Congestion pricing that respects "driver privacy"

Andrew J. Blumberg
Stanford University
Mathematics Department
blumberg@math.stanford.edu

Robin Chase
Meadow Networks
robin@meadownetworks.com

***Abstract*— In 2003, the city of London implemented a congestion pricing policy in order to reduce traffic and raise revenues for transit improvements. The dramatic success of this system has led to widespread consideration of the adoption of such variable tolling, including road pricing, in dense urban cores around the world. While from many perspectives the broad implementation of such congestion pricing systems would be socially beneficial, the likely consequences for the privacy of motorists are extremely negative. A sophisticated congestion pricing strategy will assign a cost to a specific space-time path of a vehicle through the pricing zone. Straightforward implementations of monitoring systems to assess congestion tolls thus require detailed tracking technology to monitor the paths of each individual vehicle.**

In this paper, we introduce a novel protocol for computing congestion pricing tolls in a fashion that preserves driver privacy. Our scheme uses cryptographic algorithms to guarantee that the state can collect arbitrarily nuanced congestion pricing tolls without being able to track the movements of individual drivers. That is, the system provides simultaneous guarantees that the state can correctly compute the tolls for a particular driver from the information it collects but that the state cannot reconstruct the path of the driver no matter what it does with this information.

Our system is built using a variant of the protocol we described in a previous paper to handle automated traffic enforcement (i.e. stop-light violation detection) in a way that preserves driver privacy and eliminates camera use. The protocol is relatively easy to implement with existing technology, and such implementation can be done in a fashion which is sufficiently robust to handle realistic operational requirements. In particular, we discuss methods for ensuring resistance to attempts to cheat and modifications to handle sporadic users (tourists).

I. Introduction

As a consequence of the success of the congestion pricing system in the city of London, many municipalities are considering the adoption of such variable usage pricing for their dense urban cores. It is hoped that such tolling will decrease traffic, shift travelers to more sustainable transportation modes and generate much-needed revenue to support the transit infrastructure. In January 2006, Stockholm undertook a six-month trial implementation of such a system and local governments in many U.S. cities including New York, Boston, and San Francisco are considering implementing some form of congestion pricing. Moreover, there has recently been serious discussion of the introduction of an integrated system of nuanced road usage charges on highways throughout Europe.

Steady increases in vehicular traffic on relatively fixed urban road infrastructure have caused the annual traffic delay per traveler to triple in major cities in the United States between 1982-2003 [1]. Given the greater associated costs of traffic congestion, the gathering concern about urban sprawl, and the lack of successful policy alternatives, it seems highly likely that there will be broad deployment of congestion pricing systems in the next decade. From many perspectives, the widespread adoption of such systems is a very welcome development. In particular, nuanced usage tolls which accurately charge drivers for the higher value of driving on scarce road resources during peak periods allow fine-tuned incentives to promote the reduction of non-essential driving, car-pooling, and time or mode shifts, leading to a variety of societally beneficial outcomes.

Unfortunately, from the perspective of driver privacy there are some extremely worrisome consequences of the adoption of such highly nuanced usage charge systems. A sophisticated pricing strategy will assign tolls as a function of the specific space-time path of a vehicle through the congestion pricing zone. That is, to compute the cost of a particular trip one must know the entire path for the trip. As a consequence, straightforward designs for implementing such congestion pricing systems require detailed tracking of individual drivers. For example, the less-sophisticated systems in use in London and Stockholm use extensive networks of cameras to identify and charge vehicles entering the congestion pricing zone. Despite well-meaning intentions and promises to discard data immediately, once cameras are installed and the technological capacity is in place, such systems can provide governments with tempting opportunities for the real-time tracking of citizens' movements. History suggests that assurances by government entities that such information will be used responsibly cannot be trusted over the long-term. It seems preferable to develop and implement a system that does not offer such temptation by simply not facilitating vehicle tracking. Such systems will also have much greater public acceptance than the alternatives.

In this paper we present a system which supports the implementation of arbitrarily sophisticated congestion pricing schemes while preserving driver privacy. Although there has been substantial previous work on anonymous electronic toll collection [2], existing protocols are better-suited to traditional pricing schemes. While we have taken pains to attempt to ensure that our system is reasonable for actual implementation, we are not necessarily convinced nor arguing that this is the best possible design. Rather, we simply contend that our system is a sufficiently plausible alternative so as to shift the nature of the debate — it is possible to have

sophisticated congestion pricing without surrendering driver privacy.

