PCNs requires that funds are locked in payment channels. This requirement affects the customers because they have to top up their account at their host provider before being able to purchase tickets. While customers have the guarantee that they do not loose their money by locking the money inside a payment channel, the money is locked capital that cannot be spent otherwise. Providers have to deposit an even higher amount of money inside payment channels and bear the cost for locked capital.

Liveness Requirement of PCNs. Protocols for payment channels assume that the parties watch the first payment layer so that they are notified if the counterparty closes the channel. This task needs to be fulfilled by the providers as well as the customers that have a payment channel. While there are ways to outsource this task to a third party, called watchtower, protocols for watchtowers [27]–[29] come again with their own assumptions and their deployability depends on the specific use case.

Rebalancing. If a majority of payments are performed in the same direction, the payment channels might become unbalanced, i.e., most funds in a channel are owned by one party. In an unbalanced PCN, multi-hop payments might fail because payments can only be routed into the direction from

[1] https://suredbits.com/payment-points-part-4-selling-signatures/

the party owning the funds to the other party. This problem can be approached by rebalancing a channel by performing a transaction on the first payment layer that distributes the funds so that the channel is balanced. For pairs of providers that have a balanced payment flow, PCNs reduce the amount of first payment layer transactions to a minimum of an opening and a closing transaction.

Subsidy Allocation. A further limitation of the protocol is that is provides revenue allocation only for ticket fares. Frequently, public transport is supported by subsidies. For the allocation of subsidies, an additional mechanism is required.

VIII. Conclusion

Revenue allocation is a problem that arises in the public transport domain in case providers offer customers the ability to buy tickets from one provider but pay the ticket's price to another provider. We have shown that the presented protocol that is based on PCNs fulfills the postulated properties of atomicity and redistribution. Consequently, we can affirmatively answer our initial research question of the applicability of PCNs for allocation of ticket revenues. We analyzed the benefits and limitations that the use of PCNs has for revenue allocation and found that under certain assumptions the protocol effectively provides also the other desired and required properties.

References

[1] W. Homburger and V. Vuchic, "Federation of Transit Agencies as a Solution for Service Integration," *Traffic Quarterly*, pp. 373–391, Jul. 1970.

[2] D. B. Rinks, "Revenue allocation methods for integrated transit systems," *Transportation Research Part A: General*, vol. 20, no. 1, pp. 39–50, Jan. 1986.

[3] Transportation Research Board and National Academies of Sciences, Engineering, and Medicine, *Multimodal Fare Payment Integration*, R. L. Jeroen Kok, IMG Rebel Inc., Ed. Washington, DC: The National Academies Press, 2020.

[4] ——, *The Role of Transit, Shared Modes, and Public Policy in the New Mobility Landscape*. Washington, DC: The National Academies Press, 2021.

[5] D. A. Tsamboulas and C. Antoniou, "Allocating Revenues to Public Transit Operators under an Integrated Fare System," *Transportation Research Record*, vol. 1986, no. 1, pp. 29–37, Jan. 2006.

[6] R. Huang, Y. Jiang, Z. Liu, Y. Yang, and W. Jiang, "Algorithm and Implementation of Urban Rail Transit Network Based on Joint Operation," *Journal of Transportation Systems Engineering and Information Technology*, vol. 10, no. 2, pp. 130–135, Apr. 2010.

[7] P. Yichao, "Verification and Optimization of Metro Fare Clearing Models Based on Travel Route Reconstruction," *New Metro*, vol. 1, no. 1, pp. 34–47, Dec. 2020.

[8] S. Gao and Z. Wu, "Modeling Passenger Flow Distribution Based on Travel Time of Urban Rail Transit," *Journal of Transportation Systems Engineering and Information Technology*, vol. 11, no. 6, pp. 124–130, Dec. 2011.

[9] F. Zhou, J.-g. Shi, and R.-h. Xu, "Estimation Method of Path-Selecting Proportion for Urban Rail Transit Based on AFC Data," *Mathematical Problems in Engineering*, vol. 2015, Sep. 2015.

[10] H. Xiao, L. Sun, S. Kong, G. Gong, and F. Zhang, "Passenger Travel Path Estimation Algorithm Based on High Accuracy Location Data," in *2017 Fifth International Conference on Advanced Cloud and Big Data (CBD)*, Aug. 2017, pp. 256–260.

[11] G. Cheng, S. Zhao, and S. Xu, "Estimation of passenger route choices for urban rail transit system based on automatic fare collection mined data," *Transactions of the Institute of Measurement and Control*, vol. 41, no. 11, pp. 3092–3102, Jul. 2019.