We have adapted ideas developed in previous work for automated traffic enforcement systems which preserve driver privacy [3]. The key ingredient is a cryptographic protocol for secure multi-party shared computation. Such a protocol allows mutually untrusting parties to compute functions of private information without revealing the data. Specifically, in a two-party shared computation, there are two individuals A and B who wish to compute a function f. The function f depends on two variables x and y, and we will assume that A possesses the argument x and B possesses the argument y. Certainly a naive way to compute the value $f(x, y)$ is for A and B to share their private information (the values of x and y). But via the two-party shared computation algorithm, A and B can compute $f(x, y)$ in such a way that neither A nor B learn anything about the others' private data except what could be inferred from the value of $f(x, y)$. Thus, neither A nor B needs to reveal their private information.

We will now outline our congestion pricing protocol. Every car is assumed to have a radio-frequency transponder (akin to an EZ-pass or related automatic tolling technology), and the congestion pricing zone is presumed to have an arbitarily dense distribution of monitoring devices that interact with the transponders. We will describe the interaction for a driver Dennis and the tolling agency, which we will refer to as the DMV.

1) At the beginning of the year, Dennis privately chooses a lengthy sequence of "dynamic license plates". This is just a long list of very large numbers (chosen in such a way so as to minimize the probability of overlap with any other driver). Dennis digitally signs the list of numbers, and gives the signature but not the list to the DMV.
2) As Dennis drives, the transponder in his car rapidly cycles through the list of dynamic license plates, at the rate of a new number each second.
3) When Dennis enters the congestion pricing zone, as he drives past monitoring devices, the devices record his current dynamic license plate number.
4) At the end of the a predetermined billing period, Dennis returns to the DMV to renew his registration.
    a) The DMV has a long list of numbers collected from drivers in the congestion pricing zone.
    b) Dennis has a long list of dynamic license plates.
    c) Dennis and the DMV engage in secure two-party computation of the charges Dennis owes the DMV.

    At the end of this computation, the DMV does not learn Dennis's license plate numbers, and Dennis does not learn the DMV's list of charged numbers. Because the DMV has the signature Dennis created at the beginning of the year, the DMV is assured Dennis did not lie about the license plates he chose.

The important property of this protocol is that because the DMV never learns the actual values of Dennis's choice of dynamic license plates, it is impossible for the DMV to reconstruct any information about Dennis's paths through the congestion pricing zone. Nonetheless, the DMV can collect the money it is due with confidence.

The remainder of the paper is organized as follows. We begin by reviewing and extending the notion of driver privacy we introduced in our previous paper. Then, we quickly review the properties of the cryptographic primitives we utilize. Next, we formally describe the protocol and discuss its properties. Finally, we discuss the implementation of our proposed congestion pricing scheme. A natural concern is the susceptibility of the design to scams and cheats, and we describe how to make the protocols robust against common attacks. Another significant implementation issue is accommodation of intra-regional travelers, those just passing through. That is, in the early years of adoption of such systems it is likely that they will be piecemeal — areas with congestion tolling in place will be accessible to drivers who cannot be assumed to possess appropriate transponders, for instance. Ensuring that there is a sensible strategy for handling the issue of such "tourists" is essential to any practical implementation of a system such as ours.

II. A REVIEW OF THE NOTION OF DRIVER PRIVACY

In a previous paper, we introduced and developed a notion of driver privacy. We will review and refine these ideas in this section. There has been a great deal of prior work on informational privacy, and there are now reasonably comprehensive ideas about what this should mean, developed both from a legal standpoint and from a cryptographic standpoint. However, there had previously been relatively little work on characterizing how to extend these notions to the context of driving.

The first and most central piece of information of value in this discussion is the *location* of a particular vehicle at a particular time t. However, merely protecting the specific path of the vehicle is insufficient. We must also safeguard *partial information* about the location during a timespan $[t_1, t_2]$. For example, we want also to keep private information that "a particular vehicle has crossed an intersection near the owner's house an odd number of times between 9am and 2pm" since such information indicates that the vehicle owner is not home at 2pm. In formal terms, partial information about a vehicle can be modeled by considering functions of the location of a vehicle and the time. Thus, we must also seek to limit the class of functions of the location which can be learned from data collected by any tolling system. This kind of constraint is reasonable for the purely computational aspects of the protocol.