[12] G. Tuveri, M. Garau, E. Sottile, L. Pintor, L. Atzori, and I. Meloni, "Beep4Me: Automatic Ticket Validation to Support Fare Clearing and Service Planning," *Sensors*, vol. 22, no. 4, p. 1543, Jan. 2022.

[13] J. Poon and T. Dryja, "The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments," Tech. Rep., 2016.

[14] E. Erdin, M. Cebe, K. Akkaya, S. Solak, E. Bulut, and S. Uluagac, "Building a Private Bitcoin-Based Payment Network Among Electric Vehicles and Charging Stations," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, Jul. 2018, pp. 1609–1615.

[15] M. Zichichi, S. Ferretti, and G. D'Angelo, "MOVO: a dApp for DLT-based Smart Mobility," in *2021 International Conference on Computer Communications and Networks (ICCCN)*, Jul. 2021, pp. 1–6, iSSN: 2637-9430.

[16] B. Xiao, X. Fan, S. Gao, and W. Cai, "EdgeToll: A Blockchain-based Toll Collection System for Public Sharing of Heterogeneous Edges," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Apr. 2019, pp. 1–6.

[17] A. Ojo, E. Curry, and F. A. Zeleti, "A Tale of Open Data Innovations in Five Smart Cities," in *2015 48th Hawaii International Conference on System Sciences*, Jan. 2015, pp. 2326–2335, iSSN: 1530-1605.

[18] Y. Parcianello, N. P. Kozievitch, K. V. O. Fonseca, M. d. O. Rosa, T. M. C. Gadda, and F. C. Malucelli, "Transportation: An Overview from Open Data Approach," in *2018 IEEE International Smart Cities Conference (ISC2)*, Sep. 2018, pp. 1–8.

[19] I. Gudymenko, "Privacy-preserving E-ticketing Systems for Public Transport Based on RFID/NFC Technologies," Ph.D. dissertation, TU Dresden, May 2015.

[20] G. Hinterwalder, "Privacy-preserving Payments for Transportation Systems," Ph.D. dissertation, University of Massachusetts Amherst, Nov. 2015.

[21] J. Han, L. Chen, S. Schneider, H. Treharne, and S. Wesemeyer, "Privacy-Preserving Electronic Ticket Scheme with Attribute-Based Credentials," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 4, pp. 1836–1849, Jul. 2021.

[22] M. Grundmann and H. Hartenstein, "Fundamental Properties of the Layer Below a Payment Channel Network," in *Data Privacy Management, Cryptocurrencies and Blockchain Technology*, ser. Lecture Notes in Computer Science, J. Garcia-Alfaro, G. Navarro-Arribas, and J. Herrera-Joancomarti, Eds. Cham: Springer International Publishing, 2020, pp. 409–420.

[23] L. Aumayr, O. Ersoy, A. Erwig, S. Faust, K. Hostáková, M. Maffei, P. Moreno-Sanchez, and S. Riahi, "Generalized Channels from Limited Blockchain Scripts and Adaptor Signatures," in *Advances in Cryptology – ASIACRYPT 2021*, ser. Lecture Notes in Computer Science, M. Tibouchi and H. Wang, Eds. Cham: Springer International Publishing, 2021, pp. 635–664.

[24] S. Halevi and S. Micali, "Practical and Provably-Secure Commitment Schemes from Collision-Free Hashing," in *Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '96. Springer-Verlag, Aug. 1996, pp. 201–215.

[25] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The Second-Generation Onion Router," in *Proceedings of the 13th USENIX Security Symposium*, 2004.

[26] C. Diaz, H. Halpin, and A. Kiayias, "The Nym Network," Tech. Rep. https://nymtech.net/nym-whitepaper.pdf, Feb. 2021.

[27] P. McCorry, S. Bakshi, I. Bentov, S. Meiklejohn, and A. Miller, "Pisa: Arbitration Outsourcing for State Channels," in *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, ser. AFT '19. New York, NY, USA: ACM, Oct. 2019, pp. 16–30.

[28] M. Leinweber, M. Grundmann, L. Schönborn, and H. Hartenstein, "TEE-Based Distributed Watchtowers for Fraud Protection in the Lightning Network," in *Data Privacy Management, Cryptocurrencies and Blockchain Technology*, ser. Lecture Notes in Computer Science, C. Pérez-Solà, G. Navarro-Arribas, A. Biryukov, and J. Garcia-Alfaro, Eds., vol. 11737. Springer International Publishing, Sep. 2019, pp. 177–194.

[29] M. Khabbazian, T. Nadahalli, and R. Wattenhofer, "Outpost: A Responsive Lightweight Watchtower," in *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, ser. AFT '19. New York, NY, USA: ACM, Oct. 2019, pp. 31–40.