However, driving is a physical activity which is embedded in a social context that modifies and limits the kinds of privacy guarantees that are reasonable. It is useful to carefully recall the kind of "implicit privacy" which is currently available. On the one hand, most drivers (correctly) assume that in general the path of their vehicle is basically completely unmonitored, and that the path cannot be reconstructed at a later date by some governmental entity except insofar as they

leave other traces of their movements (e.g. geographically localized credit card activity). On the other hand, it is nearly universally legal for a police car to follow a private vehicle for short periods of time without requiring any special dispensation from higher authority. Furthermore, in the event that a driver commits a traffic infraction, they immediately surrender their right to driving privacy and can be tracked and pursued.

There are two important characteristics of this "implicit privacy" that we wish to emphasize as worthy of preservation.

1) The resource costs of tracking are prohibitively high on a large scale. It is a major investment for the state to track any given driver for a sustained basis, and completely impossible for the state to track even a tiny fraction of the total number of drivers.
2) It is difficult for the state to gather information about an individual driver without the driver becoming aware of the monitoring. Since police cars tend to be distinctively marked, it is fairly obvious when a driver is being followed by one. Even when unmarked cars are employed, it often becomes apparent that monitoring is occurring. Only with a very significant use of resources can the state reliably track vehicles in an undetectable fashion.

Therefore, we propose the following rough notion of a system which preserves driver privacy. We require that the idealized technical underpinnings of the system (e.g. the transponder interactions) guarantee the stronger cryptographic level of privacy outlined above — computational hardness of computing functions of the path of a particular vehicle other than those expressly permitted by the protocol (such as the amount of the toll). And moreover we require that the actual reification of the system in a social and technical implementation guarantee the level of "implicit privacy" provided by sporadic police monitoring today.

Naive congestion pricing schemes which rely on the use of cameras or transponders with a fixed unique ID to track vehicle motion fail to preserve either of these characteristics. It becomes trivial for the state to access detailed real-time information about the path of any particular vehicle. We believe that there is an essential qualitative difference in the nature of the privacy available between the situation where a court order and the investment of activity by many police officers is required for tracking, and the situation where an intern at the mayor's office can push a button and track a driver. In addition, in such implementations the monitoring is undetectable to the driver. That is, the driver has no way of knowing whether the information which is being constantly collected is being utilized (or might in the future be utilized) for tracking purposes, or is simply being thrown away after use for toll pricing.

The system we will describe in the following sections satisfies our notion of driver privacy.

III. Infrastructure and protocols

In the next sections, we describe our protocol, which can be used for any kind of variable tolling including both road and congestion pricing. There are two components to our design for a secure privacy-preserving protocol for congestion pricing. The first important choice is a rejection of cameras in favor of radio transponders, modeled on the EZ-pass transmitters in common use in the northeastern United States.[1]. As discussed in the previous section, extensive camera networks are unsuitable for the kinds of privacy we envision. When misused, they allow arbitrary covert surveillance of individual vehicles. Relying on assurances of the goodwill of the monitoring entities is unacceptable.

The second component in our design is the employment of cryptographic algorithms (e.g. to provide secure signatures and encrypted channels) as a foundation on which we can base protocols that offer strong security guarantees.[2]

A. An intuitive sketch of our protocol

The core idea of our solution is reasonably simple. Currently, when an individual registers a car they receive a license plate. Automobile registration is authenticated using official identification of the individual as well as a vehicle identification code. A naive congestion pricing system would employ a ubiquitous network of monitoring devices (e.g. cameras) or require license plates with unique RFID tags to track the path of a specific vehicle in order to assess charges.

Our proposed solution bootstraps from the same procedural base. An individual seeks to register a car or get a residential parking permit. When they do so (either in person or online), they commit to a very long list of "digital license plates". However, the list of such license plates itself is not disclosed to the registering authority — only a "digital commitment" to the list is revealed, which assures the registering authority that the list cannot be changed later. The individual is issued a radio transponder or purchases one at outlets arranged for by the city or state (i.e. convenience stores, consumer electronics stores, gas stations) and places it in the registered vehicle. Using a cell phone or the internet, this transponder is programmed with the previously agreed upon list of "digital license plates". The transponder cycles through the list of license plates, maintaining a different one as the "current license plate" each second. As the motorist moves through the congestion pricing zone, monitoring devices record their digital license plates for use in charging the driver.

However, since the motorist is rapidly cycling their digital license plate number, it is not possible for the network of monitoring devices to track the motion of the motorist through the city. At the end of the billing period, in order to keep their registration active, the motorist must engage in

[1]Note however, although the EZ-pass system uses radio transponders, it does not preserve driver privacy. Our transponders are somewhat more sophisticated

[2]Our constructions rely on making computational assumptions about the difficulty of certain kinds of problems, like factoring large integers. These are standard assumptions that have been well-studied.

a special protocol with the registering authority to compute the charges based on the intersection of the motorist's list of digital license plate numbers and the authority's very long list of numbers collected by the monitoring devices. This special protocol enables this computation to be performed in a "zero-knowledge" fashion, so that the motorist does not disclose any information about their list of numbers other than those in the intersection.

In order to make this work in a fashion which is resistant to tampering on either side, there are a number of cryptographic techniques that must be used. We have alluded to some of them — for instance, the digital commitment the motorist discloses at the beginning of the year prevents her from changing her list later or disavowing her list of numbers at the end of the year. Similarly, we require "zero-knowledge proofs" and "secure multi-party computation algorithms" to perform the computation at the end of the year. We begin the formal description of our protocol by reviewing the cryptographic supports we require.

B. Cryptographic infrastructure

Unless otherwise noted, all of the algorithms listed are efficient probabilistic algorithms. Certain common inputs, such as *security parameters* which indicate how long keys should be, for example, are omitted for clarity. The overall introduction to these primitives is cursory and informal due to limited space.

Blind Signature Scheme. A *blind signature scheme*, first introduced by Chaum [2], consists of a four-tuple of probabilistic algorithms, GEN, SIGN, REQUEST, and VERIFY. The GEN algorithm generates a public key, pk, and a secret key sk. The SIGN(sk) and REQUEST(pk, m) algorithms together form a protocol in which a signer with input sk and signature requester, with input pk and message m interact with one another in order to generate a signature σ. At the end of the protocol, (a) the signer has no knowledge about m or σ, and (b) the requester receives σ, and nothing more. In particular, upon receiving σ at a later point, it is infeasible for the signer to associate σ with the requester; likewise, it is infeasiable for the requester to forge signatures on any other message $m' \neq m$. Finally, the VERIFY(σ', m_i, pk) algorithm runs on input, a pair of strings σ', m' and public key pk and returns either accept or reject. The VERIFY algorithm accepts all signatures generated by the signing algorithm (when run with corresponding public and secret keys). The blind signature scheme in [4] is suitable for our application.

Commitment Scheme. A *commitment scheme* consists of two algorithms, LOCK and OPEN, which are run during two separate phases. Often the following analogy with a metal safe is used: in the commitment phase, the sender locks a message m into a safe and sends the entire safe to a receiver. The message is *hidden* since the receiver cannot crack the safe, and it is *bound* since the sender cannot change the contents of the safe after having given it to the receiver. In the second, *decommitment* phase, the sender gives the safe's key to the receiver, thereby allowing the receiver to read the message. Thus, the LOCK$(x) \rightarrow (\alpha, \beta)$ algorithm takes input, a message x, and produces a pair of strings (α, β) which consists of a commitment α (safe) , and a secret opening value β (key). The recipient, upon receiving β, can open the value by running OPEN$(\alpha, \beta) \rightarrow x$. See [5] for a detailed description of this primitive.

Zero-knowledge Proof System. A *zero-knowledge proof system* consists of two programs, a PROVER and a VERIFIER, which interact in a way that allows the Prover to convince the Verifier of a true statement without revealing any extra information about the statement. The proof system guarantees three properties: (a) any true statement can be proven so that an honest Verifier will accept the proof, (b) a malicious Prover, no matter how it tries to cheat, has a vanishingly small chance of convincing a Verifier of a false statement, (c) a malicious Verifier, no matter how it tries to cheat, will not "learn" anything other than the truth of the statement. For the purpose of this paper, we shall only attempt to prove statements which have short "witnesses" (i.e. NP languages). Such a statement might be, for example, "The value committed in c does not appear in the list $a_1, a_2, \ldots, a_n$." This statement has a short witness since the opening to the commitment c, i.e. β, allows anyone to run OPEN(c, β) and verify the statement. The salient point, then is that, using zero-knowledge proof systems, a Prover can prove the same statement to a verifier, without revealing β, m or anything else other than the verity of the statement!

Since their discovery by Goldreich, Micali and Widgerson [6], zero-knowledge proofs have become an essential tool for designing modern cryptographic protocols. A good introduction is presented in [5].

Secure Multi-party Computation In a secure multi-party computation protocol, there are a series of agents A_i who each possess private information x_i and wish to compute a function $f(x_1, x_2, \ldots, x_n)$. If there existed a trusted third party, the agents could send their private information to the third party, who would perform the computation of f and send back the answer. A secure multi-party computation protocol is a means for simulating the characteristics of this ideal situation in the absense of a trusted third party. In such a protocol, the agents engage in a series of interactions such that with high probability the outcome is the computation of $f(x_1, x_2, \ldots, x_n)$ but it impossible for any particular agent A_i to recover any information about an agent's private information x_i beyond anything which can be inferred from the value of f. Various algorithms for achieving this sort of goal have been around for over a decade, and a good introduction is presented in [5].

C. Hardware infrastructure

Our system involves three physical components.

Car Transponder. Each car is equipped with a *transponder* which can be requested to broadcast its identification string. As mentioned above, a transponder's identification string can periodically change. We say that at time t, the transponder broadcasts a string ρ_t. In addition, the transponder is capable of performing standard authentication procedures to verify

that the request for identification is being made by an authorized party.

Traffic logger. A *traffic logger* L monitors the roads. When a car passes a traffic logger, the logger authenticates itself to the vehicles transponder, requests the transponder's ID and logs the string which it receives. In order to authenticate itself, the logger has a signing key which is signed by the registering authority. This interaction is triggered by a non-specific signal to the logger that a car has passed, for instance from a plate in the road or a simple optical detector.

Registration server. The registering authority D maintains a public signing key, pk for a blind signature scheme, a public signing key pk' to sign certificates, and a website.

D. The protocol

There are three simple protocols in our system.

Registration Protocol

Common Input: D's public token signing key pk, for a blind signature scheme. D's public logger signing key pk' for a signature scheme. Let n be the number of minutes in the registration period (e.g., $n = 60 * 24 * 365$ if the registration period is one year).

At the beginning of each registration period, the Transponder and D perform the following steps.

1) **(Generate tokens)** The Transponder randomly generates a sequence of k-bit token strings, $\{m'_{1,a}, m'_{1,b}, \ldots, m'_{n,a}, m'_{n,b}\}$, two for each minute of the time period. These are the "digital license plates" the motorist uses during the registration period. The Transponder computes a commitment, $\text{LOCK}(m'_{i,j}) = (\alpha_{i,j}, \beta_{i,j})$ of each token, and then sends the sequence of commitments $(\alpha_{1,a}, \ldots, \alpha_{n,b})$ to D.
2) **(Proof Challenge)** For each index $i = 1, \ldots, n$ of the registration period, D randomly assigns the variable c_i to a or b and then sends to the Transponder a list of requests, $(c_1, \ldots, c_n)$.
3) **(Proof Response)** The Transponder reveals the commitment m'_{i,c_i} for each challenge c_i requested by sending β_{i,c_i}. This is a proof that the Transponder actually knows each of the tokens that it is registering. Let the sequence of remaining, unopened tokens be re-labelled $\mathcal{M} = \{m_1, m_2, \ldots, m_n\}$, and let the sequence of corresponding commitments be labelled $\mathcal{C} = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$.
4) **(Blind Signature)** The Transponder and D engage in a blind signature protocol in order to generate signatures for each pair (i, m_i). Let σ_i denote the signature for (i, m_i) which the Transponder receives.

Congestion tolling monitoring event

Common Input: D's public token signing key pk, D's public logger signing key pk' for a signature scheme.

A traffic logger L records an event at time t involving a vehicle v as follows:

1) **(Authentication)** L authenticates itself by presenting a certificate signed by pk' and requests v to send its current token.
2) **(Reply)** The vehicle v sends (t, σ_t, m_t) to L, encrypted in L's public key pk.
3) **(Verify)** The Logger verifies that σ_t is a valid signature of m_t generated from D's token-signing key pk, and records (t, σ_t, m_t). If the signature does not verify, then L calls the local police office to indicate that a malfunctioning Transponder has passed through its location.
4) **(Post)** At the end of each day, D posts and timestamps a list, l_d of all of the events that have been logged by all traffic loggers for that day d.

Renewal

Common Input: D's public token-signing key pk, for a blind signature scheme. D's public logger-signing key pk' for a signature scheme.

At the end of each billing period, and in order to keep the registration current, a Transponder and D engage in the following steps:

1) **(Determine Tolls)** The Transponder and D engage in a secure two-party computation to determine the toll owed based on the intersection between its private sequence of tokens $\mathcal{M} = \{m_1, \ldots, m_n\}$ and the public list of traffic events, $\mathcal{Y} = \{l_1, \ldots, l_{365}\}$.
2) **(Prove consistency)** The Transponder then proves in zero-knowledge that the tolls computed are correct. This is done by proving in zero-knowledge that the commitments in the list $\mathcal{C}$ only intersect with $\mathcal{Y}$ at $(m_{e_1}, \ldots, m_{e_k})$ and that this intersection was provided as input to the secure two-party computation.
3) If D accepts the proof, the Transponder pays the tolls and then repeats the Registration process. If any step of the protocol is not completed, then the registration renewal is considered to have failed.

We emphasize that our protocols are to be executed sequentially. In other words, D must perform the Registration and Renewal process one-at-a-time with each of the Transponders, and the Logger and Transponder should also engage in only one interaction at a time. [3]

E. Why Privacy is Preserved

Intuitively, the information recorded by a traffic logger does not identify a vehicle *per se*. During the initial registration phase, the registering authority does not learn the values of the tokens which are sent, nor its own signatures of those tokens. It only learns a *cryptographic commitment* to the token which hides its value; this value, however, can be used later to guarantee that a motorist pays all of his tolls.

[3] While there are various approaches to proving that a protocol is secure in a concurrent setting when D interacts with several Transponders at the same time, this type of discussion is beyond the scope of this paper.

F. How Tolls are Paid

At the end of the billing period, the motorist is required to reconcile any outstanding tolls owed. During this period, the motorist and the registering authority engage in a secure two-party computation to compute the toll. In order to be complete, the motorist must furthermore prove via a zero-knowledge interaction that all such charges have been acknowledged. A driver who refuses to complete the renewal protocol can be barred from using the public roads through traditional mechanisms (i.e., license suspension).

IV. Implementation

Various modifications to existing bureaucratic, legal, and technical infrastructure are necessary to implement the scheme we describe. In our presentation of the protocol, we have outlined the new regulatory behaviors of the registry of motor vehicles and the technical behavior of the transponders and loggers. Implicit in this are certain required legal modifications.

Our general view is that all of the required changes are eminently reasonable and achievable based on observation of the experiences of various municipalities in rapidly constructing infrastructure for automated toll collection. Just as in our proposed system, modifications to registration policies, technical infrastructure, and enforcement behaviors were necessary to implement these toll collection systems. In fact, the obstacles to creating such systems were significantly greater than the ones our system faces precisely because of the lack of prior examples.

A. Malicious attacks

Our system is secure against a variety of malicious attacks. It is important, however, to be clear about the kinds of security guarantees we provide. We can prove that the mathematical protocols we describe achieve some notion of privacy and guarantee correctness. However, as we are describing a system which must be reified in a physical and bureaucratic implementation, there are of course limits to the kinds of guarantees we can expect simply from analysis of the protocols.

On the other hand, in our situation, we can leverage the power of the "real world" to protect against certain kinds of attacks — for instance, motorists who might choose to simply remove their transponders altogether face the threat of police enforcement. Cars traveling without emitting the requisite radio signals can be tagged as driving without a transponder and fined. Moreover, by restricting the frequency bands on which transponders and loggers transmit their information, we can prevent various spoofing attacks (see below). To avoid committing to a particular implementation technology at this point, our suggestions of this nature should be regarded as "qualitative" and necessarily requiring interpretation in the specific technical context of implementation.

Outstanding charges. Recall that drivers receive "unofficial" notification from their transponder when a logger has recorded its plate number, and so we imagine that in standard usage following such an event the driver would log on to confirm and pay. We may incentivize prompt payment (by account reconciliation) by price. For example, if a charge is paid (reconciled) with 24 hours of incurring it, the price might be p; if paid within one week, it might be $1.5p$; if paid monthly $2p$. Drivers could use a public website to reconcile their accounts, or query as to whether they have any unpaid congestion charges outstanding. A variety of payment options could be established, many of them automated, to conform with the individual's access to credit or preference for prepaid options.

Spoofing and "man-in-the-middle" attacks. One might be concerned that a rogue transponder v_1, while passing another transponder v_2, could attempt to read v_2's token, and then passes it off as its own token for that time period. Indeed, this type of attack is a serious one.

For this reason, we require any Logger to authenticate itself to the transponder and send the token over an encrypted channel. Hence, a rogue reader will be unable to convince a Transponder to send its token.

Additionally, we may legally prohibit Transponders from broadcasting outside a certain frequency band, and design the Loggers to send their requests on a different frequency band. (See the next point for how this can be enforced.) Restricting transmission bands also provides protection against attacks in which a malicious transponder sits between a logger and another (innocent) transponder, ferrying messages back and forth.[4]

Broken Transponder. Suppose the owner of a vehicle tampers with or disconnects the car transponder when using the roads. In such a situation, the Traffic logger has no information to record about this vehicle when it commits an infraction. We consider this type of attack no different than if an individual removes or obscures his license plates. It is a serious crime to drive without a license plate; it can be just as serious to drive without a transponder. We rely on the classical (and indispensible) presence of the highway patrol in order to solve this problem. Highway patrol officers can be equipped with devices that monitor whether a vehicle has a functioning transponder. Moreover, since the Loggers will receive information that a car without a transponder has passed, a sufficiently dense network of Loggers will allow the tracking of such an offending vehicle.

Challenges to toll charges. In the current system, disputing a toll charge requires considerable proof on the part of the accused that the charge was issued incorrectly. Our system is no different in this respect. One might, however, require the Traffic logger to post calibration information which indicates proper functioning to a website every day, or to record, along with each tolling event, calibration data which can be used to support the charges.

Corrupt registering authority. Since the recorded tokens do not contain any private information, every day, each of the traffic loggers sends its daily event log to a central server.

[4]We should note that traditional solutions to this type of man-in-the-middle attack do not apply because, although the Logger is "authenticated" to the Transponder, the requirement for Transponder anonymity makes the converse impossible.

This server timestamps the log using a third party time-stamping service, and posts the list to a publicly available bulletin board (or internet site). This prevents D from generating false events, *after the fact*, in order to gather information about the location of a vehicle, say at the behest of a district attorney involved in a case against a vehicle's owner.

B. Tourists and incremental adoption

An essential issue to confront is the handling of sporadic motorists, those just passing through a congestion pricing zone, who have not obtained a registered transponder. Given the local nature of decision-making regarding the implementation of congestion pricing systems, it is highly likely that there will be a lengthy period of time in which there will be a sizeable population of occasional visitors without suitable transponders. We will refer to such sporadic users as "tourists", although we intend to denote a potentially broader class than actual tourists. One wishes to be able to successfully charge such motorists in a fashion without either unduly burdening the tourist (e.g. requiring full participation in the transponder registration process) or opening up opportunities for residents to cheat the system.

The most straightforward solution is to require the tourists to purchase a disposable temporary transponder (e.g. a smart RFID device) at a gas station or convenience store. However, this transponder is not tied to any sort of registration protocol. Instead, the transponder simply has a fixed cost, perhaps the maximum charge for the time period of its applicability. The transponder has a crytographically secure timestamp and an interval of validity, and as the tourist vehicle passes a logging device the transponder engages in a short interaction proving that the transponder is valid. By equipping the transponder with a long list of timestamped certificates, this can be done in a fashion which does not permit tracking of the tourist vehicle. If full variable tolling is desired for tourists, the process can be augmented by permitting tourists to apply for a rebate by engaging in a version of the billing protocol.

V. Relationship to Other Cryptographic Protocols for Transportation

Many cryptographic notions have been applied to problems in transportation. As early as 1992, David Chaum et.al. [2] proposed and built a prototype anonymous electronic toll-system, called Dynacash, and installed it in Holland and in Japan. The critical element of his system involved smartcard technology which could be "charged up" with digital cash, and automatically debited as a vehicle passed a toll-booth. This system is far more respectful of driver privacy than a system like EZ-Pass, which must record a *static* special-purpose vehicle ID every time the vehicle passes through a toll-booth. Almost all of the "electronic cash" technologies that have subsequently been developed could be employed in this fashion. In such systems, enforcement of toll violations must be handled by an external mechanism, typically a camera. In addition, such "electronic cash" systems do not preserve sufficient information to support general usage pricing schemes. Such a system would however be a plausible means to implement the "tourist protocol" we sketched above.

Chaum has also proposed a trusted-hardware model in which each smartcard contains a tamper-proof "observer" chip which is installed and certified by the registering authority. In this model, credentials can be stored on the smartcard. A credential is simply an authorization which has been assigned to its holder. In our contexts, credentials can model the right to use a highway at a specific time (only granted to vehicles traveling under a set speed limit), or the right to cross an intersection (only granted when the light is green). Verifiers along the entire route can check whether each vehicle is authorized to travel on that route. The problem, however, the process of presenting credentials, when repeated a few times, allows the verifier to "link" the credentials of a single vehicle to one another.

Thus, neither of these classes of protocols allow the implementation of the kind of congestion pricing system we require. More recently, the authors have learned of a paper by Bangerter, Camenisch, and Lysyanskaya [7] describing a framework for anonymous and unlinkable releasing of "credential information." Their protocols, which can perhaps be used for certain parts of our system, suggest promising directions for efficient implementations.

VI. Conclusions

We envision a future in which every car has a signal transponder and there are virtually ubiquitous state-managed monitoring devices spread throughout public road space. Drivers will be charged tolls that precisely reflect their usage of public road transporation infrastructure; rather than simply congestion pricing for downtown areas, all driving will be assessed charges depending on the time, location, and vehicle specifications. There are grave and obvious threats to the privacy of the individual in such a situation, as standard implementations of such pricing schemes involve comprehensive monitoring and tracking of each vehicle. We believe that there is an essential right to "locational privacy" for individuals.

It is sometimes argued that the benefits of congestion pricing systems outweight the costs of sacrificing driver privacy. One of the main achievements of this paper is to demonstrate that this is an unnecessary choice. We have introduced a protocol which allows the collection of arbitarily nuanced variable tolls while guaranteeing the preservation of locational privacy for individuals. Our protocol is relatively straightforward to implement using existing technology, and robust against various commonplace attacks. The existence of this protocol serves to demonstrate that it is not necessary to surrender driver privacy in order to achieve sophisticated congestion pricing systems.

VII. Acknowledgements

We would like to thank Abhi Shelat for a fruitful previous collaboration which led to this research and Karl Schafer for

useful conversations about the formalization of the idea of driver privacy. We would also like to express our gratitude to Razvan Surdulescu for a careful reading of the paper and useful criticism.

REFERENCES

[1] T. transportation institute, "Urban mobility information," http://mobility.tamu.edu/ums/congestion_ data/tables/national/table_ 4.pdf.

[2] D. Chaum, "Blind signatures for untraceable payments," in *CRYPTO '82*, 1982, pp. 199–203.

[3] A. J. Blumberg, L. S. Keeler, and abhi shelat, "Automated traffic enforcement which preserves driver privacy," in *ITSC 2005*, 2005, vienna, Austria.

[4] J. Camenisch, M. Koprowski, and B. Warinschi, "Efficient blind signatures without random oracles," in *In Forth Conference on Security in Communication Networks - SCN '04*, 2004.

[5] O. Goldreich, *Foundations of Cryptography*. Cambridge University Press, 2004, vol. 2, ch. 7 (General Cryptographic Protocols), pp. 599–759.

[6] O. Goldreich, S. Micali, and A. Wigderson, "Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proofs," *Journal of the ACM*, vol. 38, no. 3, 1991.

[7] E. Bangerter, J. Camenisch, and A. Lysyanskaya, "A cryptographic framework for the controlled release of certified data," in *Twelfth International Workshop on Security Protocols*, Apr 2004, cambridge, England.